

🖉 Minnesota State University mankato

Minnesota State University, Mankato Cornerstone: A Collection of Scholarly and Creative Works for Minnesota State University, Mankato

All Graduate Theses, Dissertations, and Other Capstone Projects

Graduate Theses, Dissertations, and Other Capstone Projects

2020

Theory of Principal Components for Applications in Exploratory Crime Analysis and Clustering

Daniel Silva Minnesota State University, Mankato

Follow this and additional works at: https://cornerstone.lib.mnsu.edu/etds

Part of the Applied Statistics Commons, Multivariate Analysis Commons, and the Statistical Theory Commons

Recommended Citation

Silva, D. (2020). Theory of principal components for applications in exploratory crime analysis and clustering [Master's thesis, Minnesota State University, Mankato]. Cornerstone: A Collection of Scholarly and Creative Works for Minnesota State University, Mankato. https://cornerstone.lib.mnsu.edu/etds/996/

This Thesis is brought to you for free and open access by the Graduate Theses, Dissertations, and Other Capstone Projects at Cornerstone: A Collection of Scholarly and Creative Works for Minnesota State University, Mankato. It has been accepted for inclusion in All Graduate Theses, Dissertations, and Other Capstone Projects by an authorized administrator of Cornerstone: A Collection of Scholarly and Creative Works for Minnesota State University, Mankato. Theory of Principal Components for Applications

in Exploratory Crime Analysis and Clustering

By

Daniel Silva

A Thesis Submitted in Partial Fulfillment of the

Requirements for the Degree of

Master of Science

In

Applied Statistics

Minnesota State University, Mankato

Mankato, Minnesota

May 2020

Date: April 10th 2020

Title: Theory of Principal Components for Applications in Exploratory Crime

Analysis and Clustering

Student's Name: Daniel Silva

This thesis has been examined and approved by the following members of the student's committee.

Advisor/Chairperson,

Dr Galkande (Iresha) Premarathna

Professor of Statistics,

Minnesota State University, Mankato.

Committee Member,

Dr Mezbahur Rahman,

Professor of Statistics,

Minnesota State University, Mankato.

Committee Member,

Dr Deepak Sanjel,

Professor of Statistics,

Minnesota State University, Mankato.

Abstract

Theory of Principal Components for Applications in Exploratory Crime Analysis and Clustering By Daniel Silva Master of Science in Applied Statistics Minnesota State University, Mankato Mankato, Minnesota, 2020

The purpose of this paper is to develop the theory of principal components analysis succinctly from the fundamentals of matrix algebra and multivariate statistics. Principal components analysis is sometimes used as a descriptive technique to explain the variance-covariance or correlation structure of a dataset. However, most often, it is used as a dimensionality reduction technique to visualize a high dimensional dataset in a lower dimensional space. Principal components analysis accomplishes this by using the first few principal components, provided that they account for a substantial proportion of variation in the original dataset. In the same way, the first few principal components can be used as inputs into a cluster analysis in order to combat the curse of dimensionality and optimize the runtime for large datasets. The application portion of this paper will apply these methods to a US Crime 2018 dataset extracted from the Uniform Crime Reports on the FBI's website.

Table of Contents

ABSTRACT	1
TABLE OF CONTENTS	Il
CHAPTER 1 INTRODUCTION	1
1.1 THEORY STRUCTURE	2
1.2 Application Background and Structure	3
CHAPTER 2 MATRIX ALGEBRA	7
2.1 VECTORS	7
2.2 MATRICES	14
CHAPTER 3 MULTIVARIATE POPULATION THEORY	43
3.1 Population Random Matrix	43
3.2 Population Random Vector, Mean Vector, Variance-Covariance Matrix, and Correlation	ON MATRIX
3.2.1 Population Random Vector	44 44
<i>3.2.2 Probability Density Functions</i>	45
3.2.3 Population Parameters	46
3.2.4 Independent Random Variables	48
3.2.5 Population Mean Vector	49
3.2.6 Population Variance-Covariance Matrix	50
3.2.7 Population Standard Deviation Matrix	51
3.2.8 Population Correlation Matrix	52
3.3 POPULATION MEAN VECTOR AND VARIANCE-COVARIANCE MATRIX FOR LINEAR COMBINATIONS OF	
Continuous Random Variables	54

3.3.1 Linear Combination	54
3.3.2 Population Parameters for Linear Combinations	55
3.3.3 q Linear Combinations	62
3.3.4 Population Mean Vector for q Linear Combinations	63
3.3.5 Population Variance-Covariance Matrix for q Linear Combinations	65
3.4 Population Random Vector, Mean Vector, and Variance-Covariance Matrix for Standari	DIZED
Continuous Random Variables	67
3.4.1 Population Random Vector for Standardized Continuous Random Variables	67
3.4.2 Population Parameters for Standardized Continuous Random Variables	69
3.4.3 Population Mean Vector for Standardized Continuous Random Variables	72
3.4.4 Population Variance-Covariance Matrix for Standardized Continuous Random Varia	ables.73
3.5 MEAN VECTOR AND VARIANCE-COVARIANCE MATRIX FOR LINEAR COMBINATIONS OF STANDARDIZED	
Continuous Random Variables	75
3.5.1 Linear Combination of Standardized Continuous Random Variables	75
3.5.2 Population Parameters for Linear Combinations of Standardized Continuous Rando)m
Variables	76
3.5.3 q Linear Combinations of Standardized Continuous Random Variables	79
3.5.4 Population Mean Vector for q Linear Combinations of Standardized Continuous Rai	ndom
Variables	80
3.5.5 Population Variance-Covariance Matrix for q Linear Combinations of Standardized	
Continuous Random Variables	83
CHAPTER 4 MULTIVARIATE SAMPLE THEORY	
4.1 Organization of Multivariate Sample Data	85
4.2 RANDOM SAMPLES	87
4.2.1 Univariate Random Sample	87
4.2.2 Multivariate Random Sample	88

iii

4.2.3 Multivariate Random Sample Matrix	89
4.3 SAMPLE STATISTICS	
4.4 SAMPLE MEAN VECTOR, VARIANCE-COVARIANCE MATRIX, AND CORRELATION MATRIX	94
4.4.1 Sample Mean Vector	
4.4.2 Sample Variance-Covariance Matrix	
4.4.3 Sample Standard Deviation Matrix	
4.4.4 Sample Correlation Matrix	
4.5 SAMPLE MEAN VECTOR AND VARIANCE-COVARIANCE MATRIX FOR LINEAR COMBINATIONS OF COM	ITINUOUS
RANDOM VARIABLES	104
4.5.1 Linear Combination	104
4.5.2 Sample Statistics for Linear Combinations	
4.5.3 q Linear Combinations	110
4.5.4 Sample Mean Vector for q Linear Combinations	
4.5.5 Sample Variance-Covariance Matrix for q Linear Combinations	
4.6 Standardized Random Samples	114
4.6.1 Standardized Univariate Random Sample	
4.6.2 Standardized Multivariate Random Sample	115
4.6.3 Standardized Multivariate Random Sample Matrix	
4.7 SAMPLE STATISTICS FOR STANDARDIZED SAMPLES	118
4.8 SAMPLE MEAN VECTOR AND VARIANCE-COVARIANCE MATRIX FOR STANDARDIZED SAMPLES	121
4.8.1 Sample Mean Vector for Standardized Samples	
4.8.2 Sample Variance-Covariance Matrix for Standardized Samples	124
4.9 SAMPLE MEAN VECTOR AND VARIANCE-COVARIANCE MATRIX FOR LINEAR COMBINATIONS OF STA	NDARDIZED
SAMPLES	130
4.9.1 Linear Combination of Standardized Samples	130
4.9.2 Sample Statistics for Linear Combinations of Standardized Samples	

iv

4.9.3 q Linear Combinations of Standardized Samples	133
4.9.4 Sample Mean Vector for q Linear Combinations of Standardized Samples	134
4.9.5 Sample Variance-Covariance Matrix for q Linear Combinations of Standardized Sa	mples
	135
CHAPTER 5 PRINCIPAL COMPONENTS ANALYSIS	138
5.1 INTRODUCTION	138
5.2 Population Principal Components	138
5.3 Population Principal Components for Standardized Continuous Random Variables	149
5.4 SAMPLE PRINCIPAL COMPONENTS	153
5.5 SAMPLE PRINCIPAL COMPONENTS FOR STANDARDIZED SAMPLES	156
CHAPTER 6 RESULTS AND DISCUSSION	160
6.1 R Programming Language	160
6.2 Univariate Distribution Analysis	160
6.2.1 Descriptives for US Crime 2018	160
6.2.2 Distributions of US Crime 2018	
6.2.2.1 Murder Distribution	161
6.2.2.2 Rape Distribution	164
6.2.2.3 Robbery Distribution	166
6.2.2.4 Assault Distribution	168
6.2.2.5 Burglary Distribution	170
6.2.2.6 Larceny Distribution	172
6.2.2.7 Vehicle Distribution	174
6.3 BIVARIATE DISTRIBUTION ANALYSIS	176
6.3.1 Correlation Matrix for US Crime 2018	176
6.3.2 Contour-Scatter Matrix	177
6.4 Multivariate Distribution Analysis	

6.4.1 Testing Multivariate Normality	178
6.5 SAMPLE PCA FOR STANDARDIZED US CRIME 2018	179
6.5.1 Descriptives for Standardized US Crime 2018	179
6.5.2 Sample PCA for Standardized US Crime 2018	180
6.5.2.1 Explained Standardized Sample Variance by Principal Component for US Crime 2018	180
6.5.2.2 Sample Principal Components for Standardized Crime 2018	
6.5.2.3 Correlation Matrix for Sample Principal Components and Standardized Crime 2018	
Characteristics	185
6.5.2.4 Scatterplots for Sample Principal Components from Standardized US Crime 2018	187
6.6 k-Means Clustering Method	192
6.6.1 Choosing k	194
<i>6.6.2 k - Means, k = </i> 3	195
6.6.2.1 <i>k</i> -Means, <i>k</i> = 3, Cluster Sizes	
6.6.2.2 k-Means, $k = 3$, Differences in Cluster Assignments	195
6.6.2.3 <i>k</i> -Means, <i>k</i> = 3, Sample Cluster Mean Vectors	196
6.6.2.4 <i>k</i> -Means, $k = 3$, Scatterplots on $y1$, $y2$, $y3$	198
6.6.2.5 k-Means, $k = 3$, Scatterplots on Original Crime 2018 Dimensions	200
6.7 HIERARCHICAL CLUSTERING METHODS	202
6.7.1 Agglomerate Clustering Methods	202
6.7.1.1 Average and Ward's Method	203
6.7.2 Euclidean Distance Matrices	205
6.7.2.1 Euclidean Distance Matrix for Standardized US Crime 2018	205
6.7.2.2 Euclidean Distance Matrix for <i>y</i> 1, <i>y</i> 2, <i>y</i> 3	206
6.7.3 Wards Method	206
6.7.3.1 Choosing <i>k</i>	206
6.7.3.2 Ward, <i>k</i> = 3	207
6.7.4 Average Method	216
6.7.4.1 Choosing <i>k</i>	216

vi

6.7.4.2 Average, $k = 3$	217
6.7.5 Comparing Ward and Average Dendrograms Using Tanglegrams	224
6.7.5.1 Ward, Input S. Crime 2018 vs. Ward, Input <i>y</i> 1, <i>y</i> 2, <i>y</i> 3	224
6.7.5.2 Average, Input S. Crime 2018 vs. Average, Input y1, y2, y3	225
6.7.5.3 Ward, Input S. Crime 2018 vs. Average, Input S. Crime 2018	226
6.7.5.4 Ward, Input <i>y</i> 1, <i>y</i> 2, <i>y</i> 3 vs. Average, Input <i>y</i> 1, <i>y</i> 2, <i>y</i> 3	227
6.8 Comparison of <i>k</i> -Means, Ward, and Average	228
6.8.1 k - Means, Ward, and Average, $k = 3$, Scatterplots on y1, y2 and Cluster Sizes	228
CHAPTER 7 CONCLUSION AND FUTURE STUDY	230
REFERENCES	233
APPENDIX	235

Chapter 1

Introduction

Principal components analysis (PCA) is multivariate statistical method that seeks to transform a set of correlated variables $X_1, X_2, ..., X_p$ into a new set of uncorrelated variables $Y_1, Y_2, ..., Y_p$ that retain the total system variation. These new variables are called the principal components. Each principal component $Y_1, Y_2, ..., Y_p$ is a distinct linear combination of the original variables $X_1, X_2, ..., X_p$ derived in decreasing order of importance in the sense that Y_1 accounts for as much of the variation in the original system amongst all other linear combinations $Y_2, ..., Y_p$. Then Y_2 is chosen to account for as much as possible of the remaining system variation, subject to being uncorrelated with Y_1 . Analogously, Y_i is chosen to account for as much as possible of the remaining system variation, subject to being uncorrelated with $Y_1, Y_2, ..., Y_{i-1}$.

The general hope of PCA is that the first few components will account for a substantial proportion of the variation in the original system, $X_1, X_2, ..., X_p$, and can, consequently, be used to provide a lower-dimensional summary of these variables [1, p. 41]. These first few principal components may then replace the original $X_1, X_2, ..., X_p$ and can be used for descriptives, graphical interpretations, and even inputs into another analysis, with minimal loss of information. That is why principal components analysis is often thought of as a dimensionality reduction technique as

well as an interpretive aid in explained the original variables.

1.1 Theory Structure

In order to get a proper treatment of PCA, one needs a couple preliminaries including matrix algebra, multivariate population theory, and multivariate sample theory.

Matrix algebra is the backbone of multivariate statistics. Chapter 2 devotes itself to covering all essential notations and concepts necessary to understand later chapters. This includes, but is not limited to, vector/matrix notations, inner-product, matrix multiplication, independence, square matrices, orthogonal matrices, eigenvalues and eigenvectors, and matrix maximization of quadratic forms.

Covering matrix algebra before multivariate population theory is critical because it bridges the gap from one's knowledge of univariate population theory to multivariate population theory. Chapter 3, Multivariate Population Theory, covers population random matrices, random vectors, mean vectors, variance-covariance and correlations matrices, and the corresponding theory related to linear combination used directly in the treatment of population PCA. Further, the same topics, as above, are extended to standardized multivariate populations.

Chapter 4, Multivariate Sample Theory, follows directly from Chapter 3. It is paramount in understanding how one goes from population principal components to sample principal components. New concepts of multivariate random samples will be derived from concepts of matrix algebra, multivariate population theory, and univariate random samples learned in one's previous coursework. Then, the sample equivalents to Chapter 3 will be covered; including those related to standardized multivariate populations and linear combinations.

Chapter 5 is devoted to the main topic of PCA. Here we will cover population principal components for unstandardized and standardized continuous random variables. Similarly, we will cover sample principal components for unstandardized and standardized multivariate random samples.

1.2 Application Background and Structure

Local law enforcement agency across the United States collect data on violent and property crimes. Every year, the FBI compiles, publishes, and archives this data in the Uniform Crime Reports (UCR). The UCR Program's primary objective is to generate reliable information for use in law enforcement administration, operation, management, and analytics.

Violent crime definitions according to the FBI are:

- Murder and nonnegligent manslaughter: the willful (nonnegligent) killing of one human being by another.
- Rape: The penetration, no matter how slight, of the vagina or anus with any body part or object, or oral penetration by a sex organ of another person, without the consent of the victim.

- Robbery: The taking or attempting to take anything of value from the care, custody, or control of a person or persons by force or threat of force or violence and/or by putting the victim in fear.
- Aggravated assault: An unlawful attack by one person upon another for the purpose of inflicting severe or aggravated bodily injury. This type of assault usually is accompanied by the use of a weapon or by means likely to produce death or great bodily harm. Simple assaults are excluded.

Property crime definitions according to the FBI are:

- Burglary (breaking or entering): The unlawful entry of a structure to commit a felony or a theft. Attempted forcible entry is included.
- Larceny-theft (except motor vehicle theft): The unlawful taking, carrying, leading, or riding away of property from the possession or constructive possession of another. Examples are thefts of bicycles, motor vehicle parts and accessories, shoplifting, pocket-picking, or the stealing of any property or article that is not taken by force and violence or by fraud. Attempted larcenies are included. Embezzlement, confidence games, forgery, check fraud, etc., are excluded.
- Motor vehicle theft: The theft or attempted theft of a motor vehicle. A motor vehicle is self-propelled and runs on land surface and not on rails.
 Motorboats, construction equipment, airplanes, and farming equipment are specifically excluded from this category [2].

For our application, Chapter 6, we shall use the UCR's US Crime 2018 data for metropolitan statistical areas. Where a metropolitan statistical area is defined by a city with surrounding suburbs that are connected by some economic factors. One disclaimer is our analysis is not meant to rank local or federal law enforcement agencies based on the crime rates in their respective regions. Our analysis is only meant to group metropolitan statistical areas with similar crime profiles and compare their group averages to each-other and to the national averages. Also, note that crimes are generally underreported.

The first step in our analysis will be of a univariate nature. We will calculate descriptives and assess the shape of each of the seven crime distributions. For example, checking whether the parent distribution is perhaps normal or even lognormal. In addition, we will look at the tail-ends of the distributions checking for univariate outliers. The second step is a bivariate distribution analysis. We will graphically visualize the correlation matrix. In addition, we will look at contour- and scatter- plots of the pairs of variables. The third step will be a short multivariate distribution analysis where we will solely test for multivariate normality.

Next, we will standardize the US Crime 2018 data to prepare it for PCA. It is common practice to do this when the ranges of the variables are largely different. Once this is done, we can calculate the sample principal components. Topics of interest are explained variance by sample principal component and contributions of standardized variables to each sample principal component. Also, one can attempt to interpret the sample principal component dimensions in the context of the subject matter--crime. Then, one can look at correlations of standardized variables with the sample principal components. Finally, one can create scatterplots of the first few sample principal components and look for clusters of metropolitan statistical areas or potential multivariate outliers.

After this we will use cluster analysis to attempt to meaningfully group (or profile) metropolitan statistical areas with similar crime attributes. We will use two sets of inputs (1) the Standardized Crime 2018 variables and (2) the first three sample principal components. Three cluster algorithms will be used *k*-Means, Ward's method, and Average method with both sets of inputs. This will leave use with six cluster assignments to compare and contrast graphically and via their respective cluster mean vectors.

Chapter 2

Matrix Algebra

2.1 Vectors

Definition 2.1.1 (Vector). $An \times 1$ dimensional array $\mathbf{x}_{(n \times 1)}$ of *n* real numbers

 $x_1, x_2, ..., x_j, ..., x_n$ (n - tuple) is called a **vector**, and in general, is denoted by a boldfaced, lowercase letter. It is written as

$$\mathbf{x}_{(n\times 1)} = \begin{bmatrix} x_1\\x_2\\\vdots\\x_j\\\vdots\\x_n\\\vdots\\x_n \end{bmatrix}_{(n\times 1)}$$

[3, pp. 49, 82]*.*

A vector $\mathbf{x}_{(n \times 1)}$ can be represented geometrically as a directed line in n dimensions with component x_1 along the 1th axis, x_2 along the 2nd axis,..., x_j along the *j*th axis,..., and x_n along the *n*th axis [3, p. 50].

Definition 2.1.2 (Vector Transpose). $A \to n$ dimensional array $\mathbf{x}'_{(1 \times n)}$ of n real

numbers $x_1, x_2, ..., x_j, ..., x_n$ (n - tuple) is called a vector transpose. It is written as

$$\mathbf{x}'_{(1\times n)} = \begin{bmatrix} x_1, x_2, \dots, x_j, \dots, x_n \end{bmatrix}$$

where the prime denotes the operation of transposing a column $\underset{(n \times 1)}{\mathbf{x}}$ to a row $\underset{(1 \times n)}{\mathbf{x}'}$

[3, p. 49]*.*

Definition 2.1.3 (Zero-Vector). $\underset{(n \times 1)}{\mathbf{0}}$ vector is a $n \times 1$ dimensional array of 0's. It is

written as

$$\mathbf{0}_{(n\times1)} = \begin{bmatrix} \mathbf{0}_1 \\ \mathbf{0}_2 \\ \vdots \\ \mathbf{0}_j \\ \vdots \\ \mathbf{0}_n \\ (n\times1) \end{bmatrix}$$

often thought of as the origin in n - space.

Definition 2.1.4 (One Vector). $\mathbf{1}_{(n \times 1)}$ vector is a $n \times 1$ dimensional array of 1's. It is

written as

$$\mathbf{1}_{(n\times 1)} = \begin{bmatrix} \mathbf{1}_1 \\ \mathbf{1}_2 \\ \vdots \\ \mathbf{1}_j \\ \vdots \\ \mathbf{1}_n \\ (n\times 1) \end{bmatrix}$$

product $\underset{(n \times 1)}{cx}$ is a vector with *j*th entry cx_j . It is written as

$$c\mathbf{x}_{(n\times1)} = c \cdot \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_j \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} cx_1 \\ cx_2 \\ \vdots \\ cx_j \\ \vdots \\ cx_n \end{bmatrix}$$

[3, pp. 50, 82].

Definition 2.1.6 (Vector Addition). *The sum of two vectors* $\mathbf{x}_{(n \times 1)}$ *and* $\mathbf{y}_{(n \times 1)}$, *each*

having the same number of entries, is the vector

$$\mathbf{z}_{(n\times 1)} = \mathbf{x}_{(n\times 1)} + \mathbf{y}_{(n\times 1)}$$
 with *j*th entry $z_j = x_j + y_j$

That is,

$$\mathbf{z}_{(n\times1)} = \mathbf{x}_{(n\times1)} + \mathbf{y}_{(n\times1)} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_j \\ \vdots \\ x_n \end{bmatrix} + \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_j \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} x_1 + y_1 \\ x_2 + y_2 \\ \vdots \\ x_j + y_j \\ \vdots \\ x_n + y_n \end{bmatrix} = \mathbf{x} + \mathbf{y}_{(n\times1)}$$

[3, pp. 51, 83]*.*

The sum of two vectors emanating from the origin $\underset{(n \times 1)}{\mathbf{0}}$ is the diagonal of the parallelogram formed with the two original vectors as adjacent sides [3, p. 51].

Definition 2.1.7 (Vector Space). *The space of all real* n – *tuples (vectors), with scalar multiplication and vector addition, is called a* **vector space** [3, p. 83].

Definition 2.1.8 (Linear Span). The vector

$$\mathbf{y}_{(n\times 1)} = a_1 \mathbf{x}_1 + a_2 \mathbf{x}_2 + \dots + a_k \mathbf{x}_k + \dots + a_p \mathbf{x}_p$$
$$(n\times 1) (n\times 1) (n\times 1) (n\times 1) (n\times 1)$$

is a linear combination of the vectors x₁, x₂, ..., x_k, ..., x_p in Rⁿ where
a₁, a₂, ..., a_k, ..., a_p are real. The set of all linear combinations of x₁, x₂, ..., x_k, ..., x_p is called their linear span, denoted, span(x₁, x₂, ..., x_k, ..., x_p) [3, p. 83], [4, p. 114].
Definition 2.1.9 (Linearly Dependent). A set of vectors x₁, x₂, ..., x_k, ..., x_p is said to be linearly dependent if there exist p numbers (a₁, a₂, ..., a_k, ..., a_p), not all zero, such that

$$a_1 \mathbf{x}_1 + a_2 \mathbf{x}_2 + \dots + a_k \mathbf{x}_k + \dots + a_p \mathbf{x}_p = \mathbf{0}_{(n \times 1)} (n \times 1)$$

[3, p. 83]*.*

If one of the vectors, for example, \mathbf{x}_k , is $\mathbf{0}_{(n \times 1)}$, the set is linearly dependent

(Let a_k be the only nonzero coefficient). Linear dependence implies that at least one vector in the set can be written as a linear combination of the other vectors. Vectors of the same dimension that are not linearly dependent are said to be **linearly independent** [3, pp. 53, 83].

Definition 2.1.10 (Basis). *Any set of n linearly independent vectors is called a* **basis** *for the vector space of all n – tuples of real numbers* [3, p. 84].

Result 2.1.1. *Every vector can be expressed as a unique linear combination of a fixed basis* [3, p. 84].

Definition 2.1.11 (Inner Product). *The* inner (or dot) product of two vectors $\mathbf{x}_{(n \times 1)}$

and $\mathbf{y}_{(n \times 1)}$ with the same number of entries is defined as the sum of component

products:

$$\mathbf{x}'_{(1\times n)} \cdot \mathbf{y}_{(n\times 1)} = \begin{bmatrix} x_1, x_2, \dots, x_j, \dots, x_n \end{bmatrix} \cdot \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_j \\ \vdots \\ y_n \end{bmatrix} = x_1 y_1 + x_2 y_2 + \dots + x_j y_j + \dots + x_n y_n$$

or

$$\mathbf{y}'_{(1\times n)} \cdot \mathbf{x}_{(n\times 1)} = \begin{bmatrix} y_1, y_2, \dots, y_j, \dots, y_n \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_j \\ \vdots \\ x_n \end{bmatrix} = y_1 x_1 + y_2 x_2 + \dots + y_j x_j + \dots + y_n x_n$$

[3, pp. 52, 85]*.*

Hence,

$$\mathbf{x}'_{(1\times n)} \cdot \mathbf{y}_{(n\times 1)} = \mathbf{y}'_{(1\times n)} \cdot \mathbf{x}_{(n\times 1)}.$$

Definition 2.1.12 (Length of a Vector). *A vector has both direction and length. The* **length of a vector** $\underset{(n\times 1)}{\mathbf{x}}$ *of n elements emanating from the origin* $\underset{(n\times 1)}{\mathbf{0}}$ *is given by the Pythagorean formula:*

$$length of \underbrace{\mathbf{x}}_{(n \times 1)} = \underbrace{L_{\mathbf{x}}}_{(1 \times 1)} = \sqrt{\underbrace{\mathbf{x}'_{(1 \times n)} \cdot \mathbf{x}}_{(1 \times n)} \cdot \underbrace{\mathbf{x}'_{(1 \times n)} \cdot \mathbf{x}}_{(1 \times n)} = \sqrt{\begin{bmatrix} x_1, x_2, \dots, x_j, \dots, x_n \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_j \\ \vdots \\ x_n \\ (n \times 1)} = \sqrt{x_1^2 + x_2^2 + \dots + x_j^2 + \dots + x_n^2}$$

[3, p. 84]*.*

Multiplication by *c* does not change the direction of the vector $\underset{(n \times 1)}{\mathbf{x}}$ if c > 0. However, a negative value of *c* creates a vector with a direction opposite that of $\underset{(n \times 1)}{\mathbf{x}}$. From $L_{c\mathbf{x}} = |c|L_{\mathbf{x}}$ it is clear that $\underset{(n \times 1)}{\mathbf{x}}$ is expanded if |c| > 1 and contracted if 0 < |c| < 1. Choosing $c = L_{\mathbf{x}}^{-1}$, we obtain the *unit vector* $L_{\mathbf{x}}^{-1}\mathbf{x}$, which has length 1 and lies in the direction of $\underset{(n \times 1)}{\mathbf{x}}$ [3, p. 51]. **Definition 2.1.13** (Angle). *The* **angle** θ *between two vectors* $\underset{(n \times 1)}{\mathbf{x}}$ *and* $\underset{(n \times 1)}{\mathbf{y}}$ *in a plane,*

both having n entries, is defined from

$$\cos(\theta) = \frac{\underset{(1\times n)}{\mathbf{x}'} \cdot \mathbf{y}}{L_{\mathbf{x}}L_{\mathbf{y}}} = \frac{\underset{(1\times n)}{\mathbf{x}'} \cdot \mathbf{y}}{\sqrt{\underset{(1\times n)}{\mathbf{x}'} \cdot \underbrace{\mathbf{x}}} \sqrt{\underset{(1\times n)}{\mathbf{y}'} \cdot \underbrace{\mathbf{y}'}} \sqrt{\underset{(1\times n)}{\mathbf{y}'} \cdot \underbrace{\mathbf{y}'}}$$

0ľ

$$\cos(\theta) = \frac{\mathbf{y}' \cdot \mathbf{x}}{L_{\mathbf{x}}L_{\mathbf{y}}} = \frac{\mathbf{y}' \cdot \mathbf{x}}{\sqrt{(1 \times n)} \cdot (n \times 1)} \frac{\mathbf{y}' \cdot \mathbf{x}}{\sqrt{(1 \times n)} \sqrt{\mathbf{y}' \cdot \mathbf{y}}} \frac{\mathbf{y}' \cdot \mathbf{y}}{\sqrt{(1 \times n)} \sqrt{\mathbf{y}' \cdot \mathbf{y}}}$$

[3, pp. 52-53, 85].

Definition 2.1.14 (Perpendicular). When the angle between two vectors $\mathbf{x}_{(n \times 1)}$, $\mathbf{y}_{(n \times 1)}$ is

 $\theta = 90^{\circ} \text{ or } \theta = 270^{\circ}, \text{ we say that } \underset{(n \times 1)}{\mathbf{x}} \text{ and } \underset{(n \times 1)}{\mathbf{y}} \text{ are perpendicular (orthogonal).}$

Since $\cos(\theta) = 0$ only if $\theta = 90^{\circ}$ or $\theta = 270^{\circ}$, the condition becomes

$$\underset{(n\times 1)}{\mathbf{x}} and \underset{(n\times 1)}{\mathbf{y}} are perpendicular if \underset{(1\times n)}{\mathbf{x}'} \cdot \underset{(n\times 1)}{\mathbf{y}} = \underset{(1\times n)}{\mathbf{y}'} \cdot \underset{(n\times 1)}{\mathbf{x}} = 0$$

We write $\underset{(n \times 1)}{\mathbf{x}} \perp \underset{(n \times 1)}{\mathbf{y}}$ [3, pp. 53, 86].

Result 2.1.2.

(a) ^z_(n×1) is perpendicular to every vector if and only if ^z_(n×1) = ⁰_(n×1).
(b) If ^z_(n×1) is perpendicular to each vector x₁, x₂, ..., x_k, ..., x_p then ^z_(n×1) is perpendicular to the span(x₁, x₂, ..., x_k, ..., x_p).
(c) Mutually perpendicular vectors are linearly independent.
[3, p. 86].

2.2 Matrices

Definition 2.2.1 (Matrix). $An \times p$ dimensional array $\underset{(n \times p)}{\mathbf{A}}$ of elements with n rows and p columns is called a matrix, and in general, is denoted by a boldfaced,

uppercase letter. It is written as

$$\mathbf{A}_{(n\times p)} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1k} & \cdots & a_{1p} \\ a_{21} & a_{22} & \cdots & a_{2k} & \cdots & a_{2p} \\ \vdots & \vdots & & \vdots & & \vdots \\ a_{j1} & a_{j2} & \cdots & a_{jk} & \cdots & a_{jp} \\ \vdots & \vdots & & \vdots & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nk} & \cdots & a_{np} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1p} \\ a_{21} & a_{22} & \cdots & a_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nk} & \cdots & a_{np} \end{bmatrix}$$

j = 1, 2, ..., n, k = 1, 2, ..., p. Or more compactly as

$$\mathbf{A}_{(n \times p)} = \{a_{jk}\}$$

where the index j refers to the row and the index k refers to the column.

In our work, the matrix elements will be in \mathbb{R} or functions taking on values in \mathbb{R} [3, pp. 54, 87-88].

Definition 2.2.2 (Matrix Transpose). $Ap \times n$ dimensional array $\mathbf{A}'_{(p \times n)}$ of elements

with p rows and n columns is called a matrix transpose,

$$\mathbf{A}'_{(p \times n)} = \begin{bmatrix} a_{11} & a_{21} & \cdots & a_{j1} & \cdots & a_{n1} \\ a_{12} & a_{22} & \cdots & a_{j2} & \cdots & a_{n2} \\ \vdots & \vdots & & \vdots & & \vdots \\ a_{1k} & a_{2k} & \cdots & a_{jk} & \cdots & a_{nk} \\ \vdots & \vdots & & \vdots & & \vdots \\ a_{1p} & a_{2p} & \cdots & a_{jp} & \cdots & a_{np} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{21} & \cdots & a_{n1} \\ a_{12} & a_{22} & \cdots & a_{n2} \\ \vdots & \vdots & \ddots & \vdots \\ a_{1p} & a_{2p} & \cdots & a_{jp} & \cdots & a_{np} \end{bmatrix}$$

for j = 1, 2, ..., n, k = 1, 2, ..., p.

The transpose operation $\mathbf{A}'_{(p \times n)}$ of a matrix changes the columns into rows, so that the first column of $\mathbf{A}_{(n \times p)}$ becomes the first row of $\mathbf{A}'_{(p \times n)}$, the second column becomes the second row, and so forth [3, p. 55].

Definition 2.2.3 (Matrix Addition). *Let the matrices* $\underset{(n \times p)}{\mathbf{A}}$ *and* $\underset{(n \times p)}{\mathbf{B}}$ *both be of*

dimension $n \times p$ with arbitrary elements a_{jk} and b_{jk} , j = 1, 2, ..., n, k = 1, 2, ..., p,

respectively. The sum of the matrices $\underset{(n \times p)}{\mathbf{A}}$ and $\underset{(n \times p)}{\mathbf{B}}$ is an $n \times p$ matrix $\underset{(n \times p)}{\mathbf{C}}$, written

 $\mathbf{C}_{(n \times p)} = \mathbf{A}_{(n \times p)} + \mathbf{B}_{(n \times p)}, \text{ such that an arbitrary element of } \mathbf{C}_{(n \times p)} \text{ is given by}$

$$c_{jk} = a_{jk} + b_{jk}$$
 $j = 1, 2, ..., n$ $k = 1, 2, ..., p$

$$\mathbf{C}_{(n \times p)} = \mathbf{A}_{(n \times p)} + \mathbf{B}_{(n \times p)} = \begin{bmatrix} a_{11} + b_{11} & a_{12} + b_{12} & \cdots & a_{1p} + b_{1p} \\ a_{21} + b_{21} & a_{22} + b_{22} & \cdots & a_{2p} + b_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} + b_{n1} & a_{n2} + b_{n2} & \cdots & a_{np} + b_{np} \end{bmatrix} = \mathbf{A} + \mathbf{B}_{(n \times p)}$$

Note that the addition of matrices is defined only for matrices of the same dimension [3, p. 88].

Definition 2.2.4 (Scalar Multiplication).

Let c be an arbitrary scalar and $\underset{(n \times p)}{\mathbf{A}} = \{a_{jk}\}$. *Then* $\underset{(n \times p)}{c\mathbf{A}} = \underset{(n \times p)}{\mathbf{A}} = \underset{(n \times p)}{\mathbf{A}} = \{b_{jk}\}$, *where* $b_{jk} = ca_{jk} = a_{jk}c$, j = 1, 2, ..., n, k = 1, 2, ..., p. *That is,*

$$\underset{(n \times p)}{c\mathbf{A}} = \underset{(n \times p)}{\mathbf{A}c} = \begin{bmatrix} ca_{11} & ca_{12} & \cdots & ca_{1p} \\ ca_{21} & ca_{22} & \cdots & ca_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ ca_{n1} & ca_{n2} & \cdots & ca_{np} \end{bmatrix} = \begin{bmatrix} a_{11}c & a_{12}c & \cdots & a_{1p}c \\ a_{21}c & a_{22}c & \cdots & a_{2p}c \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1}c & a_{n2}c & \cdots & a_{np}c \end{bmatrix} = \underset{(n \times p)}{\mathbf{B}}$$

Multiplication of a matrix by a scalar produces a new matrix whose elements are the elements of the original matrix, each multiplied by the scalar [3, pp. 55, 89]. **Definition 2.2.5** (Matrix Multiplication). *The product* $\underset{(n \times m)}{\mathbf{A}} \cdot \underset{(m \times p)}{\mathbf{B}} of an n \times m$ matrix $\underset{(n \times m)}{\mathbf{A}} = \{a_{jk}\}$ and an $m \times p$ matrix $\underset{(m \times p)}{\mathbf{B}} = \{b_{jk}\}$ is the $n \times p$ matrix $\underset{(n \times p)}{\mathbf{A}} = \{a_{jk}\}$ and an $m \times p$ matrix $\underset{(m \times p)}{\mathbf{B}} = \{b_{jk}\}$ is the $n \times p$ matrix $\underset{(n \times p)}{\mathbf{C}} = \underset{(n \times m)}{\mathbf{A}} \cdot \underset{(m \times p)}{\mathbf{B}} = \{c_{jk}\}$ whose elements in the jth row and kth column is the inner product of the jth row of $\underset{(n \times m)}{\mathbf{A}}$ and the kth column of $\underset{(m \times p)}{\mathbf{B}}$ or $c_{jk} = (j, k)$ element of $\underset{(n \times m)}{\mathbf{A}} \cdot \underset{(m \times p)}{\mathbf{B}} = a_{j1}b_{1k} + a_{j2}b_{2k} + \dots + a_{jm}b_{mk} = \sum_{l=1}^{m} a_{jl}b_{lk}$ for $j = 1, 2, \dots, n, k = 1, 2, \dots, p$ [3, pp. 55-56, 90]. More generally, the matrix product is given by

$$\begin{split} \mathbf{A}_{(n\times m)} & \cdot \mathbf{B}_{(n\times p)} \\ &= \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \vdots & \vdots & & \vdots \\ a_{j1} & a_{j2} & \cdots & a_{jm} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nm} \end{bmatrix} \cdot \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1k} & \cdots & b_{1p} \\ b_{21} & b_{22} & \cdots & b_{2k} & \cdots & b_{2p} \\ \vdots & \vdots & & \vdots & & \vdots \\ b_{m1} & b_{m2} & \cdots & b_{mk} & \cdots & b_{mp} \end{bmatrix} \\ &= \begin{bmatrix} \text{Column 1} & \text{Column 2} & \cdots & \text{Column } k & \cdots & \text{Column } p \\ \text{row 1} & \sum_{l=1}^{m} a_{1l}b_{l1} & \sum_{l=1}^{m} a_{1l}b_{l2} & \cdots & \sum_{l=1}^{m} a_{1l}b_{lk} & \cdots & \sum_{l=1}^{m} a_{1l}b_{lp} \\ \text{row 2} & \sum_{l=1}^{m} a_{2l}b_{l1} & \sum_{l=1}^{m} a_{2l}b_{l2} & \cdots & \sum_{l=1}^{m} a_{2l}b_{lk} & \cdots & \sum_{l=1}^{m} a_{2l}b_{lp} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \text{row } j & \sum_{l=1}^{m} a_{jl}b_{l1} & \sum_{l=1}^{m} a_{jl}b_{l2} & \cdots & \sum_{l=1}^{m} a_{jl}b_{lk} & \cdots & \sum_{l=1}^{m} a_{jl}b_{lp} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \text{row } n & \sum_{l=1}^{m} a_{nl}b_{l1} & \sum_{l=1}^{m} a_{nl}b_{l2} & \cdots & \sum_{l=1}^{m} a_{nl}b_{lp} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \text{row } n & \sum_{l=1}^{m} a_{nl}b_{l1} & \sum_{l=1}^{m} a_{nl}b_{l2} & \cdots & \sum_{l=1}^{m} a_{nl}b_{lk} & \cdots & \sum_{l=1}^{m} a_{nl}b_{lp} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \text{row } n & \sum_{l=1}^{m} a_{nl}b_{l1} & \sum_{l=1}^{m} a_{nl}b_{l2} & \cdots & \sum_{l=1}^{m} a_{nl}b_{lk} & \cdots & \sum_{l=1}^{m} a_{nl}b_{lp} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \text{row } n & \sum_{l=1}^{m} a_{nl}b_{l1} & \sum_{l=1}^{m} a_{nl}b_{l2} & \cdots & \sum_{l=1}^{m} a_{nl}b_{lk} & \cdots & \sum_{l=1}^{m} a_{nl}b_{lp} \end{bmatrix} \\ = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1k} & \cdots & c_{1p} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ c_{j1} & c_{j2} & \cdots & c_{nk} & \cdots & c_{np} \end{bmatrix} = c_{(n\times p)} \end{bmatrix}$$

Note that for the product $\underset{(n \times m)}{\mathbf{A}} \cdot \underset{(m \times p)}{\mathbf{B}}$ to be defined, the column dimension of

 $\begin{array}{l} \mathbf{A} \\ {}_{(n \times m)} \text{ must equal the row dimension of } \mathbf{B} \\ {}_{(m \times p)} \cdot \mathbf{B} \\ {}_{(m \times m)} \cdot \mathbf{B} \\ {}_{(m \times p)} \text{ equals the row dimension of } \mathbf{A} \\ {}_{(n \times m)} \cdot \mathbf{A} \\ {}_{(m \times p)} \text{ equals the column dimension of } \mathbf{B} \\ {}_{(m \times p)} [3, \text{ pp. 55-56, 90}]. \end{array}$

Result 2.2.1. For all matrices A, B, and C (of equal dimension) and scalars c and d,

the following holds:

$$(\mathbf{a}) \mathbf{A} - \mathbf{B} = \mathbf{A} + (-1)\mathbf{B}$$

(b)
$$(A + B) + C = A + (B + C)$$

- $(\mathbf{c})\mathbf{A} + \mathbf{B} = \mathbf{B} + \mathbf{A}$
- $(\mathbf{d}) c(\mathbf{A} + \mathbf{B}) = c\mathbf{A} + c\mathbf{B}$
- $(\mathbf{e}) (c+d)\mathbf{A} = c\mathbf{A} + d\mathbf{A}$
- $(\mathbf{f}) (\mathbf{A} + \mathbf{B})' = \mathbf{A}' + \mathbf{B}'$
- $(\mathbf{g}) (cd)\mathbf{A} = c(d\mathbf{A})$
- (**h**) (cA)' = cA' (Note c' = c)
- [3, p. 89]*.*

Result 2.2.2. For all matrices A, B, and C (of dimensions such that the indicated

products are defined) and a scalar c,

- (a) c(AB) = (cA)B
- $(\mathbf{b}) \mathbf{A}(\mathbf{B}\mathbf{C}) = (\mathbf{A}\mathbf{B})\mathbf{C}$
- (c) A(B + C) = AB + AC
- $(\mathbf{d}) \ (\mathbf{B} + \mathbf{C})\mathbf{A} = \mathbf{B}\mathbf{A} + \mathbf{C}\mathbf{A}$
- $(\mathbf{e}) (\mathbf{A}\mathbf{B})' = \mathbf{B}'\mathbf{A}'$
- [3, p. 91],

Definition 2.2.6 (Zero Matrix). $\underset{(n \times p)}{\mathbf{0}}$ matrix is a rectangular array of 0's, of arbitrary

dimension n × p. It is written as

$$\mathbf{0}_{(n\times p)} = \begin{bmatrix} 0_{11} & 0_{12} & \cdots & 0_{1p} \\ 0_{21} & 0_{22} & \cdots & 0_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ 0_{n1} & 0_{n2} & \cdots & 0_{np} \end{bmatrix} = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix}$$

Note that the notation for the $\mathbf{0}_{(p \times 1)}$ vector is similar; but the dimension makes

the context clear.

Definition 2.2.7 (Square Matrix). *If an arbitrary matrix* $\underset{(p \times p)}{\mathbf{A}}$ *has the same number of rows and columns, say dimension* $p \times p$ *, then* $\underset{(p \times p)}{\mathbf{A}}$ *is called a* **square matrix**. *It is*

written as

$$\mathbf{A}_{(p \times p)} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1p} \\ a_{21} & a_{22} & \cdots & a_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ a_{p1} & a_{p2} & \cdots & a_{pp} \end{bmatrix} = \{a_{ik}\}$$

for i = 1,2, ..., *p rows and k* = 1,2, ..., *p columns* [3, p. 90].

Definition 2.2.8 (Symmetrix Matrix). Let $\underset{(p \times p)}{\mathbf{A}} = \{a_{ik}\}$ be a $p \times p$ (square) matrix.

Then $\underset{(p \times p)}{\mathbf{A}}$ is said to be a symmetric matrix if $\underset{(p \times p)}{\mathbf{A}} = \underset{(p \times p)}{\mathbf{A}'}$. That is, $\underset{(p \times p)}{\mathbf{A}}$ is symmetric

 $ifa_{ik} = a_{ki} \forall i = 1, 2, ..., p, k = 1, 2, ..., p [3, p. 90].$

Definition 2.2.9 (Determinant). *The* determinant *of a square* $p \times p$ *matrix* $\underset{(p \times p)}{\mathbf{A}}$,

denoted by |A|, is the scalar

$$|\mathbf{A}| = a_{11}$$
 if $p = 1$
 $|\mathbf{A}| = \sum_{k=1}^{p} a_{1k} |\mathbf{A}_{1k}| (-1)^{1+k}$ if $p > 1$

where \mathbf{A}_{1k} is the $(p-1) \times (p-1)$ matrix obtained by deleting the first row and kth column of $\mathbf{A}_{(p \times p)}$. Also,

$$|\mathbf{A}| = \sum_{k=1}^{p} a_{ik} |\mathbf{A}_{ik}| (-1)^{i+k} \quad \text{if } p > 1$$
with the ith row in place of the first row [3, p. 93].

Definition 2.2.10 (Identity Matrix). *The* $p \times p$ **identity matrix**, *denoted by* $\prod_{(p \times p)}$, *is the*

square matrix with ones on the main (*NW* – *SE*) *diagonal and zeros elsewhere. It is written as*

$$\mathbf{I}_{(p \times p)} = \begin{bmatrix} 1_{11} & 0 & \cdots & 0 \\ 0 & 1_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1_{pp} \end{bmatrix} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1_{pp} \end{bmatrix}$$

[3, p. 90]*.*

The matrix $\prod_{(p \times p)} acts$ like 1 in ordinary multiplication $(1 \cdot a = a \cdot 1 = a)$

$$\mathbf{I}_{(p \times p)} \cdot \mathbf{A}_{(p \times p)} = \mathbf{A}_{(p \times p)} \cdot \mathbf{I}_{(p \times p)} = \mathbf{A}_{(p \times p)} \text{ for any } \mathbf{A}_{(p \times p)}$$

so it is called the identity matrix [3, p. 58].

Remark 2.2.1. There are several important differences between the algebra of matrices and the algebra of real numbers. Two of these differences are as follows:

1. Matrix multiplication is, in general, not commutative. That is, in general,

$$\mathbf{A}_{(p \times p)} \cdot \mathbf{B}_{(p \times p)} \neq \mathbf{B}_{(p \times p)} \cdot \mathbf{A}_{(p \times p)}$$

2. Let $\underset{(n \times p)}{\mathbf{0}}$ denote the zero matrix, that is, the matrix with zero for every element. In the algebra of real numbers, if the product of two numbers, *ab*, is zero, then a = 0 or b = 0. In matrix algebra, however, the product of two nonzero matrices may be the zero matrix. Hence,

$$\mathbf{A}_{(n \times m)} \cdot \mathbf{B}_{(m \times p)} = \mathbf{0}_{(n \times p)}$$

does **not** imply that $\mathbf{A}_{(n \times m)} = \mathbf{0}_{(n \times m)}$ or $\mathbf{B}_{(m \times p)} = \mathbf{0}_{(m \times p)}$. It is true, however, that if either

$$\mathbf{A}_{(n \times m)} = \mathbf{0}_{(n \times m)}$$
 or $\mathbf{B}_{(m \times p)} = \mathbf{0}_{(m \times p)}$, then $\mathbf{A}_{(n \times m)} \cdot \mathbf{B}_{(m \times p)} = \mathbf{0}_{(n \times p)}$

[3, pp. 58, 92].

Definition 2.2.11 (Row Rank and Column Rank). *The* row rank of a matrix is the maximum number of linearly independent rows, considered as vectors. The column rank of a matrix is the rank of its set of columns, considered as vectors [3, p. 94].
Result 2.2.3 (Rank of a Matrix). *The row rank and the column rank of a matrix are equal. Thus, the* rank of a matrix is either the row rank or the column rank [3, p. 94].

Definition 2.2.12 (Nonsingular). A square matrix $\mathbf{A}_{(p \times p)}$ is nonsingular if

$$\mathbf{A}_{(p \times p)} \cdot \mathbf{x}_{(p \times 1)} = \mathbf{0}_{(p \times 1)}$$

implies

$$\mathbf{x}_{(p\times 1)} = \mathbf{0}_{(p\times 1)}.$$

If a matrix fails to be nonsingular, it is called singular. Equivalently, a square matrix is nonsingular if its rank is equal to the number of rows (or columns) it has.

Note that
$$\mathbf{A}_{(p \times p)} \cdot \mathbf{x}_{(p \times 1)} = x_1 \mathbf{a}_1 + x_2 \mathbf{a}_2 + \dots + x_k \mathbf{a}_k + \dots + x_p \mathbf{a}_p$$
, where $x_k \mathbf{a}_k$ is $(p \times 1)$ $(p \times 1)$

the *k*th column of $\mathbf{A}_{(p \times p)}$, so that the condition of nonsingularity is just the statement that the columns of $\mathbf{A}_{(p \times p)}$ are linearly independent [3, p. 95].

Definition 2.2.13 (Inverse). Let $\underset{(p \times p)}{\mathbf{A}}$ be a nonsingular square matrix of dimension

 $p \times p$. Then there is a unique $p \times p$ matrix $\underset{(p \times p)}{\mathbf{B}}$ such that

$$\mathbf{A}_{(p \times p)} \cdot \mathbf{B}_{(p \times p)} = \mathbf{B}_{(p \times p)} \cdot \mathbf{A}_{(p \times p)} = \mathbf{I}_{(p \times p)}$$

where $\underset{(p \times p)}{\mathbf{I}}$ is the $p \times p$ identity matrix. Then $\underset{(p \times p)}{\mathbf{B}}$ is called the inverse of $\underset{(p \times p)}{\mathbf{A}}$ and

is denoted by $\mathbf{A}^{-1}_{(p \times p)}$ [3, p. 95].

Result 2.2.4. For a square matrix $\underset{(p \times p)}{\mathbf{A}}$ of dimension $p \times p$, the following are

equivalent:

(a)
$$\underset{(p\times p)}{\mathbf{A}} \cdot \underset{(p\times 1)}{\mathbf{x}} = \underset{(p\times 1)}{\mathbf{0}} \text{ implies } \underset{(p\times 1)}{\mathbf{x}} = \underset{(p\times 1)}{\mathbf{0}} \left(\underset{(p\times p)}{\mathbf{A}} \text{ is nonsingular} \right).$$

(b) $|\mathbf{A}| \neq 0$ where $(| \cdot |$ denotes the determinant operator).

(c) There exists a matrix $\underset{(p \times p)}{\mathbf{A}^{-1}}$ such that $\underset{(p \times p)}{\mathbf{A}} \cdot \underset{(p \times p)}{\mathbf{A}^{-1}} = \underset{(p \times p)}{\mathbf{A}^{-1}} \cdot \underset{(p \times p)}{\mathbf{A}} = \underset{(p \times p)}{\mathbf{I}}$.

[3, p. 96]*.*

Result 2.2.5. Let $\underset{(p \times p)}{\mathbf{A}}$ and $\underset{(p \times p)}{\mathbf{B}}$ be $p \times p$ square matrices, and let the indicated

inverses exist. Then the following hold:

(a) $(\mathbf{A}^{-1})' = (\mathbf{A}')^{-1}$ $(p \times p) = (p \times p)^{-1}$ (b) $(\mathbf{AB})^{-1} = \mathbf{B}^{-1} \cdot \mathbf{A}^{-1}$ $(p \times p) \cdot (p \times p)$

[3, p. 96].

Definition 2.2.14 (Trace). Let $\underset{(p \times p)}{\mathbf{A}} = \{a_{ik}\}$ be a $p \times p$ square matrix. The trace of the

matrix $\underset{(p \times p)}{\mathbf{A}}$, written tr(A) is the sum of the diagonal elements; that is,

$$\operatorname{tr}(\mathbf{A}) = \sum_{i=1}^{p} a_{ii}$$

[3, p. 96]*.*

Result 2.2.6. Let $\underset{(p \times p)}{\mathbf{A}}$ and $\underset{(p \times p)}{\mathbf{B}}$ be $p \times p$ square matrices, $\underset{(p \times p)}{\mathbf{B}^{-1}}$ exist, and c be a

scalar.

(a) $\operatorname{tr}(c\mathbf{A}) = c\operatorname{tr}(\mathbf{A})$

- $(\mathbf{b}) \operatorname{tr}(\mathbf{AB}) = \operatorname{tr}(\mathbf{BA})$
- (c) $tr(B^{-1}AB) = tr(A)$

[3, p. 97]*.*

Definition 2.2.15 (Orthogonal). A square matrix $\underset{(p \times p)}{\mathbf{A}}$ is said to be **orthogonal** if its rows

$$\mathbf{a}_{r} = \begin{bmatrix} a_{r1} \\ a_{r2} \\ \vdots \\ a_{rp} \end{bmatrix}$$

for r = 1, 2, ..., p, considered as vectors, are mutually perpendicular,

$$\mathbf{a}'_{r} \cdot \mathbf{a}_{s} = 0 \text{ for } r \neq s$$
$$_{(1 \times p)} \cdot (p \times 1)$$

and have unit lengths

$$\mathbf{a}_r' \cdot \mathbf{a}_r = 1$$

that is,

$$\mathbf{A}_{(p\times p)} \cdot \mathbf{A}'_{(p\times p)} = \mathbf{I}_{(p\times p)'}$$

and its columns

$$\mathbf{a}_{i}_{(p\times 1)} = \begin{bmatrix} a_{1i} \\ a_{2i} \\ \vdots \\ a_{pi} \end{bmatrix}$$

for *i* = 1,2, ..., *p*, *considered as vectors, are mutually perpendicular*,

$$\mathbf{a}'_{i} \cdot \mathbf{a}_{k} = 0 \text{ for } i \neq k$$

$$(1 \times p) \quad (p \times 1)$$

and have unit lengths

$$\mathbf{a}_i' \cdot \mathbf{a}_i = 1$$

$$(1 \times p) \quad (p \times 1)$$

that is,

$$\mathbf{A}'_{(p\times p)} \cdot \mathbf{A}_{(p\times p)} = \mathbf{I}_{(p\times p)}$$

[3, pp. 59, 97]*.*

Result 2.2.7. A square matrix $\underset{(p \times p)}{\mathbf{A}}$ is orthogonal if and only if $\underset{(p \times p)}{\mathbf{A}^{-1}} = \underset{(p \times p)}{\mathbf{A}'}$. For an orthogonal matrix, $\underset{(p \times p)}{\mathbf{A}} \cdot \underset{(p \times p)}{\mathbf{A}'} = \underset{(p \times p)}{\mathbf{A}'} \cdot \underset{(p \times p)}{\mathbf{A}} = \underset{(p \times p)}{\mathbf{I}}$, so, the rows and columns are also mutually perpendicular [3, pp. 59, 97].

Definition 2.2.16 (Eigenvalues). Let $\underset{(p \times p)}{\mathbf{A}}$ be a $p \times p$ square matrix and $\underset{(p \times p)}{\mathbf{I}}$ be the $p \times p$ identity matrix. Then the scalars $\lambda_1, \lambda_2, ..., \lambda_p$ satisfying the polynomial equation $|\mathbf{A} - \lambda \mathbf{I}| = 0$ are called the **eigenvalues** (or characteristic roots) of a matrix $\underset{(p \times p)}{\mathbf{A}}$. The equation $|\mathbf{A} - \lambda \mathbf{I}| = 0$ (as a function of λ) is called the **characteristic**

equation [3, p. 97].

Definition 2.2.17 (Eigenvector). Let $\underset{(p \times p)}{\mathbf{A}}$ be a square matrix of dimension $p \times p$ and let λ be an eigenvalue of $\underset{(p \times p)}{\mathbf{A}}$. If $\underset{(p \times 1)}{\mathbf{x}}$ is a nonzero vector $\begin{pmatrix} \mathbf{x} \\ (p \times 1) \end{pmatrix} \neq \underset{(p \times 1)}{\mathbf{0}} \end{pmatrix}$ such that $\underset{(p \times p)}{\mathbf{A}} \cdot \underset{(p \times 1)}{\mathbf{x}} = \underset{(p \times 1)}{\lambda} \cdot \underset{(p \times 1)}{\mathbf{x}}$

then $\mathbf{x}_{(p \times 1)}$ is said to be an **eigenvector** (characteristic vector) of the matrix $\mathbf{A}_{(p \times p)}$ associated with the eigenvalue λ [3, p. 98].

An equivalent condition for λ to be a solution of the eigenvalue-eigenvector equation is $|\mathbf{A} - \lambda \mathbf{I}| = 0$. This follows because the statement that $\underset{(p \times p)}{\mathbf{A}} \cdot \underset{(p \times 1)}{\mathbf{x}} = \underset{(p \times 1)}{\lambda \mathbf{x}}$ for some λ and $\underset{(p \times 1)}{\mathbf{x}} \neq \underset{(p \times 1)}{\mathbf{0}}$ implies that

$$\mathbf{0}_{(p\times 1)} = (\mathbf{A} - \lambda \mathbf{I}) \cdot \mathbf{x}_{(p\times 1)} = x_1 \cdot \operatorname{col}_1(\mathbf{A} - \lambda \mathbf{I}) + \dots + x_p \cdot \operatorname{col}_p(\mathbf{A} - \lambda \mathbf{I})_{(p\times 1)}$$

That is, the columns of $(\mathbf{A} - \lambda \mathbf{I})$ are linearly dependent so, by Result 2.2.4. (b), $(p \times p)$

 $|A - \lambda I| = 0$, as asserted [3, p. 98].
Ordinarily, we normalize $\mathbf{x}_{(p \times 1)}$ so that it has length unity. It is convenient to denote normalized eigenvectors by

$$\mathbf{e}_{(p\times1)} = L_{\mathbf{x}}^{-1} \cdot \mathbf{x}_{(p\times1)} = \frac{\mathbf{x}_{(p\times1)}}{L_{\mathbf{x}}} = \frac{\mathbf{x}_{(p\times1)}}{\sqrt{\mathbf{x}_{(1\times1)}}} = \frac{\mathbf{x}_{(p\times1)}}{\sqrt{\mathbf{x}_{(1\times1)}} \cdot \mathbf{x}_{(n\times1)}}$$

and we do so in what follows [3, pp. 60, 99].

Definition 2.2.18 (Eigenvalue-Eigenvector Pairs). Let $\underset{(p \times p)}{\mathbf{A}}$ be a $p \times p$ square

symmetric matrix. Then $\underset{(p \times p)}{\mathbf{A}}$ has p eigenvalue-eigenvector pairs-namely,

$$\begin{pmatrix} \lambda_1, \mathbf{e}_1 \\ (p \times 1) \end{pmatrix}, \begin{pmatrix} \lambda_2, \mathbf{e}_2 \\ (p \times 1) \end{pmatrix}, \dots, \begin{pmatrix} \lambda_i, \mathbf{e}_i \\ (p \times 1) \end{pmatrix}, \dots, \begin{pmatrix} \lambda_p, \mathbf{e}_p \\ (p \times 1) \end{pmatrix}$$

Let the normalized eigenvectors be the columns of another matrix

$$\mathbf{E}_{(p \times p)} = \begin{bmatrix} e_{11} & e_{12} & \cdots & e_{1p} \\ e_{21} & e_{22} & \cdots & e_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ e_{p1} & e_{p2} & \cdots & e_{pp} \end{bmatrix}$$

where the columns of the $\mathop{\mathbf{E}}_{(p\times p)}$ are mutually perpendicular

$$\mathbf{e}'_{i} \cdot \mathbf{e}_{k}_{(1 \times p)} = 0 \text{ for } i \neq k$$

and have unit lengths

$$\mathbf{e}_i' \cdot \mathbf{e}_i = 1$$

that is,

$$\mathbf{\underline{E}}'_{(p\times p)} \cdot \mathbf{\underline{E}}_{(p\times p)} = \mathbf{\underline{I}}_{(p\times p)}$$

$$\mathbf{e}'_r \cdot \mathbf{e}_s = 0 \text{ for } r \neq s$$

and have unit lengths

$$\mathbf{e}_r' \cdot \mathbf{e}_r = 1$$

that is,

$$\mathop{\mathbf{E}}_{(p\times p)}\cdot\mathop{\mathbf{E}'}_{(p\times p)}=\mathop{\mathbf{I}}_{(p\times p)}$$

Thus, $\underset{(p \times p)}{\mathbf{E}}$ is orthogonal making

$$\mathop{\mathbf{E}}_{(p\times p)}\cdot\mathop{\mathbf{E}'}_{(p\times p)}=\mathop{\mathbf{E}'}_{(p\times p)}\cdot\mathop{\mathbf{E}}_{(p\times p)}=\mathop{\mathbf{I}}_{(p\times p)}$$

and

$$\mathbf{E}^{-1}_{(p \times p)} = \mathbf{E}'_{(p \times p)}$$

Let us demonstrate,

$$\begin{split} \mathbf{E}_{(p \times p)} \cdot \mathbf{E}'_{(p \times p)} \\ &= \begin{bmatrix} e_{11} & e_{12} & \cdots & e_{1p} \\ e_{21} & e_{22} & \cdots & e_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ e_{p1} & e_{p2} & \cdots & e_{pp} \end{bmatrix} \cdot \begin{bmatrix} e_{11} & e_{21} & \cdots & e_{p1} \\ e_{12} & e_{22} & \cdots & e_{p2} \\ \vdots & \vdots & \ddots & \vdots \\ e_{1p} & e_{2p} & \cdots & e_{pp} \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{e}_{1}' \mathbf{e}_{1} = 1 & \mathbf{e}_{1}' \mathbf{e}_{2} = 0 & \cdots & \mathbf{e}_{1}' \mathbf{e}_{p} = 0 \\ \mathbf{e}_{2}' \mathbf{e}_{1} = 0 & \mathbf{e}_{2}' \mathbf{e}_{2} = 1 & \cdots & \mathbf{e}_{2}' \mathbf{e}_{p} = 0 \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{e}_{p}' \mathbf{e}_{1} = 0 & \mathbf{e}_{p}' \mathbf{e}_{2} = 0 & \cdots & \mathbf{e}_{p}' \mathbf{e}_{p} = 1 \end{bmatrix}, \text{ (rows perpendicular} \end{split}$$

$$= \begin{bmatrix} 1_{1} & 0 & \cdots & 0 \\ 0 & 1_{2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1_{p} \\ & & (p \times p) \end{bmatrix}$$
$$= \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}$$
$$= \begin{bmatrix} \mathbf{I} \\ & (p \times p) \end{bmatrix}$$

and

$$\begin{split} \mathbf{E}'_{(p \times p)} \cdot \mathbf{E}_{(p \times p)} \\ &= \begin{bmatrix} e_{11} & e_{21} & \cdots & e_{p1} \\ e_{12} & e_{22} & \cdots & e_{p2} \\ \vdots & \vdots & \ddots & \vdots \\ e_{1p} & e_{2p} & \cdots & e_{pp} \end{bmatrix} \begin{bmatrix} e_{11} & e_{12} & \cdots & e_{1p} \\ e_{21} & e_{22} & \cdots & e_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ e_{p1} & e_{p2} & \cdots & e_{pp} \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{e}'_{1}\mathbf{e}_{1} = 1 & \mathbf{e}'_{1}\mathbf{e}_{2} = 0 & \cdots & \mathbf{e}'_{1}\mathbf{e}_{p} = 0 \\ \mathbf{e}'_{2}\mathbf{e}_{1} = 0 & \mathbf{e}'_{2}\mathbf{e}_{2} = 1 & \cdots & \mathbf{e}'_{2}\mathbf{e}_{p} = 0 \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{e}'_{p}\mathbf{e}_{1} = 0 & \mathbf{e}'_{p}\mathbf{e}_{2} = 0 & \cdots & \mathbf{e}'_{p}\mathbf{e}_{p} = 1 \end{bmatrix}, \text{ (columns perpendicular)} \\ &= \begin{bmatrix} 1_{1} & 0 & \cdots & 0 \\ 0 & 1_{2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1_{p} \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1_{p} \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1_{p} \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1_{p} \end{bmatrix}$$

Note that the eigenvectors are unique unless two or more eigenvalues are

equal. Clearly, \mathbf{e}_1 , \mathbf{e}_2 , ..., \mathbf{e}_i , ..., \mathbf{e}_p are the (normalized) solutions of the $(p \times 1)$ $(p \times 1)$ $(p \times 1)$

equations $\underset{(p \times p)}{\mathbf{A}} \cdot \underset{(p \times 1)}{\mathbf{e}_i} = \lambda_i \cdot \underset{(p \times 1)}{\mathbf{e}_i}$ for i = 1, 2, ..., p [3, pp. 60-61, 65].

Definition 2.2.19 (Quadratic Form). A quadratic form $Q(\mathbf{x})$ in the *p* variables (1×1)

$$x_1, x_2, \dots, x_p \text{ is } Q(\mathbf{x}) = \mathbf{x}'_{(1 \times p)} \cdot \mathbf{A}_{(p \times p)} \cdot \mathbf{x}_{(p \times 1)}, \text{ where } \mathbf{x}'_{(1 \times p)} = [x_1, x_2, \dots, x_p] \text{ and } \mathbf{A}_{(p \times p)} \text{ is a}$$

 $p \times p$ symmetric matrix.

Note that a quadratic form can be written as

$$Q(\mathbf{x})_{(1\times 1)} = \sum_{i=1}^{p} \sum_{k=1}^{p} a_{ik} x_i x_k$$

Because $\mathbf{x}'_{(1 \times p)} \cdot \mathbf{A}_{(p \times p)} \cdot \mathbf{x}_{(p \times 1)}$ has only squared terms x_i^2 and product terms $x_i x_k$, it is

called a quadratic form [3, pp. 62, 99].

If
$$\exists \mathbf{x}_{(p \times 1)} \neq \mathbf{0}_{(p \times 1)}$$
 and a $p \times p$ symmetric matrix $\mathbf{A}_{(p \times p)}$ where
$$\mathbf{0}_{(1 \times 1)} = \mathbf{x}'_{(1 \times p)} \cdot \mathbf{A}_{(p \times p)} \cdot \mathbf{x}_{(p \times 1)}$$

then the matrix $\mathbf{A}_{(p \times p)}$ and the quadratic form are said to be **positive semi-definite**. If

$$\underset{(1\times1)}{0} < \underset{(1\timesp)}{\mathbf{x}'} \cdot \underset{(p\timesp)}{\mathbf{A}} \cdot \underset{(p\times1)}{\mathbf{x}}$$

 $\forall \mathbf{x}_{(p \times 1)} \neq \mathbf{0}_{(p \times 1)}$ then the $p \times p$ symmetric matrix $\mathbf{A}_{(p \times p)}$ and the quadratic form are

said to be **positive definite** [3, p. 62].

In addition, when $\underset{(p \times p)}{\mathbf{A}}$ is positive definite the quadratic form can be interpreted as a squared distance [3, p. 64]. If the quadratic form and the matrix $\underset{(p \times p)}{\mathbf{A}}$ are positive semi-definite or positive definite they are said to be nonnegative definite [3, p. 62].

Results involving quadratic forms and symmetric matrices are, in many cases, a direct consequence of an expansion for symmetric matrices known as the **spectral decomposition**. That is, any symmetric square matrix can be reconstructed from its eigenvalues and eigenvectors. The particular expression reveals the relative importance of each pair according to the relative size of the eigenvalue and the direction of the eigenvector [3, pp. 61, 99].

Result 2.2.8 (Spectral Decomposition). *The* **Spectral Decomposition**. *Let* $\underset{(p \times p)}{\mathbf{A}}$ *be a* $p \times p$ symmetric matrix. Then $\underset{(p \times p)}{\mathbf{A}}$ *can be expressed in terms of its* p *eigenvalue-*

eigenvector pairs $\left(\lambda_{i}, \frac{\mathbf{e}_{i}}{(p \times 1)}\right)$ as $\mathbf{A}_{i} = \sum_{j=1}^{p} \lambda_{i} \cdot \mathbf{e}_{i} \cdot \mathbf{e}_{j}' = \lambda_{i} \cdot \mathbf{e}_{i} \cdot \mathbf{e}_{j}' + \dots + \lambda_{i}$

$$\mathbf{A}_{(p \times p)} = \sum_{i=1}^{n} \lambda_i \cdot \mathbf{e}_i \cdot \mathbf{e}_i = \lambda_1 \cdot \mathbf{e}_1 \cdot \mathbf{e}_1 \cdot \mathbf{e}_1 + \dots + \lambda_p \cdot \mathbf{e}_p \cdot \mathbf{e}$$

where $\lambda_1, \lambda_1, \dots, \lambda_p$ are the eigenvalues of $\underset{(p \times p)}{\mathbf{A}}$ and $\underset{(p \times 1)}{\mathbf{e}_1}$, $\underset{(p \times 1)}{\mathbf{e}_2}$, ..., $\underset{(p \times 1)}{\mathbf{e}_p}$ are the

associated normalized eigenvectors [3, pp. 61, 100].

Using the spectral decomposition, we can easily show that a $p \times p$ symmetric matrix $\underset{(p \times p)}{\mathbf{A}}$ is a positive definite matrix if and only if every eigenvalue of $\underset{(p \times p)}{\mathbf{A}}$ is positive $[\lambda_i > 0 \forall i]$. Similarly, $\underset{(p \times p)}{\mathbf{A}}$ is a positive semi-definite matrix if and only if $\exists \lambda_i = 0$ and the other eigenvalues are positive [5, pp. 212, 549].

The spectral decomposition allows us to express the inverse of a square matrix in terms of its eigenvalues and eigenvectors, and this leads to a useful square-root matrix.

Let $\underset{(p \times p)}{\mathbf{A}}$ be a $p \times p$ positive definite matrix with the spectral decomposition

$$\mathbf{A}_{(p \times p)} = \sum_{i=1}^{p} \lambda_i \cdot \mathbf{e}_i \cdot \mathbf{e}'_i \cdot \mathbf{e}'_i.$$

Let $\underset{(p \times p)}{\mathbf{E}}$ be a $p \times p$ orthogonal matrix with columns equal to the normalized

eigenvectors of $\mathbf{A}_{(p \times p)}$,

$$\mathbf{E}_{(p \times p)} = \begin{bmatrix} e_{11} & e_{12} & \cdots & e_{1p} \\ e_{21} & e_{22} & \cdots & e_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ e_{p1} & e_{p2} & \cdots & e_{pp} \end{bmatrix}$$

let and $\bigwedge_{(p \times p)}$ be the the diagonal matrix of eigenvalues of $\bigwedge_{(p \times p)}$

$$\mathbf{\Lambda}_{(p \times p)} = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0\\ 0 & \lambda_2 & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots\\ 0 & 0 & \cdots & \lambda_p \end{bmatrix}$$
$$(p \times p)$$

with inverse,

$$\mathbf{\Lambda}_{(p \times p)}^{-1} = \begin{bmatrix} \frac{1}{\lambda_1} & 0 & \cdots & 0\\ 0 & \frac{1}{\lambda_2} & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots\\ 0 & 0 & \cdots & \frac{1}{\lambda_p} \end{bmatrix}_{(p \times p)}$$

Then

$$\mathbf{A}_{(p \times p)} = \sum_{i=1}^{p} \lambda_i \cdot \mathbf{e}_i (p \times 1) \cdot \mathbf{e}_i' = \mathbf{E}_{(p \times p)} \cdot \mathbf{A}_{(p \times p)} \cdot \mathbf{E}_{(p \times p)}'$$

With inverse

$$\mathbf{A}_{(p\times p)}^{-1} = \sum_{i=1}^{p} \frac{1}{\lambda_i} \cdot \frac{\mathbf{e}_i}{(p\times 1)} \cdot \frac{\mathbf{e}'_i}{(1\times p)} = \underbrace{\mathbf{E}}_{(p\times p)} \cdot \underbrace{\mathbf{\Lambda}^{-1}}_{(p\times p)} \cdot \underbrace{\mathbf{E}'}_{(p\times p)}$$

since,

$$\begin{aligned} \mathbf{A}_{(p \times p)}^{-1} \cdot \mathbf{A}_{(p \times p)} \\ &= \begin{bmatrix} \mathbf{E}_{(p \times p)} \cdot \mathbf{A}_{(p \times p)}^{-1} \cdot \mathbf{E}_{(p \times p)}' \end{bmatrix} \begin{bmatrix} \mathbf{E}_{(p \times p)} \cdot \mathbf{A}_{(p \times p)} \cdot \mathbf{E}_{(p \times p)}' \\ (p \times p) \cdot (p \times p) \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{E}_{(p \times p)} \cdot \mathbf{A}_{(p \times p)}^{-1} \cdot \begin{bmatrix} \mathbf{E}_{(p \times p)}' \cdot \mathbf{E}_{(p \times p)} \end{bmatrix} \cdot \mathbf{A}_{(p \times p)} \cdot \mathbf{E}_{(p \times p)}', \{\text{Result 2. 2. 2 (b)}\} \\ &= \begin{bmatrix} \mathbf{E}_{(p \times p)} \cdot \mathbf{A}_{(p \times p)}^{-1} \cdot \mathbf{I}_{(p \times p)} \cdot \mathbf{A}_{(p \times p)} \cdot \mathbf{E}_{(p \times p)}', \{\text{E is orthogonal}, \mathbf{E}^{-1} = \mathbf{E}'\} \\ &= \begin{bmatrix} \mathbf{E}_{(p \times p)} \cdot \mathbf{A}_{(p \times p)}^{-1} \cdot \mathbf{A}_{(p \times p)} \cdot \mathbf{E}_{(p \times p)}', \{\text{E is orthogonal}, \mathbf{E}^{-1} = \mathbf{E}'\} \\ &= \begin{bmatrix} \mathbf{E}_{(p \times p)} \cdot \mathbf{A}_{(p \times p)}^{-1} \cdot \mathbf{A}_{(p \times p)} \cdot \mathbf{E}_{(p \times p)}', \{\mathbf{A}^{-1} \text{ is inverse of } \mathbf{A}\} \\ &= \begin{bmatrix} \mathbf{E}_{(p \times p)} \cdot \mathbf{E}_{(p \times p)}', \{\text{E is orthogonal}, \mathbf{E}^{-1} = \mathbf{E}'\} \\ &= \begin{bmatrix} \mathbf{E}_{(p \times p)} \cdot \mathbf{E}_{(p \times p)}', \{\text{E is orthogonal}, \mathbf{E}^{-1} = \mathbf{E}'\} \end{aligned}$$

and

$$\begin{aligned} \mathbf{A}_{(p \times p)} \cdot \mathbf{A}_{(p \times p)}^{-1} \\ &= \begin{bmatrix} \mathbf{E}_{(p \times p)} \cdot \mathbf{A}_{(p \times p)} \cdot \mathbf{E}_{(p \times p)}' \end{bmatrix} \begin{bmatrix} \mathbf{E}_{(p \times p)} \cdot \mathbf{A}_{(p \times p)}^{-1} \cdot \mathbf{E}_{(p \times p)}' \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{E}_{(p \times p)} \cdot \mathbf{A}_{(p \times p)} \cdot \begin{bmatrix} \mathbf{E}_{(p \times p)} \cdot \mathbf{E}_{(p \times p)} \end{bmatrix} \cdot \mathbf{A}_{(p \times p)}^{-1} \cdot \mathbf{E}_{(p \times p)}', \{\text{Result 2. 2. 2 (b)}\} \\ &= \begin{bmatrix} \mathbf{E}_{(p \times p)} \cdot \mathbf{A}_{(p \times p)} \cdot \mathbf{I}_{(p \times p)} \cdot \mathbf{A}_{(p \times p)}^{-1} \cdot \mathbf{E}_{(p \times p)}', \{\text{E is orthogonal}, \mathbf{E}^{-1} = \mathbf{E}'\} \\ &= \begin{bmatrix} \mathbf{E}_{(p \times p)} \cdot \mathbf{A}_{(p \times p)} \cdot \mathbf{A}_{(p \times p)}^{-1} \cdot \mathbf{E}_{(p \times p)}', \\ (p \times p) \cdot (p \times p) \cdot (p \times p) \cdot (p \times p) \end{pmatrix} \\ &= \begin{bmatrix} \mathbf{E}_{(p \times p)} \cdot \mathbf{I}_{(p \times p)} \cdot \mathbf{E}_{(p \times p)}', \{\mathbf{A}^{-1} \text{ is inverse of } \mathbf{A}\} \\ &= \begin{bmatrix} \mathbf{E}_{(p \times p)} \cdot \mathbf{E}_{(p \times p)}' \\ (p \times p) \cdot (p \times p) \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{I}_{(p \times p)} \cdot \mathbf{E}_{(p \times p)}' \\ (p \times p) \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{I}_{(p \times p)} \cdot \mathbf{E}_{(p \times p)}' \\ (p \times p) \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{I}_{(p \times p)} \cdot \mathbf{E}_{(p \times p)}' \\ (p \times p) \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{I}_{(p \times p)} \cdot \mathbf{E}_{(p \times p)}' \\ (p \times p) \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{I}_{(p \times p)} \cdot \mathbf{E}_{(p \times p)} \\ (p \times p) \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{I}_{(p \times p)} \cdot \mathbf{E}_{(p \times p)} \\ (p \times p) \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{I}_{(p \times p)} \cdot \mathbf{E}_{(p \times p)} \\ (p \times p) \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{I}_{(p \times p)} \cdot \mathbf{E}_{(p \times p)} \\ (p \times p) \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{I}_{(p \times p)} \cdot \mathbf{E}_{(p \times p)} \\ (p \times p) \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{I}_{(p \times p)} \cdot \mathbf{E}_{(p \times p)} \\ (p \times p) \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{I}_{(p \times p)} \cdot \mathbf{E}_{(p \times p)} \\ (p \times p) \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{I}_{(p \times p)} \cdot \mathbf{E}_{(p \times p)} \\ (p \times p) \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{I}_{(p \times p)} \cdot \mathbf{E}_{(p \times p)} \\ (p \times p) \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{I}_{(p \times p)} \cdot \mathbf{E}_{(p \times p)} \\ (p \times p) \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{I}_{(p \times p)} \cdot \mathbf{E}_{(p \times p)} \\ (p \times p) \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{I}_{(p \times p)} \cdot \mathbf{E}_{(p \times p)} \\ (p \times p) \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{I}_{(p \times p)} \cdot \mathbf{E}_{(p \times p)} \\ (p \times p) \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{I}_{(p \times p)} \cdot \mathbf{E}_{(p \times p)} \\ (p \times p) \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{I}_{(p \times p)} \cdot \mathbf{E}_{(p \times p)} \\ (p \times p) \end{bmatrix} \\ \end{bmatrix}$$

Definition 2.2.20 (Square-Root Matrix). Let $\Lambda_{(p \times p)}^{1/2}$ denote the diagonal matrix with $\sqrt{\lambda_i}$ as the *i*th diagonal element. Then the square-root matrix, of a positive definite

 $\textit{matrix}_{(p \times p)}^{\mathbf{A}} \textit{ is given by }$

$$\mathbf{A}_{(p\times p)}^{1/2} = \sum_{i=1}^{p} \sqrt{\lambda_i} \cdot \mathbf{e}_i (p \times 1) \cdot \mathbf{e}_i' = \mathbf{E}_{(p\times p)} \cdot \mathbf{A}_{(p\times p)}^{1/2} \cdot \mathbf{E}_{(p\times p)}'$$

[3, p. 66]*.*

Result 2.2.9. The square-root matrix $A_{(p \times p)}^{1/2}$ has the following properties:

(a)
$$\begin{pmatrix} \mathbf{A}^{1/2} \\ (p \times p) \end{pmatrix}' = \mathbf{A}^{1/2} \\ (p \times p)' \begin{pmatrix} \text{that is, } \mathbf{A}^{1/2} \text{ is symmetric} \end{pmatrix}$$

(b) $\mathbf{A}^{1/2} \cdot \mathbf{A}^{1/2} \\ (p \times p)' \cdot (p \times p)' = \mathbf{A} \\ (p \times p)' \begin{pmatrix} \mathbf{E} \\ (p \times p)' \cdot (p \times p) \end{pmatrix} \begin{bmatrix} \mathbf{E} \\ (p \times p)' \cdot \mathbf{A}^{1/2} \\ (p \times p)' \cdot (p \times p) \end{bmatrix} \begin{bmatrix} \mathbf{E} \\ (p \times p)' \cdot \mathbf{A}^{1/2} \\ (p \times p)' \cdot (p \times p)' \end{bmatrix} \}$
(c) $\begin{pmatrix} \mathbf{A}^{1/2} \\ (p \times p) \end{pmatrix}^{-1} = \sum_{i=1}^{p} \frac{1}{\sqrt{\lambda_i}} \cdot \mathbf{e}_i \\ (p \times 1)' \cdot (1 \times p)' = (p \times p)' \cdot \mathbf{A}^{-1/2} \\ (p \times p)' \cdot (p \times p)' \end{pmatrix}$, where $\mathbf{A}^{-1/2} \\ (p \times p)' is a$

diagonal matrix with $1/\sqrt{\lambda_i}$ as the *i*th diagonal element.

(d)
$$\mathbf{A}_{(p \times p)}^{1/2} \cdot \mathbf{A}_{(p \times p)}^{-1/2} = \mathbf{A}_{(p \times p)}^{-1/2} \cdot \mathbf{A}_{(p \times p)}^{1/2} = \mathbf{I}_{(p \times p)} (inverse), and \mathbf{A}_{(p \times p)}^{-1/2} \cdot \mathbf{A}_{(p \times p)}^{-1/2} = \mathbf{A}_{(p \times p)}^{-1}, where \mathbf{A}_{(p \times p)}^{-1/2} = \left(\mathbf{A}_{(p \times p)}^{1/2}\right)^{-1}.$$

[3, p. 66].

Theorem 2.2.1 (Maximization of Quadratic Forms for Points on the Unit Sphere).

Let **B** $_{(p \times p)}$ be a positive definite matrix with eigenvalues $\lambda_1 > \lambda_2 > \cdots > \lambda_p > 0$ and

associated normalized eigenvectors $\underset{(p\times 1)}{\mathbf{e}_1}$, $\underset{(p\times 1)}{\mathbf{e}_2}$, ... , $\underset{(p\times 1)}{\mathbf{e}_p}$. Then

$$\max_{\substack{\mathbf{x} \neq \mathbf{0} \\ (p \times 1) \stackrel{\neq}{}(p \times 1)}} \frac{\mathbf{x}' \cdot \mathbf{B} \cdot \mathbf{x}}{(1 \times p) \cdot (p \times p) \cdot (p \times 1)} = \lambda_1 \qquad \left(attained when \mathbf{x} = \mathbf{e}_1 \\ (p \times 1) \stackrel{\neq}{}(p \times 1) \stackrel{\mathbf{x}' \cdot \mathbf{X}}{(1 \times p) \cdot (p \times p) \cdot (p \times 1)} = \lambda_p \qquad \left(attained when \mathbf{x} = \mathbf{e}_p \\ (p \times 1) \stackrel{\mathbf{x}' \cdot \mathbf{B} \cdot \mathbf{x}}{(1 \times p) \cdot (p \times 1)} = \lambda_p \qquad \left(attained when \mathbf{x} = \mathbf{e}_p \\ (p \times 1) \stackrel{\mathbf{x}' \cdot \mathbf{X}}{(1 \times p) \cdot (p \times 1)} = \lambda_p \right)$$

Moreover,

$$\max_{\substack{\mathbf{x} \ (p \times 1) \ (p$$

(attained when $\underset{(p \times 1)}{\mathbf{x}} = \underset{(p \times 1)}{\mathbf{e}_{k+1}}, k = 1, 2, \dots, p-1$)

where the symbol \perp is read "is perpendicular to."

Proof: Let $\mathop{\mathbf{E}}_{(p \times p)}$ be the orthogonal matrix whose columns are the eigenvectors

 \mathbf{e}_1 , \mathbf{e}_2 , ..., \mathbf{e}_p and $\bigwedge_{(p \times p)}$ be the diagonal matrix with eigenvalues $\lambda_1, \lambda_1, \dots, \lambda_p$

along the main diagonal. Let $\mathbf{B}_{(p \times p)}^{1/2} = \mathbf{E}_{(p \times p)} \cdot \mathbf{\Lambda}_{(p \times p)}^{1/2} \cdot \mathbf{E}'_{(p \times p)}$ (square-root matrix)

and $\mathbf{y}_{(p \times 1)} = \mathbf{E}'_{(p \times p)} \cdot \mathbf{x}_{(p \times 1)}$.

Consequently,
$$\mathbf{x}_{(p \times 1)} \neq \mathbf{0}_{(p \times 1)}$$
 implies $\mathbf{y}_{(p \times 1)} \neq \mathbf{0}_{(p \times 1)}$ because $\mathbf{E}'_{(p \times p)}$ is an

orthogonal matrix and hence has inverse $\mathop{\mathbf{E}}_{(p \times p)} \left\{ \mathop{\mathbf{E}'}_{(p \times p)} = \mathop{\mathbf{E}^{-1}}_{(p \times p)} \right\}$. Thus,

 $\mathbf{x}_{(p\times 1)} = \mathbf{E}_{(p\times p)} \cdot \mathbf{y}_{(p\times 1)}. \text{ But } \mathbf{x}_{(p\times 1)} \text{ is a nonzero vector, and } \mathbf{0}_{(p\times 1)} \neq \mathbf{x}_{(p\times 1)} = \mathbf{E}_{(p\times p)} \cdot \mathbf{y}_{(p\times 1)}$

implies that $\mathbf{y}_{(p \times 1)} \neq \mathbf{0}_{(p \times 1)}$.

Thus,

$$\begin{aligned} \frac{\mathbf{x}'_{(1\times p)} \cdot (\mathbf{p} \times \mathbf{p}) \cdot (\mathbf{p} \times \mathbf{1})}{\mathbf{x}'_{(1\times p)} \cdot (\mathbf{p} \times \mathbf{1})} \\ &= \frac{(\mathbf{x}'_{p}) \cdot \left[\mathbf{B}^{1/2} \cdot \mathbf{B}^{1/2} \right] \cdot (\mathbf{p} \times \mathbf{1})}{\mathbf{x}'_{(1\times p)} \cdot (\mathbf{p} \times \mathbf{1})}, \left\{ \mathbf{B}^{1/2} \cdot \mathbf{B}^{1/2} \right] = \mathbf{B}_{(p\times p)}, \mathbf{Result 2.2.9.(b)} \right\} \\ &= \frac{(\mathbf{x}'_{p}) \cdot \left[\left[\mathbf{E}_{(p\times p)} \cdot \mathbf{A}^{1/2} \cdot \mathbf{E}_{(p\times p)} \right] \right] \left[\mathbf{E}_{(p\times p)} \cdot \mathbf{A}^{1/2} \cdot \mathbf{E}_{(p\times p)} \right] \right] \cdot (\mathbf{p} \times \mathbf{1})}{\mathbf{x}'_{p} \cdot \mathbf{x}}, \\ &= \frac{(\mathbf{x}'_{p}) \cdot \left[\left[\mathbf{E}_{(p\times p)} \cdot \mathbf{A}^{1/2} \cdot \mathbf{E}_{(p\times p)} \right] \right] \left[\mathbf{E}_{(p\times p)} \cdot \mathbf{A}^{1/2} \cdot \mathbf{E}_{(p\times p)} \right] \right] \cdot (\mathbf{p} \times \mathbf{1})}{\mathbf{x}'_{p} \cdot \mathbf{x}}, \\ &= \frac{(\mathbf{x}'_{p}) \cdot \left[\mathbf{E}_{(p\times p)} \cdot \mathbf{A}^{1/2} \cdot \mathbf{E}_{(p\times p)} \right] \cdot (\mathbf{p} \times \mathbf{p})}{\mathbf{x}'_{p} \cdot \mathbf{x}}, \\ &= \frac{(\mathbf{x}'_{p}) \cdot \left[\mathbf{E}_{(p\times p)} \cdot \mathbf{A}^{1/2} \cdot \mathbf{E}_{(p\times p)} \right] \cdot (\mathbf{p} \times \mathbf{p})}{\mathbf{x}'_{p} \cdot \mathbf{x}}, \\ &= \frac{(\mathbf{x}'_{p}) \cdot \left[\mathbf{E}_{(p\times p)} \cdot \mathbf{A}^{1/2} \cdot \mathbf{E}_{(p\times p)} \right] \cdot (\mathbf{p} \times \mathbf{p})}{\mathbf{x}'_{p} \cdot \mathbf{x}}, \\ &= \frac{(\mathbf{x}'_{p}) \cdot \left[\mathbf{E}_{(p\times p)} \cdot \mathbf{A}^{1/2} \cdot \mathbf{A}^{1/2} \cdot \mathbf{E}_{(p\times p)} \right] \cdot (\mathbf{p} \times \mathbf{p})}{\mathbf{x}'_{p} \cdot \mathbf{x}}, \\ &= \frac{(\mathbf{x}'_{p}) \cdot \left[\mathbf{E}_{(p\times p)} \cdot (\mathbf{p} \times \mathbf{p}) \right] \left[\mathbf{A}^{1/2} \cdot \mathbf{E}_{(p\times p)} \cdot (\mathbf{p} \times \mathbf{p}) \right] \cdot (\mathbf{p} \times \mathbf{p})}{\mathbf{x}'_{p} \cdot \mathbf{x}}, \\ &= \frac{\left[\mathbf{x}'_{p} \cdot \mathbf{E}_{(p\times p)} \right] \left[\mathbf{A}^{1/2} \cdot \mathbf{A}^{1/2} \right] \left[\mathbf{E}'_{p} \cdot \mathbf{x} \right] \\ &= \frac{\left[\mathbf{x}'_{p} \cdot \mathbf{E}_{p} \cdot \mathbf{E}_{p} \right] \left[\mathbf{A}^{1/2} \cdot \mathbf{A}^{1/2} \right] \left[\mathbf{E}'_{p} \cdot \mathbf{x} \right]}{\mathbf{x}'_{p} \cdot \mathbf{x}}, \\ &= \frac{\left[\mathbf{x}'_{p} \cdot \mathbf{E}_{p} \cdot \mathbf{E}_{p} \right] \left[\mathbf{A}^{1/2} \cdot \mathbf{A}^{1/2} \right] \left[\mathbf{E}'_{p} \cdot \mathbf{x} \right] \\ &= \frac{\left[\mathbf{x}'_{p} \cdot \mathbf{E}_{p} \cdot \mathbf{E}_{p} \right] \left[\mathbf{A}^{1/2} \cdot \mathbf{A}^{1/2} \right] \left[\mathbf{E}'_{p} \cdot \mathbf{x} \right]}{\mathbf{x}'_{p} \cdot \mathbf{x}}, \\ &= \frac{\left[\mathbf{x}'_{p} \cdot \mathbf{E}_{p} \cdot \mathbf{E}_{p} \right] \left[\mathbf{A}^{1/2} \cdot \mathbf{A}^{1/2} \right] \left[\mathbf{E}'_{p} \cdot \mathbf{x} \right] \\ &= \frac{\left[\mathbf{x}'_{p} \cdot \mathbf{E}_{p} \cdot \mathbf{E}_{p} \right] \left[\mathbf{A}^{1/2} \cdot \mathbf{A}^{1/2} \right] \left[\mathbf{E}'_{p} \cdot \mathbf{x} \right]}{\mathbf{x}'_{p} \cdot \mathbf{x}}, \\ &= \frac{\left[\mathbf{x}'_{p} \cdot \mathbf{E}_{p} \cdot \mathbf{E}_{p} \right] \left[\mathbf{A}^{1/2} \cdot \mathbf{E}_{p} \cdot \mathbf{E}_{p} \right] \left[\mathbf{E}'_{p} \cdot \mathbf{E}'_{p} \cdot \mathbf{E}'_{p} \right]}{\mathbf{x}'_{p} \cdot \mathbf{E}'_{p} \cdot \mathbf{E}'_{p} \cdot \mathbf{E}'_{p} \right]} \\ &= \frac{\left[\mathbf{x}'_{p} \cdot \mathbf{E}_{p} \cdot \mathbf{E}_{p} \cdot \mathbf{E}_{p} \right] \left[\mathbf{x}'_$$

$$= \frac{\begin{bmatrix} \mathbf{x}' & \mathbf{E} \\ (1\times p) & (p\times p) \end{bmatrix} \begin{bmatrix} \mathbf{A}^{1/2} & \mathbf{A}^{1/2} \\ (p\times p) & (p\times p) \end{bmatrix} \begin{bmatrix} \mathbf{E}' & \mathbf{x} \\ (p\times p) & (p\times 1) \end{bmatrix}}{\begin{bmatrix} \mathbf{x}' & \mathbf{E} \\ (1\times p) & (p\times p) & (p\times p) \end{bmatrix} \begin{bmatrix} \mathbf{E}' & \mathbf{x} \\ (p\times p) & (p\times 1) \end{bmatrix}}, \{\text{Result 2.2.2.(b)}\}$$

$$= \frac{\mathbf{y}' \begin{bmatrix} \mathbf{A}^{1/2} & \mathbf{A}^{1/2} \\ (1\times p) & (p\times p) & (p\times p) \end{bmatrix} (p\times 1)}{\mathbf{y}' & \mathbf{y} \\ (1\times p) & (p\times 1) \end{bmatrix}}, \begin{bmatrix} \mathbf{A}_{1} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_{2} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{A}_{p} \end{bmatrix}}, \begin{bmatrix} \mathbf{y} \\ (p\times 1) \end{bmatrix}$$

$$= \frac{\mathbf{y}' & \mathbf{A} \\ (1\times p) & (p\times 1) \\ \mathbf{y}' & \mathbf{y} \\ (1\times p) & (p\times 1) \end{bmatrix}}{\begin{bmatrix} \mathbf{A}_{1} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_{2} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{A}_{p} \end{bmatrix}}, \begin{bmatrix} \mathbf{y}_{1} \\ \mathbf{y}_{1} \\ \mathbf{y}_{2} \\ (1\times p) & (p\times 1) \end{bmatrix}$$

$$= \frac{\begin{bmatrix} \mathbf{y}' & \mathbf{A} & \mathbf{y} \\ (1\times p) & (p\times 1) \\ \mathbf{y}' & \mathbf{y} \\ (1\times p) & (p\times 1) \end{bmatrix}}{\begin{bmatrix} \mathbf{y}_{1} & \mathbf{y}_{2} \\ (1\times p) & (p\times 1) \end{bmatrix}} \begin{bmatrix} \mathbf{A}_{1} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{A}_{2} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{A}_{p} \end{bmatrix}}, \begin{bmatrix} \mathbf{y}_{1} \\ \mathbf{y}_{2} \\ \mathbf{y}_{p} \\ (p\times 1) \end{bmatrix}$$

$$= \frac{\begin{bmatrix} \mathbf{y}_{1} & \mathbf{y}_{2} & \cdots & \mathbf{y}_{1} \\ \begin{bmatrix} \mathbf{y}_{1} & \mathbf{y}_{2} & \cdots & \mathbf{y}_{1} \\ (1\times p) & (p\times 1) \end{bmatrix}}{\begin{bmatrix} \mathbf{y}_{1} & \mathbf{y}_{2} \\ (1\times p) & (p\times 1) \end{bmatrix}}$$

$$= \frac{\sum_{i=1}^{p} \lambda_i y_i^2}{\sum_{i=1}^{p} y_i^2}$$
$$\leq \lambda_1 \cdot \frac{\sum_{i=1}^{p} y_i^2}{\sum_{i=1}^{p} y_i^2}$$
$$= \lambda_1$$

Setting,

$$\mathbf{x}_{(p\times 1)} = \mathbf{e}_{1} = \begin{bmatrix} e_{11} \\ e_{21} \\ \vdots \\ e_{p1} \\ (p\times 1) \end{bmatrix}$$

$$\mathbf{y} \\
 (p \times 1) \\
 = \underbrace{\mathbf{E}'_{(p \times p)} \cdot \mathbf{e}_{1}}_{(p \times 1)} \\
 = \begin{bmatrix} e_{11} & e_{21} & \cdots & e_{p1} \\ e_{12} & e_{22} & \cdots & e_{p2} \\ \vdots & \vdots & \ddots & \vdots \\ e_{1p} & e_{2p} & \cdots & e_{pp} \end{bmatrix} \begin{bmatrix} e_{11} \\ e_{21} \\ \vdots \\ e_{p1} \\ (p \times p) \end{bmatrix} \\
 (p \times p)$$

$$= \begin{bmatrix} \mathbf{e}_1' \mathbf{e}_1 \\ \mathbf{e}_2' \mathbf{e}_1 \\ \vdots \\ \mathbf{e}_p' \mathbf{e}_1 \end{bmatrix} _{(p \times 1)}$$



That is,

$$= \frac{\begin{bmatrix} 1_1, 0_2, \dots, 0_p \end{bmatrix} \cdot \begin{bmatrix} \lambda_1 \\ 0_2 \\ \vdots \\ 0_p \end{bmatrix}}{1} = \frac{\lambda_1}{1} = \lambda_1$$

A similar argument produces the second part.

Now,

$$\mathbf{x}_{(p\times1)} = \mathbf{E}_{(p\timesp)} \cdot \mathbf{y}_{(p\times1)} = y_1 \cdot \mathbf{e}_1 + y_2 \cdot \mathbf{e}_2 + \dots + y_p \cdot \mathbf{e}_p,$$

(p×1)
so
$$\mathbf{x}_{(p\times1)} \perp \mathbf{e}_1, \mathbf{e}_2, \dots, \mathbf{e}_k \text{ implies}$$

(p×1)

$$0 = \mathbf{e}'_i \cdot \mathbf{x}_{(1 \times p)} = y_1 \cdot \mathbf{e}'_i \cdot \mathbf{e}_1 + \dots + y_p \cdot \mathbf{e}'_i \cdot \mathbf{e}_p = y_i, \qquad i \le k$$

Therefore, for $\mathbf{x}_{(p \times 1)}$ perpendicular to the first k eigenvectors \mathbf{e}_i , the left-hand side

of the inequality in becomes

$$\frac{\mathbf{x}' \cdot \mathbf{B} \cdot \mathbf{x}}{\frac{(1 \times p)}{\mathbf{x}'} \cdot \frac{\mathbf{B}}{(p \times p)} \cdot \frac{\mathbf{x}}{(p \times 1)}} = \frac{\sum_{i=k+1}^{p} \lambda_i y_i^2}{\sum_{i=k+1}^{p} y_i^2}$$

Taking $y_{k+1} = 1$, $y_{k+2} = \cdots = y_p = 0$ gives the asserted maximum.

For a fixed $\mathbf{x}_{0}_{(p \times 1)} \neq \mathbf{0}_{(p \times 1)}$,

$$\frac{\mathbf{x}_{0}^{\prime} \cdot \mathbf{B} \cdot \mathbf{x}_{0}}{\left(1 \times p\right) \cdot \left(p \times p\right) \cdot \left(p \times 1\right)}}{\mathbf{x}_{0}^{\prime} \cdot \mathbf{x}_{0}}_{(1 \times p) \quad (p \times 1)}$$

has the same value as

$$\mathbf{x}' \cdot \mathbf{B} \cdot \mathbf{x}_{(1 \times p)} \cdot (p \times p) \cdot (p \times 1)'$$

where

$$\mathbf{x}'_{(1\times p)} = \frac{\mathbf{x}'_{0}}{\sqrt{\frac{(1\times p)}{\sqrt{\mathbf{x}'_{0} \cdot \mathbf{x}_{0}}}}} = \frac{\mathbf{x}'_{0}}{\frac{(1\times p)}{L_{\mathbf{x}_{0}}}}$$

is of unit length. Consequently,

$$\max_{\substack{\mathbf{x} \neq \mathbf{0} \\ (p \times 1) \neq (p \times 1)}} \frac{\mathbf{x}' \cdot \mathbf{B} \cdot \mathbf{x}}{(1 \times p) \cdot (p \times 1)} = \lambda_1 \qquad \left(attained when \mathbf{x}_{(p \times 1)} = \mathbf{e}_1 \\ (p \times 1) \neq (p \times 1) = \lambda_1 \right)$$

says that the largest eigenvalue, λ_1 , is the maximum value of the quadratic form

$$\mathbf{x}' \cdot \mathbf{B} \cdot \mathbf{x} \\ {}_{(1 \times p)} \cdot {}_{(p \times p)} \cdot {}_{(p \times 1)} \mathbf{x}$$

for all points $\underset{(p \times 1)}{\mathbf{x}}$ whose distance from the origin is unity. Similarly, λ_p is the smallest value of the quadratic form for all points $\underset{(p \times 1)}{\mathbf{x}}$ one unit from the origin. The largest and smallest eigenvalues thus represent extreme values of

$$\mathbf{x}' \cdot \mathbf{B} \cdot \mathbf{x} \\ {}_{(1 \times p)} \cdot {}_{(p \times p)} \cdot {}_{(p \times 1)} \mathbf{x}$$

for points on the unit sphere. The "intermediate" eigenvalues of the $p \times p$ positive definite matrix $\underset{(p \times p)}{\mathbf{B}}$ also have an interpretation as extreme values when $\underset{(p \times 1)}{\mathbf{x}}$ is further restricted to be perpendicular to the earlier choices [3, pp. 80-81].

Chapter 3

Multivariate Population Theory

3.1 Population Random Matrix

Definition 3.1.1 (Population Random Matrix **X**). *A* **population random matrix** $\underset{(n \times p)}{\mathbf{X}}$ *for continuous variables is a matrix whose elements are population continuous random variables. Specifically, let* $\underset{(n \times p)}{\mathbf{X}} = \{X_{ij}\}$ *be an* $n \times p$ *population random matrix*

$$\mathbf{X}_{(n \times p)} = \begin{bmatrix} X_{11} & X_{12} & \cdots & X_{1j} & \cdots & X_{1p} \\ X_{21} & X_{22} & \cdots & X_{2j} & \cdots & X_{2p} \\ \vdots & \vdots & & \vdots & & \vdots \\ X_{i1} & X_{i2} & \cdots & X_{ij} & \cdots & X_{ip} \\ \vdots & \vdots & & \vdots & & \vdots \\ X_{n1} & X_{n2} & \cdots & X_{nj} & \cdots & X_{np} \end{bmatrix} = \begin{bmatrix} X_{11} & X_{12} & \cdots & X_{1p} \\ X_{21} & X_{22} & \cdots & X_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ X_{n1} & X_{n2} & \cdots & X_{nj} & \cdots & X_{np} \end{bmatrix}$$

for i = 1,2, ..., *n rows and j* = 1,2, ..., *p columns* [3, p. 66].

3.2 Population Random Vector, Mean Vector, Variance-Covariance Matrix, and Correlation Matrix

3.2.1 Population Random Vector

Definition 3.2.1 (Population Random Vector **X**). *A* **population random vector** $\underset{(p \times 1)}{\mathbf{X}}$

for continuous random variables is a vector whose elements are population continuous random variables from a p – variate population. Specifically, let $\mathbf{X}_{(p \times 1)} = \{X_i\}$ be a $p \times 1$ population random vector

$$\mathbf{X}_{(p\times1)} = \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_p \end{bmatrix}$$

[3, p. 68]*.*

3.2.2 Probability Density Functions

Definition 3.2.2 (Joint Probability Density Function). *The collective behavior of the* p *continuous random variables* $X_1, X_2, ..., X_p$ *or, equivalently, the population random vector* $\mathbf{X}_{(p \times 1)}$, *is described by a* **joint probability density function (pdf)**

$$f\left(\mathbf{x}_{(p\times 1)}\right) = f_{12\cdots p}(x_1, x_2, \dots, x_p)$$

[3, p. 68] where $x_i \in \mathbb{R}$, i = 1, 2, ..., p. Satisfying constraints,

(a)
$$f_{12\cdots p}(x_1, x_2, \dots, x_p) \ge 0$$

(b) $\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} f_{12\cdots p}(x_1, x_2, \dots, x_p) dx_1 dx_2 \cdots dx_p = 1$

Definition 3.2.3 (Univariate Marginal Probability Density Function). Each element of

 $\mathbf{X}_{(p \times 1)}$ is a population random variable with its own univariate marginal pdf defined as

 $f_i(x_i)$. Specifically,

$$f_i(x_i) = \begin{cases} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} f_{12\cdots p}(x_1, x_2, \dots, x_p) dx_1 dx_2 \cdots dx_{i-1} dx_{i+1} \cdots dx_p \\ 0 & \text{otherwise} \end{cases}$$

[3, p. 68] for $x_i \in \mathbb{R}$, i = 1, 2, ..., p. Satisfying constraints,

1

(a) $f_i(x_i) \ge 0$ (b) $\int_{-\infty}^{\infty} f_i(x_i) dx_i = 0$

(b)
$$\int_{-\infty} f_i(x_i) dx_i =$$

Definition 3.2.4 (Bivariate Marginal Probability Density Function). *Each pair of* elements of $\underset{(p \times 1)}{\mathbf{X}}$ is a bivariate population random vector (X_i, X_k) with a bivariate

(joint) marginal pdf defined as $f_{ik}(x_i, x_k)$. Specifically,

$$f_{ik}(x_i, x_k) = \begin{cases} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} f_{12\cdots p}(x_1, x_2, \dots, x_p) dx_1 dx_2 \cdots dx_{i-1} dx_{i+1} \cdots dx_{k-1} dx_{k+1} \cdots dx_p \\ 0 & \text{otherwise} \end{cases}$$

- [3, p. 68] for $(x_i, x_k) \in \mathbb{R}$, $i, k = 1, 2, ..., p, i \neq k$. Satisfying constraints,
- (a) $f_{ik}(x_i, x_k) \ge 0$ (b) $\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f_{ik}(x_i, x_k) dx_i dx_k = 1$

3.2.3 Population Parameters

Definition 3.2.5 (Univariate Marginal Population Mean). *The* **univariate marginal population means** μ_i *are defined as* $\mu_i = E(X_i)$ *with pdf* $f_i(x_i)$. *Specifically, if they exist (finite)*

$$\mu_i = E(X_i) = \int_{-\infty}^{\infty} x_i f_i(x_i) dx_i$$

for i = 1, 2, ..., p *where* $-\infty < \mu_i < \infty$ [3, p. 68].

Definition 3.2.6 (Univariate Marginal Population Variance). *The* **univariate marginal population variances** σ_{ii} are defined as $\sigma_{ii} = var(X_i) = E(X_i - \mu_i)^2$ with pdf $f_i(x_i)$. *Specifically, if they exist*

$$\sigma_{ii} = \operatorname{var}(X_i) = E(X_i - \mu_i)^2 = \int_{-\infty}^{\infty} (x_i - \mu_i)^2 f_i(x_i) dx_i$$

for i = 1, 2, ..., p where $0 < \sigma_{ii} < \infty$. The univariate marginal population standard deviation is the square-root of the variance $\sqrt{\sigma_{ii}}$ [3, p. 68].

Definition 3.2.7 (Bivariate Marginal Population Covariance). *The* **bivariate marginal population covariances** σ_{ik} *are defined as* $\sigma_{ik} = \text{cov}(X_i, X_k) = E(X_i - \mu_i)(X_k - \mu_k)$ *with pdf* $f_{ik}(x_i, x_k)$. *Specifically, if they exist*

$$\sigma_{ik} = \operatorname{cov}(X_i, X_k) = E(X_i - \mu_i)(X_k - \mu_k)$$
$$= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (x_i - \mu_i)(x_k - \mu_k) f_{ik}(x_i, x_k) dx_i dx_k$$

for i, *k* = 1,2, ..., *p where* $-\infty < \sigma_{ik} < \infty$ [3, p. 68].

Note that $\sigma_{ik} = \sigma_{ki}$ and when i = k the bivariate marginal population

covariance becomes the univariate marginal population variance σ_{ii} .

Definition 3.2.8 (Bivariate Marginal Population Correlation). *The* **bivariate marginal population correlations** ρ_{ik} *are defined as* $\rho_{ik} = \operatorname{corr}(X_i, X_k) = \frac{\sigma_{ik}}{\sqrt{\sigma_{ii}}\sqrt{\sigma_{kk}}}$. Specifically,

if they exist

$$\rho_{ik} = \operatorname{corr}(X_i, X_k) = \frac{\sigma_{ik}}{\sqrt{\sigma_{ii}}\sqrt{\sigma_{kk}}}$$

$$= \frac{E(X_{i} - \mu_{i})(X_{k} - \mu_{k})}{\sqrt{E(X_{i} - \mu_{i})^{2}}\sqrt{E(X_{k} - \mu_{k})^{2}}}$$
$$= \frac{\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (x_{i} - \mu_{i})(x_{k} - \mu_{k})f_{ik}(x_{i}, x_{k})dx_{i}dx_{k}}{\sqrt{\int_{-\infty}^{\infty} (x_{i} - \mu_{i})^{2}f_{i}(x_{i})dx_{i}}\sqrt{\int_{-\infty}^{\infty} (x_{k} - \mu_{k})^{2}f_{k}(x_{k})dx_{k}}}$$

for i, k = 1, 2, ..., p where $-1 \le \rho_{ik} \le 1$ [3, p. 71].

Note that $\rho_{ik} = \rho_{ki}$ and when i = k the bivariate marginal population correlation becomes $\rho_{ii} = \frac{\sigma_{ii}}{\sqrt{\sigma_{ii}}\sqrt{\sigma_{ii}}} = \frac{\sigma_{ii}}{\sigma_{ii}} = 1.$

3.2.4 Independent Random Variables

Definition 3.2.9 (Statistically Independent). *If the bivariate marginal pdf* $f_{ik}(x_i, x_k)$ *for continuous random variables* (X_i, X_k) *, can be written as the product of the corresponding univariate marginal pdf's* $f_i(x_i)$, $f_k(x_k)$ *so that*

$$f_{ik}(x_i, x_k) \equiv f_i(x_i) f_k(x_k)$$

then X_i and X_k are said to be statistically independent.

Furthermore, if, (X_i, X_k) are statistical independent, then $\sigma_{ik} = 0$ and $\rho_{ik} = 0$ [3, pp. 69, 71].

Definition 3.2.10 (Mutually Statistically Independent). *The p population continuous random variables* $(X_1, X_2, ..., X_p)$ *are* **mutually statistically independent** *if their joint pdf can be factored as a product of their univariate marginal pdf's*

$$f_{12\cdots p}(x_1, x_2, \dots, x_p) \equiv f_1(x_1)f_2(x_2)\cdots f_p(x_p)$$

[3, p. 69]*.*

In addition, if, $(X_1, X_2, ..., X_p)$ are mutually statistically independent, then every subset of continuous population random variables ≥ 2 are also mutually statistically independent.

3.2.5 Population Mean Vector

Definition 3.2.11 (Population Mean Vector for X). *The* population mean vector for

 $\mathbf{X}_{(p \times 1)}$ or expected value of a population random vector is a random vector consisting of the univariate marginal expectations of each of its elements. Then, if these expectations exist, the population mean vector for $\mathbf{X}_{(p \times 1)}$, denoted by $\mathbf{\mu}_{\mathbf{X}} = E(\mathbf{X})$, is

the $p \times 1$ *vector*

$$\boldsymbol{\mu}_{\mathbf{X}} = E(\mathbf{X})_{(p\times1)} = \begin{bmatrix} E(X_1) \\ E(X_2) \\ \vdots \\ E(X_p) \end{bmatrix} = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_p \end{bmatrix}_{(p\times1)}$$

where $-\infty < \mu_i < \infty$ *, for* i = 1, 2, ..., p [3, p. 69].

3.2.6 Population Variance-Covariance Matrix

Theorem 3.2.1 (Population Variance-Covariance Matrix for **X**). *The* **population variance-covariance matrix for** $\underset{(p\times1)}{\mathbf{X}}$ *is a symmetric matrix containing the p univariate marginal population variances* σ_{ii} *and the* p(p-1)/2 *distinct bivariate marginal population covariances* σ_{ik} (i < k). *Then, if these variances and covariances exist, the* $p \times p$ *population variance-covariance matrix for* $\underset{(p\times1)}{\mathbf{X}}$ *is given*

by

$$\sum_{(p \times p)} = \operatorname{Cov}(\mathbf{X})_{(p \times p)} = E(\mathbf{X} - \boldsymbol{\mu}_{\mathbf{X}})(\mathbf{X} - \boldsymbol{\mu}_{\mathbf{X}})'_{(1 \times p)}$$

where $\mu_{\mathbf{X}} = E(\mathbf{X})$ is the population mean vector.

Proof. Use linearity of the operator *E*, Definition 2.1.2, 2.1.11, 2.2.5, 3.2.6., and 3.2.7.

 $\sum_{(p \times p)} \mathbf{x}$

 $= \operatorname{Cov}_{(p \times p)}(\mathbf{X})$

$$= E\left(\begin{pmatrix} X_{1} - \mu_{1} \\ X_{2} - \mu_{2} \\ \vdots \\ X_{p} - \mu_{p} \\ (p \times 1) \end{pmatrix}' \begin{bmatrix} X_{1} - \mu_{1} \\ X_{2} - \mu_{2} \\ \vdots \\ X_{p} - \mu_{p} \end{bmatrix} \cdot \begin{bmatrix} X_{1} - \mu_{1}, X_{2} - \mu_{2}, \dots, X_{p} - \mu_{p} \end{bmatrix} \right)$$

$$= E\left(\begin{pmatrix} (X_{1} - \mu_{1})^{2} & (X_{1} - \mu_{1})(X_{2} - \mu_{2}) & \cdots & (X_{1} - \mu_{1})(X_{p} - \mu_{p}) \\ (X_{2} - \mu_{2})(X_{1} - \mu_{1}) & (X_{2} - \mu_{2})^{2} & \cdots & (X_{2} - \mu_{2})(X_{p} - \mu_{p}) \\ \vdots & \vdots & \ddots & \vdots \\ (X_{p} - \mu_{p})(X_{1} - \mu_{1}) & (X_{p} - \mu_{p})(X_{2} - \mu_{2}) & \cdots & (X_{p} - \mu_{p})^{2} \end{bmatrix} \right)$$

$$= E \begin{bmatrix} (X_1 - \mu_1)^2 & (X_1 - \mu_1)(X_2 - \mu_2) & \cdots & (X_1 - \mu_1)(X_p - \mu_p) \\ (X_2 - \mu_2)(X_1 - \mu_1) & (X_2 - \mu_2)^2 & \cdots & (X_2 - \mu_2)(X_p - \mu_p) \\ \vdots & \vdots & \ddots & \vdots \\ (X_p - \mu_p)(X_1 - \mu_1) & (X_p - \mu_p)(X_2 - \mu_2) & \cdots & (X_p - \mu_p)^2 \end{bmatrix}$$
$$= \begin{bmatrix} E(X_1 - \mu_1)^2 & E(X_1 - \mu_1)(X_2 - \mu_2) & \cdots & E(X_1 - \mu_1)(X_p - \mu_p)^2 \\ E(X_2 - \mu_2)(X_1 - \mu_1) & E(X_2 - \mu_2)^2 & \cdots & E(X_2 - \mu_2)(X_p - \mu_p) \end{bmatrix}$$

$$\begin{bmatrix} \vdots & \ddots & \vdots \\ E(X_p - \mu_p)(X_1 - \mu_1) & E(X_p - \mu_p)(X_2 - \mu_2) & \cdots & E(X_p - \mu_p)^2 \\ (p \times p) & & \end{bmatrix}$$

$$= \begin{bmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1p} \\ \sigma_{21} & \sigma_{22} & \cdots & \sigma_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{p1} & \sigma_{p2} & \cdots & \sigma_{pp} \end{bmatrix}$$

[3, pp. 69-70] ■

3.2.7 Population Standard Deviation Matrix

Definition 3.2.12 (Population Standard Deviation Matrix for **X**). *The* **population standard deviation matrix for** $\underset{(p\times 1)}{\mathbf{X}}$ *is a diagonal matrix containing the p univariate marginal population standard deviations* $\sqrt{\sigma_{ii}}$ *along the main diagonal. Then, if these standard deviations exist, the population standard deviation matrix for* $\underset{(p\times 1)}{\mathbf{X}}$ *is*

denoted by $\mathbf{V}_{(p \times p)}^{1/2}$, is the $p \times p$ matrix

$$\mathbf{V}_{(p \times p)}^{1/2} = \begin{bmatrix} \sqrt{\sigma_{11}} & 0 & \cdots & 0 \\ 0 & \sqrt{\sigma_{22}} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sqrt{\sigma_{pp}} \end{bmatrix}$$

with inverse

$$(\mathbf{V}_{(p\times p)}^{1/2})^{-1} = \mathbf{V}_{(p\times p)}^{-1/2} = \begin{bmatrix} \frac{1}{\sqrt{\sigma_{11}}} & 0 & \cdots & 0\\ 0 & \frac{1}{\sqrt{\sigma_{22}}} & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots\\ 0 & 0 & \cdots & \frac{1}{\sqrt{\sigma_{pp}}} \end{bmatrix}_{(p\times p)}$$

[3, pp. 59, 72]*.*

3.2.8 Population Correlation Matrix

Theorem 3.2.2 (Population Correlation Matrix for **X**). *The* **population correlation matrix for** $\underset{(p \times 1)}{\mathbf{X}}$ *is a symmetric matrix containing the p bivariate marginal population correlations* $\rho_{ii} = 1$ *along the main diagonal and the* p(p - 1)/2 *distinct bivariate marginal population correlations* ρ_{ik} (i < k). *Then, if these correlations exist, the* $p \times p$ *population correlation matrix for* $\underset{(p \times 1)}{\mathbf{X}}$ *is given by*

$$\boldsymbol{\rho}_{(p \times p)} = \operatorname{Corr}_{(p \times p)} (\mathbf{X}) = (\mathbf{V}^{1/2})^{-1} \cdot \sum_{\substack{X \\ (p \times p)}} \cdot (\mathbf{V}^{1/2})^{-1} (\mathbf{V}^{1/2})^{-1}$$

where $(\mathbf{V}_{(p \times p)}^{1/2})^{-1}$ is the inverse population standard deviation matrix and $\sum_{(p \times p)}$ is the

population variance-covariance matrix [3, p. 72].

Proof. Use Definition 2.2.5, 3.2.6. and 3.2.7.

$$\boldsymbol{\rho} = \operatorname{Corr}(\mathbf{X})$$
$$= (\mathbf{V}_{(p \times p)}^{1/2})^{-1} \cdot \sum_{(p \times p)} \cdot (\mathbf{V}_{(p \times p)}^{1/2})^{-1}$$
$$= \mathbf{V}_{(p \times p)}^{-1/2} \cdot \sum_{(p \times p)} \cdot \mathbf{V}_{(p \times p)}^{-1/2}$$

$$= \begin{bmatrix} \frac{1}{\sqrt{\sigma_{11}}} & 0 & \cdots & 0\\ 0 & \frac{1}{\sqrt{\sigma_{22}}} & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots\\ 0 & 0 & \cdots & \frac{1}{\sqrt{\sigma_{pp}}} \end{bmatrix} \cdot \begin{bmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1p}\\ \sigma_{21} & \sigma_{22} & \cdots & \sigma_{2p}\\ \vdots & \vdots & \ddots & \vdots\\ \sigma_{p1} & \sigma_{p2} & \cdots & \sigma_{pp} \end{bmatrix} \cdot \begin{bmatrix} \frac{1}{\sqrt{\sigma_{11}}} & 0 & \cdots & 0\\ 0 & \frac{1}{\sqrt{\sigma_{22}}} & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots\\ 0 & 0 & \cdots & \frac{1}{\sqrt{\sigma_{pp}}} \end{bmatrix}_{(p \times p)}$$

$$\begin{bmatrix} \frac{\sigma_{11}}{\sqrt{\sigma_{11}}} & \frac{\sigma_{12}}{\sqrt{\sigma_{11}}} & \cdots & \frac{\sigma_{1p}}{\sqrt{\sigma_{11}}} \\ \frac{\sigma_{21}}{\sqrt{\sigma_{22}}} & \frac{\sigma_{22}}{\sqrt{\sigma_{22}}} & \cdots & \frac{\sigma_{2p}}{\sqrt{\sigma_{22}}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\sigma_{p1}}{\sqrt{\sigma_{pp}}} & \frac{\sigma_{p2}}{\sqrt{\sigma_{pp}}} & \cdots & \frac{\sigma_{pp}}{\sqrt{\sigma_{pp}}} \end{bmatrix} \cdot \begin{bmatrix} \frac{1}{\sqrt{\sigma_{11}}} & 0 & \cdots & 0 \\ 0 & \frac{1}{\sqrt{\sigma_{22}}} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{1}{\sqrt{\sigma_{pp}}} \end{bmatrix}$$

$$= \begin{bmatrix} \frac{\sigma_{11}}{\sqrt{\sigma_{11}}\sqrt{\sigma_{11}}} & \frac{\sigma_{12}}{\sqrt{\sigma_{11}}\sqrt{\sigma_{22}}} & \cdots & \frac{\sigma_{1p}}{\sqrt{\sigma_{11}}\sqrt{\sigma_{pp}}} \\ \frac{\sigma_{21}}{\sqrt{\sigma_{22}}\sqrt{\sigma_{11}}} & \frac{\sigma_{22}}{\sqrt{\sigma_{22}}\sqrt{\sigma_{22}}} & \cdots & \frac{\sigma_{2p}}{\sqrt{\sigma_{22}}\sqrt{\sigma_{pp}}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\sigma_{p1}}{\sqrt{\sigma_{pp}}\sqrt{\sigma_{11}}} & \frac{\sigma_{p2}}{\sqrt{\sigma_{pp}}\sqrt{\sigma_{22}}} & \cdots & \frac{\sigma_{pp}}{\sqrt{\sigma_{pp}}\sqrt{\sigma_{pp}}} \end{bmatrix}$$

$$= \begin{bmatrix} \rho_{11} & \rho_{12} & \cdots & \rho_{1p} \\ \rho_{21} & \rho_{22} & \cdots & \rho_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{p1} & \rho_{p2} & \cdots & \rho_{pp} \end{bmatrix}$$
$$= \begin{bmatrix} 1 & \rho_{12} & \cdots & \rho_{1p} \\ \rho_{21} & 1 & \cdots & \rho_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{p1} & \rho_{p2} & \cdots & 1 \\ (p \times p) \end{bmatrix}$$

Thus, $\rho_{(p \times p)}$ can be obtained from $(\mathbf{V}_{(p \times p)}^{1/2})^{-1}$ and $\sum_{\mathbf{X}} \blacksquare$

Corollary 3.2.1. Let $V_{(p \times p)}^{1/2}$ be the population standard deviation matrix and $\rho_{(p \times p)}$ be

the population correlation matrix. Then $\sum_{(p \times p)} the population variance-covariance$

matrix can be obtained. That is,

 $\begin{aligned} \boldsymbol{\rho}_{(p \times p)} &= (\mathbf{V}_{(p \times p)}^{1/2})^{-1} \cdot \sum_{(p \times p)} \cdot (\mathbf{V}_{(p \times p)}^{1/2})^{-1} \\ \mathbf{V}_{(p \times p)}^{1/2} \cdot \boldsymbol{\rho}_{(p \times p)} \cdot \mathbf{V}_{(p \times p)}^{1/2} &= \mathbf{V}_{(p \times p)}^{1/2} \cdot (\mathbf{V}_{(p \times p)}^{1/2})^{-1} \cdot \sum_{(p \times p)} \cdot (\mathbf{V}_{(p \times p)}^{1/2})^{-1} \cdot \mathbf{V}_{(p \times p)}^{1/2} \\ \mathbf{V}_{(p \times p)}^{1/2} \cdot \boldsymbol{\rho}_{(p \times p)} \cdot \mathbf{V}_{(p \times p)}^{1/2} &= \mathbf{I}_{(p \times p)} \cdot \sum_{(p \times p)} \cdot \mathbf{I}_{(p \times p)} \cdot \left\{ (\mathbf{V}_{(p \times p)}^{1/2})^{-1} \text{ inverse of } \mathbf{V}_{(p \times p)}^{1/2} \right\} \\ \sum_{(p \times p)} \mathbf{X}_{(p \times p)}^{1/2} &= \mathbf{V}_{(p \times p)}^{1/2} \cdot \mathbf{\rho}_{(p \times p)} \cdot \mathbf{V}_{(p \times p)}^{1/2} \\ [3, p. 72]. \end{aligned}$

3.3 Population Mean Vector and Variance-Covariance

Matrix for Linear Combinations of Continuous Random

Variables

3.3.1 Linear Combination

Definition 3.3.1 (Linear Combination of **X**). Let $\underset{(p \times 1)}{c}$ be a $p \times 1$ vector of constants

defined as

$$\mathbf{c}_{(p\times 1)} = \begin{bmatrix} c_1\\c_2\\\vdots\\c_p \end{bmatrix}$$

$$(p\times 1)$$

$$\mathbf{X}_{(p\times1)} = \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_p \end{bmatrix}_{(p\times1)}$$

i = 1, 2, ..., p. Then a linear combination of $\underset{(p \times 1)}{\mathbf{X}}$, is given by the inner product

$$\mathbf{c}'_{(1\times p)} \cdot \mathbf{X}_{(p\times 1)} = \begin{bmatrix} c_1, c_2, \dots, c_p \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_p \end{bmatrix} = c_1 X_1 + c_2 X_2 + \dots + c_p X_p$$

[3, p. 76]*.*

3.3.2 Population Parameters for Linear Combinations

Theorem 3.3.1 (Mean of a Linear Combination of **X**). Suppose a linear combination $\mathbf{c}'_{(1\times p)} \cdot \mathbf{X}_{(p\times 1)}$ is given by Definition 3.3.1 and a population mean vector $\boldsymbol{\mu}_{\mathbf{X}} = E(\mathbf{X})$ is given by Definition 3.2.11. Then the **expected value** or **mean of a linear combination** of $\mathbf{X}_{(p\times 1)}$, is given by

$$E\left(\mathbf{c}'_{(1\times p)}\cdot \mathbf{X}_{(p\times 1)}\right) = \mathbf{c}'_{(1\times p)}\cdot \mathbf{\mu}_{\mathbf{X}}_{(p\times 1)}$$

Proof. Using linearity of *E* and Definition 3.2.5.

$$E\left(\mathbf{c}'_{(1\times p)}\cdot\mathbf{X}_{(p\times 1)}\right)$$

$$= E\left(\begin{bmatrix} c_1, c_2, \dots, c_p \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_p \end{bmatrix} \right)$$

= $E(c_1X_1 + c_2X_2 + \dots + c_pX_p)$
= $c_1E(X_1) + c_2E(X_2) + \dots + c_pE(X_p)$
= $c_1\mu_1 + c_2\mu_2 + \dots + c_p\mu_p$
= $\begin{bmatrix} c_1, c_2, \dots, c_p \end{bmatrix} \begin{bmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_p \end{bmatrix}_{(p \times 1)}$
= $\underbrace{\mathbf{c}'_{(1 \times p)} \cdot \underbrace{\boldsymbol{\mu}_X}_{(p \times 1)} \bullet$

Theorem 3.3.2 (Variance of a Linear Combination of X). *Suppose a linear*

combination $\mathbf{c}'_{(1 \times p)} \cdot \mathbf{X}_{(p \times 1)}$ is given by Definition 3.3.1 and a population variance-

covariance $\sum_{(p \times p)} = \operatorname{Cov}(\mathbf{X})$ *is given by Theorem 3.2.1. Then the* **variance of a linear**

combination of $\underset{(p \times 1)'}{\mathbf{X}}$, is given by

$$\operatorname{var}\left(\underset{(1\times p)}{\mathbf{c}'}\cdot\underset{(p\times 1)}{\mathbf{X}}\right) = \underset{(1\times p)}{\mathbf{c}'}\cdot\underset{(p\times p)}{\sum}\cdot\underset{(p\times 1)}{\mathbf{c}} = \sum_{i}^{p}\sum_{k=1}^{p}c_{i}c_{k}\sigma_{ik}$$
$$= \sum_{i=1}^{p}c_{i}^{2}\sigma_{ii} + \underbrace{\sum}_{i\neq k}\sum_{i\neq k}c_{i}c_{k}\sigma_{ik} = \sum_{i=1}^{p}c_{i}^{2}\sigma_{ii} + 2\sum_{i$$

Proof. Using properties of variance and covariance.

$$\begin{aligned} \operatorname{var} \left(\underbrace{\mathbf{c}'_{(1\times p)} \cdot \underbrace{\mathbf{X}}_{(p\times 1)} \right) \\ &= \operatorname{var} \left(\left[c_1, c_2, \dots, c_p \right] \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_p \\ (p\times 1) \end{bmatrix} \right) \\ &= \operatorname{var} \left(c_1 X_1 + c_2 X_2 + \dots + c_p X_p \right) \\ &= \operatorname{cov} \left(c_1 X_1 + c_2 X_2 + \dots + c_p X_p, c_1 X_1 + c_2 X_2 + \dots + c_p X_p \right) \\ &= c_1^2 \operatorname{var} (X_1) + c_1 c_2 \operatorname{cov} (X_1, X_2) + \dots + c_1 c_p \operatorname{cov} (X_1, X_p) \\ &+ c_2 c_1 \operatorname{cov} (X_2, X_1) + c_2^2 \operatorname{var} (X_2) + \dots + c_2 c_p \operatorname{cov} (X_2, X_p) \\ &+ \dots + \\ c_p c_1 \operatorname{cov} (X_p, X_1) + c_p c_2 \operatorname{cov} (X_p, X_2) + \dots + c_p^2 \operatorname{var} (X_p) \\ &= c_1^2 \operatorname{var} (X_1) + c_2^2 \operatorname{var} (X_2) + \dots + c_p^2 \operatorname{var} (X_p) \\ &+ 2c_1 c_2 \operatorname{cov} (X_1, X_2) + \dots + 2c_{p-1} c_p \operatorname{cov} (X_{p-1}, X_p) \\ &= c_1^2 \sigma_{11} + c_2^2 \sigma_{22} + \dots + c_p^2 \sigma_{pp} \\ &+ 2c_1 c_2 \sigma_{12} + \dots + 2c_{p-1} c_p \sigma_{(p-1)(p)} \\ &= \sum_{i=1}^p \sum_{k=1}^p c_i c_k \sigma_{ik} = \sum_{i=1}^p c_i^2 \sigma_{ii} + \sum_{i \neq k} \sum_{i \neq k} c_i c_k \sigma_{ik} \\ &= \sum_{i=1}^p c_i^2 \sigma_{ii} + 2 \sum_{i \leq k} c_i c_k \sigma_{ik} \end{aligned}$$

$$\begin{split} &= \underbrace{\mathbf{c}'_{(1\times p)} \cdot \sum_{(p\times p)} \cdot \underbrace{\mathbf{c}}_{(p\times 1)}}_{(p\times p)} \left[\begin{bmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1p} \\ \sigma_{21} & \sigma_{22} & \cdots & \sigma_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{p1} & \sigma_{p2} & \cdots & \sigma_{pp} \end{bmatrix} \begin{bmatrix} c_{1} \\ c_{2} \\ \vdots \\ c_{p} \end{bmatrix} \\ &= \begin{bmatrix} c_{1}, c_{2}, \dots, c_{p} \end{bmatrix} \begin{bmatrix} c_{1}\sigma_{11} + c_{2}\sigma_{12} + \cdots + c_{p}\sigma_{1p} \\ c_{1}\sigma_{21} + c_{2}\sigma_{22} + \cdots + c_{p}\sigma_{2p} \\ \vdots \\ c_{1}\sigma_{p1} + c_{2}\sigma_{p2} + \cdots + c_{p}\sigma_{pp} \end{bmatrix} \\ &= c_{1}(c_{1}\sigma_{11} + c_{2}\sigma_{12} + \cdots + c_{p}\sigma_{1p}) \\ &+ c_{2}(c_{2}c_{1}\sigma_{21} + c_{2}\sigma_{22} + \cdots + c_{p}\sigma_{2p}) \\ &+ \cdots + \\ c_{p}(c_{1}\sigma_{p1} + c_{2}\sigma_{p2} + \cdots + c_{p}\sigma_{pp}) \\ &= c_{1}^{2}\sigma_{11} + c_{1}c_{2}\sigma_{12} + \cdots + c_{1}c_{p}\sigma_{1p} \\ &+ c_{2}c_{1}\sigma_{21} + c_{2}^{2}\sigma_{22} + \cdots + c_{p}^{2}\sigma_{pp} \\ &+ \cdots + \\ c_{p}c_{1}\sigma_{p1} + c_{p}c_{2}\sigma_{p2} + \cdots + c_{p}^{2}\sigma_{pp} \\ &= c_{1}^{2}\sigma_{11} + c_{2}^{2}\sigma_{22} + \cdots + c_{p}^{2}\sigma_{pp} \\ &= c_{1}^{2}\sigma_{11} + c_{2}^{2}\sigma_{22} + \cdots + c_{p}^{2}\sigma_{pp} \\ &= c_{1}^{2}\sigma_{11} + c_{2}^{2}\sigma_{22} + \cdots + c_{p}^{2}\sigma_{pp} \\ &= c_{1}^{2}\sigma_{11} + c_{2}^{2}\sigma_{22} + \cdots + c_{p}^{2}\sigma_{pp} \\ &= c_{1}^{2}\sigma_{11} + c_{2}^{2}\sigma_{22} + \cdots + c_{p}^{2}\sigma_{pp} \\ &= c_{1}^{2}\sigma_{11} + c_{2}^{2}\sigma_{22} + \cdots + c_{p}^{2}\sigma_{pp} \\ &= c_{1}^{2}\sigma_{11} + c_{2}^{2}\sigma_{22} + \cdots + 2c_{p-1}c_{p}\sigma_{(p-1)(p)} \\ &= \sum_{i=1}^{p}\sum_{k=1}^{p}c_{i}c_{k}\sigma_{ik} = \sum_{i=1}^{p}c_{i}^{2}\sigma_{ii} + \sum_{i \neq k}\sum_{i \neq k}c_{i}c_{k}\sigma_{ik} \\ &= \sum_{i=1}^{p}c_{i}^{2}\sigma_{ii} + 2\sum_{i \leqslant k}c_{i}c_{k}\sigma_{ik} = \sum_{i=1}^{p}c_{i}^{2}\sigma_{ii} + \sum_{i \neq k}\sum_{i \neq k}$$

Theorem 3.3.3 (Covariance of Two Linear Combinations of X). Suppose two linear

combinations $\mathbf{b}'_{(1 \times p)} \cdot \mathbf{X}_{(p \times 1)}$ and $\mathbf{c}'_{(1 \times p)} \cdot \mathbf{X}_{(p \times 1)}$ are given following Definition 3.3.1 and a

population variance-covariance $\sum_{(p \times p)} = Cov(\mathbf{X})$ is given by Theorem 3.2.1.Then the

covariance of two linear combinations of $\underset{(p \times 1)}{\mathbf{X}}$, is given by

$$\operatorname{cov}\left(\underset{(1\times p)}{\mathbf{b}'} \cdot \underset{(p\times 1)'}{\mathbf{X}}, \underset{(1\times p)}{\mathbf{c}'} \cdot \underset{(p\times 1)}{\mathbf{X}}\right) = \underset{(1\times p)}{\mathbf{b}'} \cdot \underset{(p\times p)}{\sum} \cdot \underset{(p\times p)}{\mathbf{c}_k} \cdot \underset{(p\times 1)}{\mathbf{c}_k}$$
$$= \sum_{i=1}^p \sum_{k=1}^p b_i c_k \sigma_{ik} = \sum_{i=1}^p b_i c_i \sigma_{ii} + \underbrace{\sum_{i\neq k} b_i c_k \sigma_{ik}}_{i\neq k}$$

Proof. Using properties of variance and covariance.

$$\operatorname{cov}\left(\underbrace{\mathbf{b}'_{(1\times p)} \cdot \underbrace{\mathbf{X}}_{(p\times 1)'(1\times p)} \cdot \underbrace{\mathbf{X}}_{(p\times 1)}}_{(1\times p)} \underbrace{\mathbf{X}_{1}}_{(1\times p)} \right)$$
$$= \operatorname{cov}\left(\begin{bmatrix}b_{1}, b_{2}, \dots, b_{p}\end{bmatrix} \begin{bmatrix}X_{1}\\X_{2}\\\vdots\\X_{p}\\(1\times p)\end{bmatrix}}, \begin{bmatrix}c_{1}, c_{2}, \dots, c_{p}\end{bmatrix} \begin{bmatrix}X_{1}\\X_{2}\\\vdots\\X_{p}\\(1\times p)\end{bmatrix}}\right)$$
$$= \operatorname{cov}(b_{1}X_{1} + b_{2}X_{2} + \dots + b_{p}X_{p}, c_{1}X_{1} + c_{2}X_{2} + \dots + c_{p}X_{p})$$

$$= b_{1}c_{1}\operatorname{var}(X_{1}) + b_{1}c_{2}\operatorname{cov}(X_{1}, X_{2}) + \dots + b_{1}c_{p}\operatorname{cov}(X_{1}, X_{p}) + b_{2}c_{1}\operatorname{cov}(X_{2}, X_{1}) + b_{2}c_{2}\operatorname{var}(X_{2}) + \dots + b_{2}c_{p}\operatorname{cov}(X_{2}, X_{p}) + \dots + b_{p}c_{1}\operatorname{cov}(X_{p}, X_{1}) + b_{p}c_{2}\operatorname{cov}(X_{p}, X_{2}) + \dots + b_{p}c_{p}\operatorname{var}(X_{p}) = b_{1}c_{1}\sigma_{11} + b_{2}c_{2}\sigma_{22} + \dots + b_{p}c_{p}\sigma_{pp} + b_{1}c_{2}\sigma_{12} + b_{2}c_{1}\sigma_{21} + \dots + b_{p-1}c_{p}\sigma_{(p-1)(p)} + b_{p}c_{p-1}\sigma_{(p)(p-1)} = \sum_{i=1}^{p}\sum_{k=1}^{p}b_{i}c_{k}\sigma_{ik} = \sum_{i=1}^{p}b_{i}c_{i}\sigma_{ii} + \sum_{i \neq k}\sum_{i \neq k}b_{i}c_{k}\sigma_{ik} = \sum_{(1\times p)}^{p}\cdot\sum_{(p\times p)}\cdot\sum_{(p\times 1)}\cdot\sum_{(p\times 1)}c_{p} = \begin{bmatrix}b_{1}, b_{2}, \dots, b_{p}\\(1\times p)\end{bmatrix}\begin{bmatrix}\sigma_{11} & \sigma_{12} & \dots & \sigma_{1p}\\\sigma_{21} & \sigma_{22} & \dots & \sigma_{2p}\\\vdots & \vdots & \ddots & \vdots\\\sigma_{p1} & \sigma_{p2} & \dots & \sigma_{pp}\end{bmatrix}\begin{bmatrix}c_{1}\\c_{2}\\\vdots\\c_{p}\\(p\times 1)\end{bmatrix} \\= \begin{bmatrix}b_{1}, b_{2}, \dots, b_{p}\end{bmatrix}\begin{bmatrix}c_{1}\sigma_{11} + c_{2}\sigma_{12} + \dots + c_{p}\sigma_{1p}\\c_{1}\sigma_{21} + c_{2}\sigma_{22} + \dots + c_{p}\sigma_{2p}\\\vdots\\c_{1}\sigma_{p1} + c_{2}\sigma_{p2} + \dots + c_{p}\sigma_{pp}\end{bmatrix}$$

$$= b_1(c_1\sigma_{11} + c_2\sigma_{12} + \dots + c_p\sigma_{1p}) + b_2(c_1\sigma_{21} + c_2\sigma_{22} + \dots + c_p\sigma_{2p}) + \dots + b_p(c_1\sigma_{p1} + c_2\sigma_{p2} + \dots + c_p\sigma_{pp}) = b_1c_1\sigma_{11} + b_1c_2\sigma_{12} + \dots + b_1c_p\sigma_{1p} + b_2c_1\sigma_{21} + b_2c_2\sigma_{22} + \dots + b_2c_p\sigma_{2p} + \dots + b_pc_1\sigma_{p1} + b_pc_2\sigma_{p2} + \dots + b_pc_p\sigma_{pp} = b_1c_1\sigma_{11} + b_2c_2\sigma_{22} + \dots + b_pc_p\sigma_{pp} + b_1c_2\sigma_{12} + b_2c_1\sigma_{21} + \dots + b_{p-1}c_p\sigma_{(p-1)(p)} + b_pc_{p-1}\sigma_{(p)(p-1)} = \sum_{i=1}^p \sum_{k=1}^p b_ic_k\sigma_{ik} = \sum_{i=1}^p b_ic_i\sigma_{ii} + \sum_{i\neq k} \sum_{i\neq k} b_ic_k\sigma_{ik} \bullet$$

3.3.3 q Linear Combinations

Definition 3.3.2 (*q* Linear Combinations of **X**). Consider $\underset{(q \times p)}{\mathsf{C}}$ a matrix of real

constants and the **q** linear combinations of $\underset{(p \times 1)}{\mathbf{X}}$, Y_i ,

$$Y_{1} = \mathbf{c}_{(1\times p)}' \cdot \mathbf{X}_{(p\times 1)} = c_{11}X_{1} + c_{12}X_{2} + \dots + c_{1p}X_{p}$$
$$Y_{2} = \mathbf{c}_{2}' \cdot \mathbf{X}_{(1\times p)} \cdot c_{(p\times 1)} = c_{21}X_{1} + c_{22}X_{2} + \dots + c_{2p}X_{p}$$
$$\vdots \qquad \vdots$$
$$Y_{q} = \mathbf{c}_{q}' \cdot \mathbf{X}_{(p\times 1)} = c_{q1}X_{1} + c_{q2}X_{2} + \dots + c_{qp}X_{p}$$

or in matrix notation,

$$\mathbf{Y}_{(q\times1)} = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_q \\ (q\times1) \end{bmatrix} = \begin{bmatrix} \mathbf{c}'_1 & \cdot \mathbf{X}_{(1\times p) \quad (p\times1)} \\ \mathbf{c}'_2 & \cdot \mathbf{X}_{(1\times p) \quad (p\times1)} \\ \vdots \\ \mathbf{c}'_q & \cdot \mathbf{X}_{(1\times p) \quad (p\times1)} \\ (q\times1) \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1p} \\ c_{21} & c_{22} & \cdots & c_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ c_{q1} & c_{q2} & \cdots & c_{qp} \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_p \end{bmatrix} = \mathbf{c} \\ \begin{pmatrix} \mathbf{c} \\ (q \times p) \\ (p \times 1) \end{bmatrix}$$

[3, p. 76]*.*
3.3.4 Population Mean Vector for q Linear Combinations

Theorem 3.3.4 (Population Mean Vector for *q* Linear Combinations of **X**). Suppose *q* linear combinations $Y_i = \mathbf{c}'_i \cdot \mathbf{X}_{(p \times 1)}$ are given by Definition 3.3.2 and a population

mean vector $\mu_{\mathbf{X}} = E(\mathbf{X})$ is given by Definition 3.2.11. Then the population mean $(p \times 1)$

vector for *q* linear combinations of $\underset{(p \times 1)}{\mathbf{X}}, \underset{(p \times 1)}{\mathbf{Y}}$, is given by

$$\boldsymbol{\mu}_{\mathbf{Y}} = E(\mathbf{Y}) = E\left(\underbrace{\mathbf{C}}_{(q \times p)} \cdot \underbrace{\mathbf{X}}_{(p \times 1)} \right) = \underbrace{\mathbf{C}}_{(q \times p)} \cdot \underbrace{\boldsymbol{\mu}_{\mathbf{X}}}_{(p \times 1)} = \begin{bmatrix} \mathbf{C}_{1} \cdot \boldsymbol{\mu}_{\mathbf{X}} \\ (1 \times p) \quad (p \times 1) \\ \mathbf{C}_{2}' \cdot \boldsymbol{\mu}_{\mathbf{X}} \\ (1 \times p) \quad (p \times 1) \\ \vdots \\ \mathbf{C}_{q}' \cdot \boldsymbol{\mu}_{\mathbf{X}} \\ (1 \times p) \quad (p \times 1) \end{bmatrix}}_{(q \times 1)}$$

[3, p. 76].

Proof. Using the linearity of *E* and Definition 2.2.5.

 $\mu_{\mathbf{Y}}_{(q \times 1)}$

$$= E(\mathbf{Y}) \\ = E\left(\begin{pmatrix} \mathbf{C} & \mathbf{X} \\ (q \times p) & (p \times 1) \end{pmatrix} \right) \\ = E\left(\begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1p} \\ c_{21} & c_{22} & \cdots & c_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ c_{q1} & c_{q2} & \cdots & c_{qp} \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_p \\ (p \times 1) \end{pmatrix} \right)$$

$$= \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1p} \\ c_{21} & c_{22} & \cdots & c_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ c_{q1} & c_{q2} & \cdots & c_{qp} \end{bmatrix} E \begin{pmatrix} \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_p \\ p \times 1 \end{pmatrix}$$
$$= \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1p} \\ c_{21} & c_{22} & \cdots & c_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ c_{q1} & c_{q2} & \cdots & c_{qp} \end{bmatrix} \begin{bmatrix} E(X_1) \\ E(X_2) \\ \vdots \\ E(X_p) \end{bmatrix}$$
$$= \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1p} \\ c_{21} & c_{22} & \cdots & c_{qp} \\ \vdots & \vdots & \ddots & \vdots \\ c_{q1} & c_{q2} & \cdots & c_{qp} \end{bmatrix} \begin{bmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_p \\ (q \times p) \end{bmatrix}$$
$$= \begin{bmatrix} c_{11} \mu_1 + c_{12}\mu_2 + \cdots + c_{1p}\mu_p \\ c_{21}\mu_1 + c_{22}\mu_2 + \cdots + c_{2p}\mu_p \\ \vdots \\ c_{q1}\mu_1 + c_{q2}\mu_2 + \cdots + c_{qp}\mu_p \end{bmatrix}$$
$$= \begin{bmatrix} c_1' & \mu_X \\ (1 \times p) & (p \times 1) \\ c_2' & \mu_X \\ (1 \times p) & (p \times 1) \\ \vdots \\ c_q' & \cdot \mu_X \\ (1 \times p) & (p \times 1) \\ \vdots \\ (q \times 1) \end{bmatrix}$$

Thus, the *i*th row of $\mathbf{Y}_{(q \times 1)}$ has population mean

$$\bar{Y}_i = E(Y_i) = E\left(\mathbf{c}'_i \cdot \mathbf{X}_{(1 \times p)}\right) = \mathbf{c}'_i \cdot \mathbf{\mu}_{\mathbf{X}}_{(1 \times p)}$$

for i = 1, 2, ..., q.

3.3.5 Population Variance-Covariance Matrix for q Linear

Combinations

Theorem 3.3.5. (Population Variance-Covariance Matrix for *q* Linear Combinations of **X**). Suppose *q* linear combinations $Y_i = \underset{(1 \times p)}{\mathbf{c}'_{(1 \times p)}} \cdot \underset{(p \times 1)}{\mathbf{X}}$ are given by Definition 3.3.2 and a population variance-covariance $\sum_{(p \times p)} = \underset{(p \times p)}{\operatorname{Cov}(\mathbf{X})}$ is given by Theorem 3.2.1.

Then the symmetric population variance-covariance matrix for q linear

combinations of $X_{(p\times 1)}, Y_{(q\times 1)}$, is given by

$$\sum_{(q \times q)} = \operatorname{Cov}(\mathbf{Y}) = \mathbf{C}_{(q \times p)} \cdot \sum_{(p \times p)} \cdot \mathbf{C}'_{(p \times q)}$$

[3, p. 76]*.*

Proof. Using Definition 2.2.5 for matrix multiplication and following Theorem 3.3.2 for computation of diagonal elements and Theorem 3.3.3 for computation of off-diagonal elements.

 $\sum_{(q \times q)}$

$$= \operatorname{Cov}(\mathbf{Y})_{(q \times q)}$$

$$= \underbrace{\mathbf{C}}_{(q \times p)} \cdot \underbrace{\sum_{(p \times p)} \cdot \mathbf{C}'_{(p \times q)}}_{(p \times q)}$$

$$= \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1p} \\ c_{21} & c_{22} & \cdots & c_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ c_{q1} & c_{q2} & \cdots & c_{qp} \end{bmatrix} \begin{bmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1p} \\ \sigma_{21} & \sigma_{22} & \cdots & \sigma_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{p1} & \sigma_{p2} & \cdots & \sigma_{pp} \end{bmatrix} \begin{bmatrix} c_{11} & c_{21} & \cdots & c_{q1} \\ c_{12} & c_{22} & \cdots & c_{q2} \\ \vdots & \vdots & \ddots & \vdots \\ c_{1p} & c_{2p} & \cdots & c_{qp} \end{bmatrix}$$

$$= \begin{bmatrix} \mathbf{c}'_{1} \cdot \sum_{\mathbf{X}} \cdot \mathbf{c}_{1} & \mathbf{c}'_{1} \cdot \sum_{\mathbf{X}} \cdot \mathbf{c}_{2} & \cdots & \mathbf{c}'_{1} \cdot \sum_{\mathbf{X}} \cdot \mathbf{c}_{q} \\ (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) \\ \mathbf{c}'_{2} \cdot \sum_{\mathbf{X}} \cdot \mathbf{c}_{1} & \mathbf{c}'_{2} \cdot \sum_{\mathbf{X}} \cdot \mathbf{c}_{2} & \cdots & \mathbf{c}'_{2} \cdot \sum_{\mathbf{X}} \cdot \mathbf{c}_{q} \\ (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{c}'_{q} \cdot \sum_{\mathbf{X}} \cdot \mathbf{c}_{1} & \mathbf{c}'_{q} \cdot \sum_{\mathbf{X}} \cdot \mathbf{c}_{2} & \cdots & \mathbf{c}'_{q} \cdot \sum_{\mathbf{X}} \cdot \mathbf{c}_{q} \\ (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{c}'_{q} \cdot \sum_{\mathbf{X}} \cdot \mathbf{c}_{1} & \mathbf{c}'_{q} \cdot \sum_{\mathbf{X}} \cdot \mathbf{c}_{2} & \cdots & \mathbf{c}'_{q} \cdot \sum_{\mathbf{X}} \cdot \mathbf{c}_{q} \\ (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) \end{bmatrix}$$

Thus, the *i*th row of $\mathbf{Y}_{(q \times 1)}$ has population variance

$$\operatorname{var}(Y_i) = \frac{\mathbf{c}'_i}{(1 \times p)} \cdot \frac{\sum_{\mathbf{X}} \cdot \mathbf{c}_i}{(p \times p)} \cdot \frac{\mathbf{c}_i}{(p \times 1)}$$

for i = 1, 2, ..., q.

And the *i*th row and *k*th row of $\underset{(q \times 1)}{\mathbf{Y}}$ have population covariance

$$\operatorname{cov}(Y_i, Y_k) = \frac{\mathbf{c}'_i}{(1 \times p)} \cdot \frac{\sum_{\mathbf{X}}}{(p \times p)} \cdot \frac{\mathbf{c}_k}{(p \times 1)} = \frac{\mathbf{c}'_k}{(1 \times p)} \cdot \frac{\sum_{\mathbf{X}}}{(p \times p)} \cdot \frac{\mathbf{c}_i}{(p \times 1)}$$

for i, k = 1, 2, ..., q.

3.4 Population Random Vector, Mean Vector, and Variance-Covariance Matrix for Standardized Continuous Random Variables

3.4.1 Population Random Vector for Standardized Continuous Random Variables

Definition 3.4.1 (Population Random Vector **Z**). *A population random vector for standardized continuous variables is a vector whose elements are standardized population continuous random variables from a* p - variate *population. Each standardized continuous random variable is of the form*

$$Z_i = \frac{X_i - \mu_i}{\sqrt{\sigma_{ii}}}$$

for i = 1, 2, ..., p.

Specifically, let the **population random vector** $\sum_{(p \times 1)} = \{Z_i\}$ *be* defined by

$$\mathbf{Z}_{(p\times1)} = \mathbf{V}_{(p\timesp)}^{-1/2} \cdot \left(\mathbf{X}_{(p\times1)} - \boldsymbol{\mu}_{\mathbf{X}}_{(p\times1)} \right) = \begin{bmatrix} Z_1 \\ Z_2 \\ \vdots \\ Z_p \end{bmatrix} = \begin{bmatrix} \frac{X_1 - \mu_1}{\sqrt{\sigma_{11}}} \\ \frac{X_2 - \mu_2}{\sqrt{\sigma_{22}}} \\ \vdots \\ \frac{X_p - \mu_p}{\sqrt{\sigma_{pp}}} \end{bmatrix}$$

where $\mathbf{X}_{(p \times 1)}$ is a population random vector defined in Definition 3.2.1, $\boldsymbol{\mu}_{\mathbf{X}}$ is a $(p \times 1)$

population mean vector defined in Definition 3.2.11., and $\mathbf{V}_{(p \times p)}^{-1/2}$ is an inverse

population standard deviation matrix defined in Definition 3.2.12.

$$\begin{split} \mathbf{Z}_{(p\times1)} &= \mathbf{V}_{(p\timesp)}^{-1/2} \cdot \begin{pmatrix} \mathbf{X}_{(p\times1)} - \boldsymbol{\mu}_{\mathbf{X}} \\ (p\times1) - \begin{pmatrix} \mathbf{\chi}_{p\times1} \end{pmatrix} \end{pmatrix} \\ &= \begin{bmatrix} \frac{1}{\sqrt{\sigma_{11}}} & 0 & \cdots & 0 \\ 0 & \frac{1}{\sqrt{\sigma_{22}}} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{1}{\sqrt{\sigma_{pp}}} \end{bmatrix} \cdot \begin{pmatrix} \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_p \\ (p\times1) \end{pmatrix} - \begin{bmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_p \\ (p\times1) \end{pmatrix} \\ & \begin{pmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_p \\ (p\times1) \end{pmatrix} \end{pmatrix} \\ &= \begin{bmatrix} \frac{1}{\sqrt{\sigma_{11}}} & 0 & \cdots & 0 \\ 0 & \frac{1}{\sqrt{\sigma_{22}}} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{1}{\sqrt{\sigma_{pp}}} \end{bmatrix} \cdot \begin{pmatrix} \begin{bmatrix} X_1 - \mu_1 \\ X_2 - \mu_2 \\ \vdots \\ X_p - \mu_p \\ (p\times1) \end{pmatrix} \end{pmatrix} = \begin{bmatrix} \frac{X_1 - \mu_1}{\sqrt{\sigma_{11}}} \\ \frac{X_2 - \mu_2}{\sqrt{\sigma_{22}}} \\ \vdots \\ \frac{X_p - \mu_p}{\sqrt{\sigma_{pp}}} \end{bmatrix} = \begin{bmatrix} Z_1 \\ Z_2 \\ \vdots \\ Z_p \\ (p\times1) \end{pmatrix} \end{split}$$

[3, pp. 436-437].

 $(p \times p)$

3.4.2 Population Parameters for Standardized Continuous Random Variables

Theorem 3.4.1 (Univariate Marginal Population Mean for Z_i). Suppose the univariate marginal population means $\mu_i = E(X_i)$ are given by Definition 3.2.5 and univariate marginal population standard deviations $\sqrt{\sigma_{ii}}$ are given by Definition 3.2.6. Then the univariate marginal population means for Z_i are given by

$$\mu_{z,i} = E(Z_i) = E\left(\frac{X_i - \mu_i}{\sqrt{\sigma_{ii}}}\right) = 0$$

for i = 1, 2, ..., p.

Proof. Using linearity of *E*.

 $\mu_{z,i}$

$$= E(Z_i)$$

$$= E\left(\frac{X_i - \mu_i}{\sqrt{\sigma_{ii}}}\right)$$

$$= \frac{1}{\sqrt{\sigma_{ii}}} E(X_i - \mu_i)$$

$$= \frac{1}{\sqrt{\sigma_{ii}}} [E(X_i) - \mu_i]$$

$$= \frac{1}{\sqrt{\sigma_{ii}}} [\mu_i - \mu_i]$$

$$= 0 \bullet$$

Theorem 3.4.2 (Univariate Marginal Population Variance for Z_i). Suppose the univariate marginal population means $\mu_i = E(X_i)$ are given by Definition 3.2.5 and univariate marginal population standard deviations $\sqrt{\sigma_{ii}}$ are given by Definition

3.2.6. Then the **univariate marginal population variances for** Z_i *are given by*

$$\sigma_{z,ii} = E(Z_i - \mu_{z,i})^2 = \operatorname{var}(Z_i) = \operatorname{var}\left(\frac{X_i - \mu_i}{\sqrt{\sigma_{ii}}}\right) = 1$$

for i = 1, 2, ..., p.

Proof. Using properties of variance and covariance.

$$\sigma_{z,ii}$$

$$= E(Z_i - \mu_{z,i})^2$$

$$= \operatorname{var}(Z_i)$$

$$= \operatorname{var}\left(\frac{X_i - \mu_i}{\sqrt{\sigma_{ii}}}\right)$$

$$= \frac{1}{\sigma_{ii}}\operatorname{var}(X_i - \mu_i)$$

$$= \frac{1}{\sigma_{ii}}\operatorname{cov}(X_i - \mu_i, X_i - \mu_i)$$

$$= \frac{1}{\sigma_{ii}}[\operatorname{cov}(X_i, X_i) - \operatorname{cov}(X_i, \mu_i) - \operatorname{cov}(\mu_i, X_i) + \operatorname{cov}(\mu_i, \mu_i)]$$

$$= \frac{1}{\sigma_{ii}}\operatorname{cov}(X_i, X_i) = \frac{1}{\sigma_{ii}}\operatorname{var}(X_i)$$

$$= \frac{\sigma_{ii}}{\sigma_{ii}} = 1 \blacksquare$$

Theorem 3.4.3 (Bivariate Marginal Population Covariance for Z_i and Z_k). Suppose the univariate marginal population means $\mu_i = E(X_i)$ are given by Definition 3.2.5 and univariate marginal population standard deviations $\sqrt{\sigma_{ii}}$ are given by Definition 3.2.6. Then the **bivariate marginal population covariances for** Z_i and Z_k are given by

 $(Y_1 - u_1, Y_2 - u_2)$

$$\sigma_{z,ik} = E(Z_i - \mu_{z,i})(Z_k - \mu_{z,k}) = \operatorname{cov}(Z_i, Z_k) = \operatorname{cov}\left(\frac{X_i - \mu_i}{\sqrt{\sigma_{ii}}}, \frac{X_k - \mu_k}{\sqrt{\sigma_{kk}}}\right) = \rho_{ik}$$

for i, k = 1, 2, ..., p.

Proof. Using properties of covariance and Definition 3.2.8.

$$\sigma_{z,ik}$$

$$= E(Z_i - \mu_{z,i})(Z_k - \mu_{z,k})$$

$$= \operatorname{cov}(Z_i, Z_k)$$

$$= \operatorname{cov}\left(\frac{X_i - \mu_i}{\sqrt{\sigma_{ii}}}, \frac{X_k - \mu_k}{\sqrt{\sigma_{kk}}}\right)$$

$$= \frac{1}{\sqrt{\sigma_{ii}}\sqrt{\sigma_{kk}}} \operatorname{cov}(X_i - \mu_i, X_k - \mu_k)$$

$$= \frac{1}{\sqrt{\sigma_{ii}}\sqrt{\sigma_{kk}}} [\operatorname{cov}(X_i, X_k) - \operatorname{cov}(X_i, \mu_k) - \operatorname{cov}(\mu_i, X_k) + \operatorname{cov}(\mu_i, \mu_k)]$$

$$= \frac{1}{\sqrt{\sigma_{ii}}\sqrt{\sigma_{kk}}} \operatorname{cov}(X_i, X_k)$$

$$= \frac{\sigma_{ik}}{\sqrt{\sigma_{ii}}\sqrt{\sigma_{kk}}} = \operatorname{corr}(X_i, X_k) = \rho_{ik} \blacksquare$$

Thus, standardizing population continuous random variables turns bivariate marginal population covariances $\sigma_{z,ik}$ into bivariate marginal population correlations ρ_{ik} . That is, $\sigma_{z,ik} = \rho_{ik}$ for i, k = 1, 2, ..., p. If X_i, X_k are statistically independent, then $\sigma_{z,ik} = \rho_{ik} = 0$. Note $\sigma_{z,ik} = \sigma_{z,ki}$, and when $i = k, \sigma_{z,ii} = \rho_{ii} = 1$.

3.4.3 Population Mean Vector for Standardized Continuous

Random Variables

Definition 3.4.2 (Population Mean Vector for **Z**). *The* **population mean vector for** $\mathbf{Z}_{(p \times 1)}$ or expected value of $\mathbf{Z}_{(p \times 1)}$ is a random vector consisting of the univariate marginal expectations of each of its standardized elements. Then the population mean vector for $\mathbf{Z}_{(p \times 1)}$ or expected value of $\mathbf{Z}_{(p \times 1)}$ denoted by $\boldsymbol{\mu}_{\mathbf{Z}} = E(\mathbf{Z})$, is the $p \times 1$ vector

$$\boldsymbol{\mu}_{\mathbf{Z}} = E(\mathbf{Z}) = \begin{bmatrix} E(Z_1) \\ E(Z_2) \\ \vdots \\ E(Z_p) \\ (p \times 1) \end{bmatrix} = \begin{bmatrix} \mu_{z,1} \\ \mu_{z,2} \\ \vdots \\ \mu_{z,p} \\ (p \times 1) \end{bmatrix} = \begin{bmatrix} 0_{z,1} \\ 0_{z,2} \\ \vdots \\ 0_{z,p} \\ (p \times 1) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ (p \times 1) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ (p \times 1) \end{bmatrix}$$

Thus, the population mean vector for $\sum_{(p \times 1)} is$ the $\sum_{(p \times 1)} - vector [3, p. 437]$.

3.4.4 Population Variance-Covariance Matrix for Standardized

Continuous Random Variables

Theorem 3.4.4 (Population Variance-Covariance Matrix for **Z**). *The* **population variance-covariance matrix for** $\sum_{(p \times 1)} is a symmetric matrix containing the p$ *univariate marginal population variances* $\sigma_{z,ii} = 1$ *and the* p(p - 1)/2 *distinct bivariate marginal population covariances* $\sigma_{z,ik} = \rho_{ik}$ (i < k). *Then, if these variances and covariances exist, the* $p \times p$ *population variance-covariance matrix for* $\sum_{(p \times 1)} is given by$

$$\sum_{(p \times p)} = \operatorname{Cov}(\mathbf{Z}) = E(\mathbf{Z} - \boldsymbol{\mu}_{\mathbf{Z}})(\mathbf{Z} - \boldsymbol{\mu}_{\mathbf{Z}})' = \boldsymbol{\rho}_{(p \times p)}$$

where $\mu_{\mathbf{Z}} = E(\mathbf{Z}) = \mathbf{0}_{(p \times 1)}$ is the population mean vector for $\mathbf{Z}_{(p \times 1)}$ [3, p. 437].

Proof. Use linearity of the operator *E*, Definition 2.1.2, 2.1.11, and 2.2.5, Theorem 3.4.2 and Theorem 3.4.3.

$$\sum_{\substack{(p \times p) \\ (p \times p)}} \sum_{\substack{(p \times p) \\ (p \times p)}} E(\mathbf{Z} - \boldsymbol{\mu}_{\mathbf{Z}}) (\mathbf{Z} - \boldsymbol{\mu}_{\mathbf{Z}})' \\ = E\left(\begin{bmatrix} Z_1 - \boldsymbol{\mu}_{Z,1} \\ Z_2 - \boldsymbol{\mu}_{Z,2} \\ \vdots \\ Z_p - \boldsymbol{\mu}_{Z,p} \end{bmatrix} \cdot \begin{bmatrix} Z_1 - \boldsymbol{\mu}_{Z,1}, Z_2 - \boldsymbol{\mu}_{Z,2}, \dots, Z_p - \boldsymbol{\mu}_{Z,p} \end{bmatrix} \right)$$

$$= E \begin{pmatrix} (Z_1 - \mu_{z,1})^2 & (Z_1 - \mu_{z,1})(Z_2 - \mu_{z,2}) & \cdots & (Z_1 - \mu_{z,1})(Z_p - \mu_{z,p}) \\ (Z_2 - \mu_{z,2})(Z_1 - \mu_{z,1}) & (Z_2 - \mu_{z,2})^2 & \cdots & (Z_2 - \mu_{z,2})(Z_p - \mu_{z,p}) \\ \vdots & \vdots & \ddots & \vdots \\ (Z_p - \mu_{z,p})(Z_1 - \mu_{z,1}) & (Z_p - \mu_{z,p})(Z_2 - \mu_{z,2}) & \cdots & (Z_p - \mu_{z,p})^2 \end{pmatrix} \end{pmatrix}$$

$$= E \begin{bmatrix} (Z_1 - \mu_{z,1})^2 & (Z_1 - \mu_{z,1})(Z_2 - \mu_{z,2}) & \cdots & (Z_1 - \mu_{z,1})(Z_p - \mu_{z,p}) \\ (Z_2 - \mu_{z,2})(Z_1 - \mu_{z,1}) & (Z_2 - \mu_{z,2})^2 & \cdots & (Z_2 - \mu_{z,2})(Z_p - \mu_{z,p}) \\ \vdots & \vdots & \ddots & \vdots \\ (Z_p - \mu_{z,p})(Z_1 - \mu_{z,1}) & (Z_p - \mu_{z,p})(Z_2 - \mu_{z,2}) & \cdots & (Z_p - \mu_{z,p})^2 \end{bmatrix}$$

$$= \begin{bmatrix} E(Z_1 - \mu_{z,1})^2 & E(Z_1 - \mu_{z,1})(Z_2 - \mu_{z,2}) & \cdots & E(Z_1 - \mu_{z,1})(Z_p - \mu_{z,p}) \\ E(Z_2 - \mu_{z,2})(Z_1 - \mu_{z,1}) & E(Z_2 - \mu_{z,2})^2 & \cdots & E(Z_2 - \mu_{z,2})(Z_p - \mu_{z,p}) \\ \vdots & \vdots & \ddots & \vdots \\ E(Z_p - \mu_{z,p})(Z_1 - \mu_{z,1}) & E(Z_p - \mu_{z,p})(Z_2 - \mu_{z,2}) & \cdots & E(Z_p - \mu_{z,p})^2 \end{bmatrix}$$

$$= \begin{bmatrix} \sigma_{z,11} & \sigma_{z,12} & \cdots & \sigma_{z,1p} \\ \sigma_{z,21} & \sigma_{z,22} & \cdots & \sigma_{z,2p} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{z,p1} & \sigma_{z,p2} & \cdots & \sigma_{z,pp} \end{bmatrix} = \begin{bmatrix} \frac{\sigma_{11}}{\sqrt{\sigma_{11}}\sqrt{\sigma_{11}}} & \frac{\sigma_{12}}{\sqrt{\sigma_{11}}\sqrt{\sigma_{22}}} & \cdots & \frac{\sigma_{1p}}{\sqrt{\sigma_{11}}\sqrt{\sigma_{pp}}} \\ \frac{\sigma_{21}}{\sqrt{\sigma_{22}}\sqrt{\sigma_{11}}} & \frac{\sigma_{22}}{\sqrt{\sigma_{22}}\sqrt{\sigma_{22}}} & \cdots & \frac{\sigma_{2p}}{\sqrt{\sigma_{22}}\sqrt{\sigma_{pp}}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\sigma_{p1}}{\sqrt{\sigma_{pp}}\sqrt{\sigma_{11}}} & \frac{\sigma_{p2}}{\sqrt{\sigma_{pp}}\sqrt{\sigma_{22}}} & \cdots & \frac{\sigma_{pp}}{\sqrt{\sigma_{pp}}\sqrt{\sigma_{pp}}} \end{bmatrix}$$

$$= \begin{bmatrix} \rho_{11} & \rho_{12} & \cdots & \rho_{1p} \\ \rho_{21} & \rho_{22} & \cdots & \rho_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{p1} & \rho_{p2} & \cdots & \rho_{pp} \end{bmatrix} = \begin{bmatrix} 1 & \rho_{12} & \cdots & \rho_{1p} \\ \rho_{21} & 1 & \cdots & \rho_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{p1} & \rho_{p2} & \cdots & 1 \end{bmatrix} = \underset{(p \times p)}{\boldsymbol{\rho}}$$

Hence, the population variance-covariance matrix for $\sum_{(p \times 1)} \mathbf{Z}$ is equal to the population

correlation matrix of $\underset{(p \times 1)}{\mathbf{X}}$. That is, $\underset{(p \times p)}{\sum_{\mathbf{Z}}} = \frac{\boldsymbol{\rho}}{(p \times p)} \blacksquare$

3.5 Mean Vector and Variance-Covariance Matrix for Linear Combinations of Standardized Continuous Random Variables

3.5.1 Linear Combination of Standardized Continuous Random Variables

Definition 3.5.1 (Linear Combination of **Z**). Let $\underset{(p \times 1)}{\mathbf{c}}$ be a $p \times 1$ vector of constants defined as

$$\mathbf{c}_{(p\times1)} = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_p \end{bmatrix}_{(p\times1)}$$

and let $\mathbf{Z}_{(p \times 1)}$ be a $p \times 1$ population random vector of standardized continuous random variables

$$\mathbf{Z}_{(p\times 1)} = \begin{bmatrix} Z_1 \\ Z_2 \\ \vdots \\ Z_p \end{bmatrix}_{(p\times 1)}$$

Then a **linear combination of** $\sum_{(p \times 1)}$, *p standardized random variables, is given by the inner product*

$$\mathbf{c}_{(1\times p)}^{\prime} \cdot \mathbf{Z}_{(p\times 1)} = \begin{bmatrix} c_1, c_2, \dots, c_p \end{bmatrix} \begin{bmatrix} Z_1 \\ Z_2 \\ \vdots \\ Z_p \end{bmatrix} = c_1 Z_1 + c_2 Z_2 + \dots + c_p Z_p.$$

3.5.2 Population Parameters for Linear Combinations of

Standardized Continuous Random Variables

Theorem 3.5.1 (Mean of a Linear Combination of **Z**). Suppose a linear combination of $\underset{(p\times1)'}{\mathbf{Z}}$, $\underset{(p\times1)}{\mathbf{C}'}$, $\underset{(p\times1)}{\mathbf{Z}}$, is given by Definition 3.5.1 and a population mean vector of $\underset{(p\times1)'}{\mathbf{Z}}$, $\underset{(p\times1)}{\boldsymbol{\mu}_{\mathbf{Z}}} = \underset{(p\times1)}{E}(\underset{(p\times1)}{\mathbf{Z}}) = \underset{(p\times1)}{\mathbf{0}}$, is given by Definition 3.4.1. Then the **expected value** or

mean of a linear combination of $\sum_{(p \times 1)}$, is given by

$$E\left(\mathbf{c}'_{(1\times p)}\cdot \mathbf{Z}_{(p\times 1)}\right) = \mathbf{c}'_{(1\times p)}\cdot \mathbf{\mu}_{\mathbf{Z}}_{(p\times 1)} = 0$$

Proof. Using linearity of *E* and Theorem 3.4.1.

$$E\left(\begin{array}{c} \mathbf{c}'_{(1\times p)} \cdot \mathbf{Z}_{(p\times 1)} \right)$$

$$= E\left(\begin{bmatrix} c_1, c_2, \dots, c_p \end{bmatrix} \begin{bmatrix} Z_1 \\ Z_2 \\ \vdots \\ Z_p \end{bmatrix} \right)$$

$$= E\left(c_1 Z_1 + c_2 Z_2 + \dots + c_p Z_p \right)$$

$$= c_1 E(Z_1) + c_2 E(Z_2) + \dots + c_p E(Z_p)$$

$$= c_1 \mu_{z,1} + c_2 \mu_{z,2} + \dots + c_p \mu_{z,p}$$

$$= \begin{bmatrix} c_1, c_2, \dots, c_p \end{bmatrix} \begin{bmatrix} \mu_{z,1} \\ \mu_{z,2} \\ \vdots \\ \mu_{z,p} \end{bmatrix}$$
$$= \underbrace{\mathbf{c}'}_{(1 \times p)} \cdot \underbrace{\mathbf{\mu}_{\mathbf{Z}}}_{(p \times 1)}$$
$$= \underbrace{\mathbf{c}'}_{(1 \times p)} \cdot \underbrace{\mathbf{0}}_{(p \times 1)}$$
$$= \mathbf{0} \blacksquare$$

Theorem 3.5.2 (Variance of a Linear Combination of **Z**). Suppose a linear combination of $\underset{(p\times1)}{\mathbf{Z}}$, $\underset{(p\times1)}{\mathbf{C}'}$, $\underset{(p\times1)}{\mathbf{Z}}$, is given by Definition 3.5.1 and a population variance-covariance of $\underset{(p\times1)}{\mathbf{Z}}$, $\underset{(p\timesp)}{\sum} = \underset{(p\timesp)}{\boldsymbol{\rho}}$, is given by Theorem 3.4.4. Then the

variance of a linear combination of $\underset{(p \times 1)}{\mathbf{Z}}$, is given by

$$\operatorname{var}\left(\underset{(1\times p)}{\mathbf{c}} \cdot \underset{(p\times 1)}{\mathbf{Z}}\right) = \underset{(1\times p)}{\mathbf{c}} \cdot \underset{(p\times p)}{\sum} \cdot \underset{(p\times p)}{\mathbf{c}} \cdot \underset{(p\times 1)}{\mathbf{c}} = \sum_{i}^{p} \sum_{k=1}^{p} c_{i}c_{k}\sigma_{z,ik}$$
$$= \sum_{i=1}^{p} c_{i}^{2}\sigma_{z,ii} + \sum_{i\neq k} \sum_{i\neq k} c_{i}c_{k}\sigma_{z,ik} = \sum_{i=1}^{p} c_{i}^{2}\sigma_{z,ii} + 2\sum_{i
$$= \underset{(1\times p)}{\mathbf{c}} \cdot \underset{(p\times p)}{\mathbf{\rho}} \cdot \underset{(p\times 1)}{\mathbf{c}} = \sum_{i}^{p} \sum_{k=1}^{p} c_{i}c_{k}\rho_{ik}$$
$$= \sum_{i=1}^{p} c_{i}^{2}\rho_{ii} + \underbrace{\sum_{i\neq k} c_{i}c_{k}\rho_{ik}}_{i\neq k} = pc_{i}^{2} + 2\sum_{i$$$$

Proof. Follows directly from Theorem 3.3.2, Theorem 3.4.3, and Theorem 3.4.4 **Theorem 3.5.3** (Covariance of Two Linear Combinations of **Z**). *Suppose two linear combinations of* $\sum_{(p\times 1)'} \sum_{(1\times p)} \sum_{(p\times 1)} \sum_{(p\times 1)} \sum_{(p\times 1)'} \sum_{(p\times p)} and \sum_{(p\times 1)'} \sum_{(p\times p)} \sum_{(p\times$

3.4.4.Then the covariance of two linear combinations of $\underset{(p \times 1)}{\mathbf{Z}}$, is given by

$$\operatorname{cov}\left(\underbrace{\mathbf{b}_{(1\times p)}' \cdot \underbrace{\mathbf{Z}}_{(p\times 1)'(1\times p)} \cdot \underbrace{\mathbf{Z}}_{(p\times 1)}\right) = \underbrace{\mathbf{b}_{(1\times p)}' \cdot \underbrace{\sum_{(p\times p)} \cdot \underbrace{\mathbf{c}}_{(p\times 1)}}_{(p\times p)} \cdot \underbrace{\mathbf{c}}_{(p\times 1)} = \sum_{i=1}^{p} \sum_{k=1}^{p} b_{i}c_{k}\sigma_{z,ik}$$
$$= \sum_{i=1}^{p} \sum_{k=1}^{p} b_{i}c_{k}\sigma_{z,ik} = pb_{i}c_{i} + \underbrace{\sum_{i\neq k} \sum_{i\neq k} b_{i}c_{k}\sigma_{z,ik}}_{i\neq k}$$
$$= pb_{i}c_{i} + \underbrace{\sum_{i\neq k} \sum_{i\neq k} b_{i}c_{k}\rho_{ik}}_{i\neq k} = \sum_{i=1}^{p} \sum_{k=1}^{p} b_{i}c_{k}\rho_{ik}$$
$$= \underbrace{\mathbf{b}_{(1\times p)}' \cdot \underbrace{\mathbf{\rho}}_{(p\times p)} \cdot \underbrace{\mathbf{c}}_{(p\times 1)}}_{(p\times p)} \cdot \underbrace{\mathbf{c}}_{(p\times 1)}.$$

Proof. Follows directly from Theorem 3.3.3, Theorem 3.4.3, and Theorem 3.4.4 ■

3.5.3 q Linear Combinations of Standardized Continuous

Random Variables

Definition 3.5.2 (*q* Linear Combinations of **Z**). Consider $\underset{(q \times p)}{\mathsf{C}}$ a matrix of real

constants and the **q** linear combinations of $\sum_{(p \times 1)} Y_i$,

$$Y_{1} = \underset{(1\times p)}{\mathbf{c}'_{1}} \cdot \underset{(p\times 1)}{\mathbf{Z}} = c_{11}Z_{1} + c_{12}Z_{2} + \dots + c_{1p}Z_{p}$$
$$Y_{2} = \underset{(1\times p)}{\mathbf{c}'_{2}} \cdot \underset{(p\times 1)}{\mathbf{Z}} = c_{21}Z_{1} + c_{22}Z_{2} + \dots + c_{2p}Z_{p}$$
$$\vdots \qquad \vdots$$
$$Y_{q} = \underset{(1\times p)}{\mathbf{c}'_{q}} \cdot \underset{(p\times 1)}{\mathbf{Z}} = c_{q1}Z_{1} + c_{q2}Z_{2} + \dots + c_{qp}Z_{p}$$

or in matrix notation,

$$\mathbf{Y}_{(q\times1)} = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_q \\ (q\times1) \end{bmatrix} = \begin{bmatrix} \mathbf{c}'_1 & \cdot \mathbf{Z}_{(1\times p) \quad (p\times1)} \\ \mathbf{c}'_2 & \cdot \mathbf{Z}_{(1\times p) \quad (p\times1)} \\ \vdots \\ \mathbf{c}'_q & \cdot \mathbf{Z}_{(1\times p) \quad (p\times1)} \\ (q\times1) \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1p} \\ c_{21} & c_{22} & \cdots & c_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ c_{q1} & c_{q2} & \cdots & c_{qp} \end{bmatrix} \begin{bmatrix} Z_1 \\ Z_2 \\ \vdots \\ Z_p \end{bmatrix} = \mathbf{c}_{(q\times p) \quad (p\times1)} \mathbf{c}_{(q\times p)} \mathbf{c}_{(p\times1)} \mathbf{c}_{(p$$

3.5.4 Population Mean Vector for q Linear Combinations of

Standardized Continuous Random Variables

Theorem 3.5.4 (Population Mean Vector for *q* Linear Combinations of **Z**). *Suppose q*

linear combinations of $\mathbf{Z}_{(p \times 1)}$, $Y_i = \mathbf{c}'_i \cdot \mathbf{Z}_{(p \times 1)}$ are given by Definition 3.5.2 and a

population mean vector of $\mathbf{Z}_{(p \times 1)}$, $\boldsymbol{\mu}_{\mathbf{Z}} = E(\mathbf{Z}) = \mathbf{0}_{(p \times 1)}$ is given by Definition

3.4.1.Then the population mean vector for q linear combinations of $\sum_{(p \times 1)'} \sum_{(p \times 1)'} Y$, is

given by

$$\boldsymbol{\mu}_{\mathbf{Y}} = E(\mathbf{Y}) = E\left(\begin{pmatrix} \mathbf{C} \\ (q \times p) \end{pmatrix} \cdot \begin{pmatrix} \mathbf{Z} \\ (p \times 1) \end{pmatrix} \right) = \begin{pmatrix} \mathbf{C} \\ (q \times p) \end{pmatrix} \cdot \begin{pmatrix} \boldsymbol{\mu}_{\mathbf{Z}} \\ (p \times 1) \end{pmatrix} = \begin{bmatrix} \mathbf{C}_{1} & \mathbf{\mu}_{\mathbf{Z}} \\ (1 \times p) & (p \times 1) \\ \vdots \\ \mathbf{C}_{q}' & \mathbf{\mu}_{\mathbf{Z}} \\ (1 \times p) & (p \times 1) \end{bmatrix} = \begin{pmatrix} \mathbf{0} \\ (q \times 1) \end{pmatrix}$$

Proof. Using the linearity of *E* and Definition 2.2.5.

$$\begin{split} \mu_{\mathbf{Y}} \\ {}_{(q \times 1)} \\ &= E(\mathbf{Y}) \\ {}_{(q \times 1)} \\ &= E\left(\begin{pmatrix} \mathbf{C} & \mathbf{Z} \\ {}_{(q \times p)} \cdot {}_{(p \times 1)} \end{pmatrix} \\ \\ &= E\left(\begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1p} \\ c_{21} & c_{22} & \cdots & c_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ c_{q1} & c_{q2} & \cdots & c_{qp} \end{bmatrix} \begin{bmatrix} Z_1 \\ Z_2 \\ \vdots \\ Z_p \\ (p \times 1) \end{pmatrix} \right) \end{split}$$

$$= \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1p} \\ c_{21} & c_{22} & \cdots & c_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ c_{q1} & c_{q2} & \cdots & c_{qp} \end{bmatrix} E \begin{pmatrix} \begin{bmatrix} Z_1 \\ Z_2 \\ \vdots \\ Z_p \\ p \times 1 \end{pmatrix}^{(q \times p)}$$

$$= \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1p} \\ c_{21} & c_{22} & \cdots & c_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ c_{q1} & c_{q2} & \cdots & c_{qp} \end{bmatrix} \begin{bmatrix} E(Z_1) \\ E(Z_2) \\ \vdots \\ E(Z_p) \end{bmatrix}$$

$$= \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1p} \\ c_{21} & c_{22} & \cdots & c_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ c_{q1} & c_{q2} & \cdots & c_{qp} \end{bmatrix} \begin{bmatrix} \mu_{z,1} \\ \mu_{z,2} \\ \vdots \\ \mu_{z,p} \\ (p \times 1) \end{bmatrix}$$

$$= \begin{bmatrix} c_{11} \mu_{z,1} + c_{12} \mu_{z,2} + \cdots + c_{1p} \mu_{z,p} \\ (p \times 1) \\ e^{(q \times p)} & (p \times 1) \end{bmatrix}$$

$$= \begin{bmatrix} c_{11} \mu_{z,1} + c_{12} \mu_{z,2} + \cdots + c_{1p} \mu_{z,p} \\ c_{21} \mu_{z,1} + c_{22} \mu_{z,2} + \cdots + c_{2p} \mu_{z,p} \\ \vdots \\ c_{q1} \mu_{z,1} + c_{q2} \mu_{z,2} + \cdots + c_{qp} \mu_{z,p} \end{bmatrix}$$

$$= \begin{bmatrix} c_{11} \mu_{z,1} + c_{12} \mu_{z,2} + \cdots + c_{qp} \mu_{z,p} \\ (p \times 1) \\ e^{(1 \times p)} & (p \times 1) \\ (q \times 1) \end{bmatrix}$$

$$= \begin{bmatrix} c_{1} & \cdot \mu_{z} \\ (1 \times p) & (p \times 1) \\ \vdots \\ c'_{q} & \cdot \mu_{z} \\ (1 \times p) & (p \times 1) \\ (q \times 1) \end{bmatrix}$$

Thus, the *i*th row of $\underset{(p \times 1)}{\mathbf{Y}}$ has population mean

$$\bar{Y}_i = E(Y_i) = E\left(\mathbf{c}'_i \cdot \mathbf{Z}_{(1 \times p)}\right) = \mathbf{c}'_i \cdot \mathbf{\mu}_{\mathbf{Z}}_{(1 \times p)} = 0$$

for i = 1, 2, ..., q.

3.5.5 Population Variance-Covariance Matrix for *q* Linear

Combinations of Standardized Continuous Random Variables

Theorem 3.5.5. (Population Variance-Covariance Matrix for *q* Linear Combinations

of **Z**). Suppose q linear combinations of $\underset{(p \times 1)}{\mathbf{Z}}$, $Y_i = \underset{(1 \times p)}{\mathbf{c}'_i} \cdot \underset{(p \times 1)}{\mathbf{Z}}$ are given by Definition

3.5.2 and a population variance-covariance of $\sum_{(p \times 1)'} \sum_{(p \times p)} \sum_{(p \times p)} p$ is given by

Theorem 3.4.4. Then the symmetric population variance-covariance matrix for q linear combinations of $\underset{(p \times 1)}{\mathbf{Z}}, \underset{(p \times 1)}{\mathbf{Y}}$, is given by

$$\sum_{(q \times q)} = \operatorname{Cov}(\mathbf{Y}) = \underset{(q \times p)}{\mathbf{C}} \cdot \underset{(p \times p)}{\sum_{\mathbf{Z}}} \cdot \underset{(p \times q)}{\mathbf{C}'} = \underset{(q \times p)}{\mathbf{C}} \cdot \underset{(p \times p)}{\boldsymbol{\rho}} \cdot \underset{(p \times q)}{\mathbf{C}'}$$

Proof. Using Definition 2.2.5 for matrix multiplication and following Theorem 3.5.2 for computation of diagonal elements and Theorem 3.5.3 for computation of off-diagonal elements.

$$\begin{split} \sum_{\substack{(q \times q) \\ (q \times q)}} &= \underset{(q \times p)}{\text{Cov}(\mathbf{Y})} \\ &= \underset{(q \times p)}{\mathbf{C}} \cdot \underset{(p \times p)}{\sum_{\substack{(p \times p) \\ (p \times p) \\ \vdots \\ \vdots \\ (q \times p) \\ (q \times p) \\ (p \times p) \\ (p \times p) \\ (p \times p) \\ \end{array}} \begin{bmatrix} \sigma_{z,11} & \sigma_{z,12} & \cdots & \sigma_{z,1p} \\ \sigma_{z,21} & \sigma_{z,22} & \cdots & \sigma_{z,2p} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{z,p1} & \sigma_{z,p2} & \cdots & \sigma_{z,pp} \\ \vdots \\ (p \times p) \\ \end{array} \right] \begin{bmatrix} c_{11} & c_{21} & \cdots & c_{q1} \\ c_{12} & c_{22} & \cdots & c_{q2} \\ \vdots \\ c_{1p} & c_{2p} & \cdots & c_{qp} \\ (p \times q) \\ (p \times q) \\ (p \times q) \\ (p \times q) \\ \end{split}$$

$$= \begin{bmatrix} \mathbf{c}_{1}^{\prime} \cdot \sum_{\mathbf{Z}} \cdot \mathbf{c}_{1} & \mathbf{c}_{1}^{\prime} \cdot \sum_{\mathbf{Z}} \cdot \mathbf{c}_{2} & \cdots & \mathbf{c}_{1}^{\prime} \cdot \sum_{\mathbf{Z}} \cdot \mathbf{c}_{q} \\ (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) \\ \mathbf{c}_{2}^{\prime} \cdot \sum_{\mathbf{Z}} \cdot \mathbf{c}_{1} & \mathbf{c}_{2}^{\prime} \cdot \sum_{\mathbf{Z}} \cdot \mathbf{c}_{2} & \cdots & \mathbf{c}_{2}^{\prime} \cdot \sum_{\mathbf{Z}} \cdot \mathbf{c}_{q} \\ (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{c}_{q}^{\prime} \cdot \sum_{\mathbf{Z}} \cdot \mathbf{c}_{1} & \mathbf{c}_{q}^{\prime} \cdot \sum_{\mathbf{Z}} \cdot \mathbf{c}_{2} & \cdots & \mathbf{c}_{q}^{\prime} \cdot \sum_{\mathbf{Z}} \cdot \mathbf{c}_{q} \\ (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{c}_{q}^{\prime} \cdot \mathbf{c}_{\mathbf{Z}} \cdot \mathbf{c}_{\mathbf{Z}} & \mathbf{c}_{\mathbf{Z}} & \mathbf{c}_{\mathbf{Z}} & \cdots & \mathbf{c}_{q}^{\prime} \cdot \sum_{\mathbf{Z}} \cdot \mathbf{c}_{\mathbf{Z}} \\ (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{c}_{q}^{\prime} \cdot \mathbf{c}_{22} & \cdots & \mathbf{c}_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{c}_{q1} & \mathbf{c}_{22} & \cdots & \mathbf{c}_{qp} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{c}_{q1} & \mathbf{c}_{q2} & \cdots & \mathbf{c}_{qp} \end{bmatrix} \begin{bmatrix} \rho_{11} & \rho_{12} & \cdots & \rho_{1p} \\ \rho_{21} & \rho_{22} & \cdots & \rho_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{p1} & \rho_{p2} & \cdots & \rho_{pp} \end{bmatrix} \begin{bmatrix} c_{11} & c_{21} & \cdots & c_{q1} \\ c_{12} & c_{22} & \cdots & c_{q2} \\ \vdots & \vdots & \ddots & \vdots \\ c_{1p} & c_{2p} & \cdots & c_{qp} \end{bmatrix}$$

$$= \begin{bmatrix} \mathbf{c}_1' \cdot \boldsymbol{\rho} \cdot \mathbf{c}_1 & \mathbf{c}_1' \cdot \boldsymbol{\rho} \cdot \mathbf{c}_2 & \cdots & \mathbf{c}_1' \cdot \boldsymbol{\rho} \cdot \mathbf{c}_q \\ (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) \\ \mathbf{c}_2' \cdot \boldsymbol{\rho} \cdot \mathbf{c}_1 & \mathbf{c}_2' \cdot \boldsymbol{\rho} \cdot \mathbf{c}_2 & \cdots & \mathbf{c}_2' \cdot \boldsymbol{\rho} \cdot \mathbf{c}_q \\ (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{c}_q' \cdot \boldsymbol{\rho} \cdot \mathbf{c}_1 & \mathbf{c}_q' \cdot \boldsymbol{\rho} \cdot \mathbf{c}_2 & \cdots & \mathbf{c}_q' \cdot \boldsymbol{\rho} \cdot \mathbf{c}_q \\ (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{c}_q' \cdot \boldsymbol{\rho} \cdot \mathbf{c}_1 & \mathbf{c}_q' \cdot \boldsymbol{\rho} \cdot \mathbf{c}_2 & \cdots & \mathbf{c}_q' \cdot \boldsymbol{\rho} \cdot \mathbf{c}_q \\ (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) \end{bmatrix}$$

Thus, the *i*th row $\mathbf{Y}_{(q \times 1)}$ has population variance

$$\operatorname{var}(Y_i) = \underset{(1 \times p)}{\mathbf{c}'_i} \cdot \underset{(p \times p)}{\sum_{i \in p}} \cdot \underset{(p \times 1)}{\mathbf{c}_i} = \underset{(1 \times p)}{\mathbf{c}'_i} \cdot \underset{(p \times p)}{\boldsymbol{\rho}} \cdot \underset{(p \times 1)}{\mathbf{c}_i}$$

for i = 1, 2, ..., q.

And the *i*th row and *k*th row of $\mathbf{Y}_{(q \times 1)}$ have population covariance

$$\operatorname{cov}(Y_i, Y_k) = \frac{\mathbf{c}'_i}{(1 \times p)} \cdot \sum_{\substack{(p \times p) \\ (p \times p)}} \frac{\mathbf{c}_k}{(p \times 1)} = \frac{\mathbf{c}'_k}{(1 \times p)} \cdot \sum_{\substack{(p \times p) \\ (1 \times p)}} \frac{\mathbf{c}_k}{(p \times 1)} = \frac{\mathbf{c}'_k}{(1 \times p)} \cdot \frac{\mathbf{\rho}}{(p \times 1)} \cdot \frac{\mathbf{c}_k}{(p \times 1)}$$

for i, k = 1, 2, ..., q.

Chapter 4

Multivariate Sample Theory

4.1 Organization of Multivariate Sample Data

Multivariate sample data arise whenever an investigator, seeking to understand a social or physical phenomenon, selects a number p > 1 of **variables** or **characteristics** to record. The values of these variables are all recorded for each distinct **multivariate observation**.

We will use the notation x_{jk} , for realized samples, to indicate the particular value of the *k*th variable (characteristic) on the *j*th multivariate observation. That is,

 x_{jk} = measurement of the *k*th variable on the *j*th multivariate observation Consequently, *n* multivariate observations on *p* variables (characteristic) can be displayed as follows:

	Variable 1	Variable 2	•••	Variable <i>k</i>	•••	Variable p
Observation 1:	<i>x</i> ₁₁	<i>x</i> ₁₂		x_{1k}	•••	x_{1p}
Observation 2:	<i>x</i> ₂₁	<i>x</i> ₂₂	•••	x_{2k}	•••	x_{2p}
:	:	:		:		:
Observation <i>j</i> :	x_{j1}	x_{j2}	•••	x_{jk}	•••	x_{jp}
:	:	:		:		
Observation <i>n</i> :	x_{n1}	x_{n2}	•••	x_{nk}	•••	x_{np}

for j = 1, 2, ..., n multivariate observations and k = 1, 2, ..., p variables [3, p. 5].

A variable or column of the multivariate sample data array is called a realized **characteristic vector** of dimension $n \times 1$

$$\mathbf{x}_{k}_{(n\times1)} = \begin{bmatrix} x_{1k} \\ x_{2k} \\ \vdots \\ x_{nk} \\ (n\times1) \end{bmatrix}$$

for k = 1, 2, ..., p. Where the transpose of the characteristic vector is of dimension $1 \times n$

$$\mathbf{x}'_{k} = [x_{1k}, x_{2k}, \dots, x_{nk}].$$

A realized **multivariate observation vector** of dimension $p \times 1$ is given by

$$\mathbf{x}_{j}_{(p\times1)} = \begin{bmatrix} x_{j1} \\ x_{j2} \\ \vdots \\ x_{jp} \end{bmatrix}_{(p\times1)}$$

for j = 1, 2, ..., n. Where a row of the multivariate sample data array is given by the transpose of a multivariate observation vector of dimension $1 \times p$

$$\mathbf{x}'_{j} = \begin{bmatrix} x_{j1}, x_{j2}, \dots, x_{jp} \end{bmatrix}.$$

The $n \times p$ multivariate sample matrix $\underset{(n \times p)}{\mathbf{X}}$ can also be displayed as n

realized transposed multivariate observation vectors, stacked on top of each-other, each with *p* characteristics or variable elements.

$$\mathbf{X}_{(n \times p)} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1k} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2k} & \cdots & x_{2p} \\ \vdots & \vdots & & \vdots & & \vdots \\ x_{j1} & x_{j2} & \cdots & x_{jk} & \cdots & x_{jp} \\ \vdots & \vdots & & \vdots & & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nk} & \cdots & x_{np} \end{bmatrix} = \begin{bmatrix} \mathbf{x}'_1 \\ \mathbf{x}'_2 \\ \vdots \\ \mathbf{x}'_j \\ \vdots \\ \mathbf{x}'_n \end{bmatrix}$$

for j = 1, 2, ..., n multivariate observations and k = 1, 2, ..., p variables.

4.2 Random Samples

4.2.1 Univariate Random Sample

Definition 4.2.1 (Univariate Random Sample). *If random variables* X_{jk} *for*

j = 1, 2, ..., n are independent and identically distributed (iid) from a common population continuous random variable X_k , with univariate marginal pdf $f_k(x_k)$, population mean μ_k , and population variance σ_{kk} ; then, $X_{1k}, X_{2k}, ..., X_{nk}$ constitute a univariate random sample of size n [6, p. 226].

One should be aware that the elements X_{jk} for j = 1, 2, ..., n must be independent; however, random variables (characteristics) X_k from k = 1, 2, ..., p are generally not assumed independent--especially when realized on the same multivariate observations [3, p. 119].

4.2.2 Multivariate Random Sample

Definition 4.2.2 (Multivariate Random Sample). If random vectors

$$\mathbf{X}_{j} = \begin{bmatrix} X_{j1} \\ X_{j2} \\ \vdots \\ X_{jp} \end{bmatrix}$$

$$(p \times 1)$$

for j = 1,2, ..., *n are independent and identically distributed* (*iid*) *observed from a common population random vector of continuous random variables*

$$\mathbf{X}_{(p \times 1)} = \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_p \end{bmatrix},$$
$$(p \times 1)$$

defined in Definition 3.21., with joint pdf

$$f\left(\mathbf{x}_{(p\times 1)}\right) = f_{12\cdots p}(x_1, x_2, \dots, x_p),$$

defined in definition 3.2.2., population mean vector

$$\boldsymbol{\mu}_{\mathbf{X}} = E(\mathbf{X}) = \begin{bmatrix} E(X_1) \\ E(X_2) \\ \vdots \\ E(X_p) \end{bmatrix} = \begin{bmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_p \end{bmatrix},$$
$$(p \times 1) \quad (p \times 1) \quad$$

defined in Definition 3.2.11., and population variance-covariance matrix

$$\sum_{(p \times p)} = \operatorname{Cov}(\mathbf{X}) = E(\mathbf{X} - \boldsymbol{\mu}_{\mathbf{X}})(\mathbf{X} - \boldsymbol{\mu}_{\mathbf{X}})'_{(1 \times p)}$$

defined in Theorem 3.2.1.,then, these random vectors \mathbf{X}_1 , \mathbf{X}_2 , ..., \mathbf{X}_n constitute $(p \times 1)$ $(p \times 1)$ $(p \times 1)$ $(p \times 1)$

a **multivariate random sample** *of size n from a p – variate population.*

4.2.3 Multivariate Random Sample Matrix

Definition 4.2.3 (Multivariate Random Sample Matrix). *A* **multivariate random sample matrix** *is a random matrix whose row vectors are unrealized multivariate sample observations*

$$\mathbf{X}'_{j} = \begin{bmatrix} X_{j1}, X_{j2}, \dots, X_{jp} \end{bmatrix}$$
(1×p)

for j = 1, 2, ..., n. In addition, the column vectors of the matrix are unrealized variables or characteristics taken on each of the n multivariate sample observations

$$\mathbf{X}_{k}_{(n\times 1)} = \begin{bmatrix} X_{1k} \\ X_{2k} \\ \vdots \\ X_{nk} \end{bmatrix}_{(n\times 1)}$$

for k = 1, 2, ..., p. Let the (j, k)th entry be a continuous random variable X_{jk} , then the $n \times p$ multivariate random sample matrix $\underset{(n \times p)}{\mathbf{X}} = \{X_{jk}\}$ is defined by

$$\mathbf{X}_{(n \times p)} = \begin{bmatrix} X_{11} & X_{12} & \cdots & X_{1k} & \cdots & X_{1p} \\ X_{21} & X_{22} & \cdots & X_{2k} & \cdots & X_{2p} \\ \vdots & \vdots & & \vdots & & \vdots \\ X_{j1} & X_{j2} & \cdots & X_{jk} & \cdots & X_{jp} \\ \vdots & \vdots & & \vdots & & \vdots \\ X_{n1} & X_{n2} & \cdots & X_{nk} & \cdots & X_{np} \end{bmatrix} = \begin{bmatrix} \mathbf{X}_{1}^{\prime} \\ \mathbf{X}_{2}^{\prime} \\ \vdots \\ \mathbf{X}_{j}^{\prime} \\ \vdots \\ \mathbf{X}_{j}^{\prime} \\ \vdots \\ \mathbf{X}_{n1}^{\prime} \end{bmatrix}$$

for j = 1, 2, ..., n and k = 1, 2, ..., p. Since the row vectors \mathbf{X}'_1 , \mathbf{X}'_2 , ..., \mathbf{X}'_n $(1 \times p)$ $(1 \times p)$, $(1 \times p)$

represent iid multivariate sample observations with common joint pdf,

 \mathbf{X}_1 , \mathbf{X}_2 , ..., \mathbf{X}_n are said to form a multivariate random sample [3, p. 119]. (p×1) (p×1) (p×1) One often refers to each

$$\mathbf{X}_{j}_{(p\times1)} = \begin{bmatrix} X_{j1} \\ X_{j2} \\ \vdots \\ X_{jp} \end{bmatrix}_{(p\times1)}$$

for j = 1, 2, ..., n, as an *unrealized* multivariate sample observation (vector). When the multivariate sample observation (vector) has been *realized* (drawn) the notation becomes

$$\mathbf{x}_{j}_{(p\times1)} = \begin{bmatrix} x_{j1} \\ x_{j2} \\ \vdots \\ x_{jp} \end{bmatrix}_{(p\times1)}$$

for j = 1, 2, ..., n. Similarly, one often refers to each

$$\mathbf{X}_{k}_{(n\times1)} = \begin{bmatrix} X_{1k} \\ X_{2k} \\ \vdots \\ X_{nk} \end{bmatrix}$$

for k = 1, 2, ..., p, as an *unrealized* sample characteristic (vector).

When the sample characteristic (vector) has been realized the notation becomes

$$\mathbf{x}_{k}_{(n\times1)} = \begin{bmatrix} x_{1k} \\ x_{2k} \\ \vdots \\ x_{nk} \\ (n\times1) \end{bmatrix}$$

for k = 1, 2, ..., p.

4.3 Sample Statistics

Definition 4.3.1 (Sample Mean). Let X_{jk} for j = 1, 2, ..., n be iid continuous random variables with common population univariate marginal pdf $f_k(x_k)$, mean μ_k , and variance σ_{kk} . Then the unrealized sample mean \overline{X}_k is defined by

$$\bar{X}_k = \frac{1}{n} \sum_{j=1}^n X_{jk}$$

for k = 1, 2, ..., p where $-\infty < \overline{X}_k < \infty$.

Because $E(\overline{X}_k) = \mu_k$, one can say \overline{X}_k is an unbiased estimator for the univariate marginal population mean μ_k .

Definition 4.3.2 (Sample Variance). Let X_{jk} for j = 1, 2, ..., n be iid continuous random variables with common population univariate marginal pdf $f_k(x_k)$, mean μ_k , and variance σ_{kk} . Then the unrealized sample variance S_{kk} is defined by

$$S_{kk} = \frac{1}{n-1} \sum_{j=1}^{n} (X_{jk} - \bar{X}_k)^2$$

for k = 1, 2, ..., p where $0 < S_{kk} < \infty$.

Because $E(S_{kk}) = \sigma_{kk}$, one can say S_{kk} is an unbiased estimator for the univariate marginal population variance σ_{kk} . Although, the **sample standard deviation** $\sqrt{S_{kk}}$ is a biased estimator for the univariate marginal population standard deviation $\sqrt{\sigma_{kk}}$; given, $E(\sqrt{S_{kk}}) \neq \sqrt{\sigma_{kk}}$.

Definition 4.3.3 (Sample Covariance). Let $\mathbf{X}_{j}_{(2 \times 1)} = \begin{bmatrix} X_{ji} \\ X_{jk} \end{bmatrix}$ for j = 1, 2, ..., n be iid

continuous random vectors with common population bivariate (joint) marginal pdf $f_{ik}(x_i, x_k)$. Denote the common population univariate marginal pdf for X_{ji} as $f_i(x_i)$ with mean and variance $\begin{bmatrix} \mu_i \\ \sigma_{ii} \end{bmatrix}$ and common population univariate marginal pdf for $\begin{bmatrix} \mu_i \\ \sigma_{ii} \end{bmatrix}$ and common population univariate marginal pdf for

 X_{jk} as $f_k(x_k)$ with mean and variance $\begin{bmatrix} \mu_k \\ \sigma_{kk} \\ (2 \times 1) \end{bmatrix}$. Then the unrealized sample covariance

S_{ik} is defined by

$$S_{ik} = \frac{1}{n-1} \sum_{j=1}^{n} (X_{ji} - \bar{X}_i) (X_{jk} - \bar{X}_k)$$

for i, k = 1, 2, ..., p where $-\infty < S_{ik} < \infty$.

Because $E(S_{ik}) = \sigma_{ik}$, one can say S_{ik} is an unbiased estimator for the

bivariate marginal population covariance σ_{ik} . Given that $\mathbf{X}_{j} = \begin{bmatrix} X_{ji} \\ X_{jk} \end{bmatrix} \subseteq \begin{bmatrix} X_{j1} \\ X_{j2} \\ \vdots \\ (2 \times 1) \end{bmatrix}$ for $\begin{bmatrix} X_{j1} \\ X_{j2} \\ \vdots \\ X_{jp} \end{bmatrix}$ (p×1)

j = 1, 2, ..., n, the multivariate random sample is collected on p characteristics and then subset into bivariate pairs. Furthermore, $S_{ik} = S_{ki}$ and when i = k the sample covariance becomes the sample variance S_{kk} . **Definition 4.3.4** (Sample Correlation). Let $\mathbf{X}_{j}_{(2 \times 1)} = \begin{bmatrix} X_{ji} \\ X_{jk} \\ (2 \times 1) \end{bmatrix}$ for j = 1, 2, ..., n be iid

continuous random vectors with common population bivariate (joint) marginal pdf $f_{ik}(x_i, x_k)$. Denote the common population univariate marginal pdf for X_{ji} as $f_i(x_i)$ with mean and variance $\begin{bmatrix} \mu_i \\ \sigma_{ii} \end{bmatrix}$ and common population univariate marginal pdf for $\begin{bmatrix} \mu_i \\ \sigma_{ii} \end{bmatrix}$

 X_{jk} as $f_k(x_k)$ with mean and variance $\begin{bmatrix} \mu_k \\ \sigma_{kk} \\ (2 \times 1) \end{bmatrix}$. Then the unrealized sample correlation

 R_{ik} is defined by

$$R_{ik} = \frac{S_{ik}}{\sqrt{S_{ii}}\sqrt{S_{kk}}}$$

$$= \frac{\frac{1}{n-1}\sum_{j=1}^{n} (X_{ji} - \bar{X}_{i})(X_{jk} - \bar{X}_{k})}{\sqrt{\frac{1}{n-1}\sum_{j=1}^{n} (X_{ji} - \bar{X}_{i})^{2}}\sqrt{\frac{1}{n-1}\sum_{j=1}^{n} (x_{jk} - \bar{x}_{k})^{2}}}$$

$$= \frac{\sum_{j=1}^{n} (X_{ji} - \bar{X}_{i})(X_{jk} - \bar{X}_{k})}{\sqrt{\sum_{j=1}^{n} (X_{ji} - \bar{X}_{i})^{2}}\sqrt{\sum_{j=1}^{n} (X_{jk} - \bar{X}_{k})^{2}}}$$

for *i*, k = 1, 2, ..., p where $-1 \le R_{ik} \le 1$.

Because $E(R_{ik}) \neq \rho_{ik}$, one can say R_{ik} is a biased estimator for the bivariate marginal population correlation ρ_{ik} . Next, $R_{ik} = R_{ki}$ and when i = k the sample correlation becomes $R_{kk} = \frac{S_{kk}}{\sqrt{S_{kk}}\sqrt{S_{kk}}} = \frac{S_{kk}}{S_{kk}} = 1$.

4.4 Sample Mean Vector, Variance-Covariance Matrix, and Correlation Matrix

4.4.1 Sample Mean Vector

Theorem 4.4.1 (Sample Mean Vector for **X**). Let random vectors \mathbf{X}_1 , \mathbf{X}_2 , ..., \mathbf{X}_n ($p \times 1$) constitute a multivariate random sample defined in Definition 4.2.2. Then the $p \times 1$

unrealized sample mean vector for $\underset{(n \times p)}{\mathbf{X}}$ *is defined by*

$$\overline{\mathbf{X}}_{(p\times1)} = \frac{1}{n} \sum_{j=1}^{n} \mathbf{X}_{j} = \frac{1}{n} \cdot \mathbf{X}'_{(p\times n)} \cdot \mathbf{1}_{(n\times1)} = \begin{bmatrix} X_{1} \\ \overline{X}_{2} \\ \vdots \\ \overline{X}_{p} \end{bmatrix}_{(p\times1)}$$

where $-\infty < \overline{X}_k < \infty$, for k = 1, 2, ..., p [3, p. 138].

Proof. Use Definition 2.1.4, Definition 2.2.2, Definition 2.2.3, and Definition 4.3.1.

$$\begin{split} \mathbf{X}_{(p \times 1)} \\ &= \frac{1}{n} \sum_{j=1}^{n} \mathbf{X}_{j} \\ &= \frac{1}{n} \left(\mathbf{X}_{1} + \mathbf{X}_{2} + \dots + \mathbf{X}_{n} \\ (p \times 1) + (p \times 1) + (p \times 1) + (p \times 1) \right) \\ &= \frac{1}{n} \left(\begin{bmatrix} X_{11} \\ X_{12} \\ \vdots \\ X_{1p} \\ (p \times 1) + (p \times 1) \end{bmatrix} + \begin{bmatrix} X_{21} \\ X_{22} \\ \vdots \\ X_{2p} \\ (p \times 1) \end{bmatrix} + \dots + \begin{bmatrix} X_{n1} \\ X_{n2} \\ \vdots \\ X_{np} \\ (p \times 1) \end{bmatrix} \right) \end{split}$$

$$= \frac{1}{n} \begin{bmatrix} X_{11} + X_{21} + \dots + X_{n1} \\ X_{12} + X_{22} + \dots + X_{n2} \\ \vdots \\ X_{1p} + X_{2p} + \dots + X_{np} \end{bmatrix}$$
$$= \frac{1}{n} \begin{bmatrix} \sum_{j=1}^{n} X_{j1} \\ \sum_{j=1}^{n} X_{j2} \\ \vdots \\ \sum_{j=1}^{n} X_{jp} \end{bmatrix}$$
$$= \begin{bmatrix} \frac{1}{n} \sum_{j=1}^{n} X_{j1} \\ \frac{1}{n} \sum_{j=1}^{n} X_{j2} \\ \vdots \\ \frac{1}{n} \sum_{j=1}^{n} X_{jp} \end{bmatrix}$$
$$= \begin{bmatrix} \frac{\bar{X}_{1}}{\bar{X}_{2}} \\ \vdots \\ \bar{X}_{p} \end{bmatrix}_{(p \times 1)}$$

In terms of matrix operations $\overline{\mathbf{X}}_{(p imes 1)}$ can be obtained by

$$\begin{split} \mathbf{\overline{X}}_{(p\times1)} &= \frac{1}{n} \cdot \underbrace{\mathbf{X}'}_{(p\times n)} \cdot \underbrace{\mathbf{1}}_{(n\times1)} \\ &= \frac{1}{n} \begin{bmatrix} X_{11} & X_{21} & \cdots & X_{n1} \\ X_{12} & X_{22} & \cdots & X_{n2} \\ \vdots & \vdots & \ddots & \vdots \\ X_{1p} & X_{2p} & \cdots & X_{np} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{1}_1 \\ \mathbf{1}_2 \\ \vdots \\ \mathbf{1}_n \\ (p\times n) \end{bmatrix} \end{split}$$

$$= \frac{1}{n} \begin{bmatrix} X_{11} + X_{21} + \dots + X_{n1} \\ X_{12} + X_{22} + \dots + X_{n2} \\ \vdots \\ X_{1p} + X_{2p} + \dots + X_{np} \end{bmatrix}$$

$$= \frac{1}{n} \begin{bmatrix} \sum_{j=1}^{n} X_{j1} \\ \sum_{j=1}^{n} X_{j2} \\ \vdots \\ \sum_{j=1}^{n} X_{jp} \end{bmatrix}$$

$$= \begin{bmatrix} \frac{1}{n} \sum_{j=1}^{n} X_{j1} \\ \frac{1}{n} \sum_{j=1}^{n} X_{j2} \\ \vdots \\ \frac{1}{n} \sum_{j=1}^{n} X_{jp} \end{bmatrix}$$

$$= \begin{bmatrix} \frac{\bar{X}_{1}}{\bar{X}_{2}} \\ \vdots \\ \bar{X}_{p} \\ (p \times 1) \end{bmatrix} \bullet$$
Because $F\left(= \bar{X} \right) = -M_{12}$

Because $E\left(\overline{\mathbf{X}}_{(p\times 1)}\right) = \mu_{\mathbf{X}}$, one can say $\overline{\mathbf{X}}_{(p\times 1)}$ is an unbiased estimator for the

population mean vector μ_X .

4.4.2 Sample Variance-Covariance Matrix

Theorem 4.4.2 (Sample Variance-Covariance Matrix for X). Let random

vectors \mathbf{X}_1 , \mathbf{X}_2 , ..., \mathbf{X}_n constitute a multivariate random sample defined in $(p \times 1)$ $(p \times 1)$ $(p \times 1)$

Definition 4.2.2. Assume the sample mean vector $\overline{\mathbf{X}}_{(p \times 1)}$ defined in Theorem 4.4.1

exists. Then the $p \times p$ *symmetric unrealized* **sample variance-covariance matrix for**

$$\begin{split} \mathbf{X}_{(n\times p)} &\text{ is defined by} \\ \mathbf{S}_{\mathbf{X}}_{(p\times p)} &= \frac{1}{n-1} \sum_{j=1}^{n} \left(\mathbf{X}_{j} - \overline{\mathbf{X}}_{(p\times 1)} \right) \left(\mathbf{X}_{j} - \overline{\mathbf{X}}_{(p\times 1)} \right)' \\ &= \frac{1}{n-1} \cdot \left(\mathbf{X}_{(n\times p)} - \frac{1}{n} \cdot \mathbf{1}_{(n\times 1)} \cdot \mathbf{1}'_{(1\times n)} \cdot \mathbf{X}_{(n\times p)} \right)' \cdot \left(\mathbf{X}_{(n\times p)} - \frac{1}{n} \cdot \mathbf{1}_{(n\times 1)} \cdot \mathbf{1}'_{(1\times n)} \cdot \mathbf{X}_{(n\times p)} \right) \\ &= \frac{1}{n-1} \cdot \left(\mathbf{X}_{(n\times p)} - \mathbf{1}_{n} \cdot \mathbf{1}_{(n\times 1)} \cdot \mathbf{X}_{(1\times p)} \right)' \cdot \left(\mathbf{X}_{(n\times p)} - \mathbf{1}_{n} \cdot \mathbf{X}_{(n\times 1)} \cdot \mathbf{1}_{(1\times n)} \cdot \mathbf{X}_{(n\times p)} \right) \\ &= \frac{1}{n-1} \cdot \left(\mathbf{X}_{(n\times p)} - \mathbf{1}_{(n\times 1)} \cdot \mathbf{X}_{(1\times p)} \right)' \cdot \left(\mathbf{X}_{(n\times p)} - \mathbf{1}_{(n\times 1)} \cdot \mathbf{X}_{(1\times p)} \right) = \begin{bmatrix} S_{11} & S_{12} & \cdots & S_{1p} \\ S_{21} & S_{22} & \cdots & S_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ S_{p1} & S_{p2} & \cdots & S_{pp} \end{bmatrix} \end{split}$$

[3, pp. 123,138].

Proof. Use Definition 2.1.4, Definition 2.2.2, Definition 2.2.3, Definition 2.2.3, Definition 4.2.3, Definition 4.3.2, and Definition 4.3.3.

$$\begin{split} \mathbf{S}_{\mathbf{X}} &= \frac{1}{n-1} \sum_{j=1}^{n} \left(\mathbf{X}_{j} - \bar{\mathbf{X}}_{(p\times 1)} \right) \left(\mathbf{X}_{j} - \bar{\mathbf{X}}_{(p\times 1)} \right)' \\ &= \frac{1}{n-1} \sum_{j=1}^{n} \left(\begin{bmatrix} X_{j1} \\ X_{j2} \\ \vdots \\ X_{jp} \end{bmatrix} - \begin{bmatrix} \bar{X}_{1} \\ \bar{X}_{2} \\ \vdots \\ \bar{X}_{p} \end{bmatrix} \right) \left(\begin{bmatrix} X_{j1} \\ X_{j2} \\ \vdots \\ X_{jp} \end{bmatrix} - \begin{bmatrix} \bar{X}_{1} \\ \bar{X}_{2} \\ \vdots \\ \bar{X}_{p} \end{bmatrix} \right)' \end{split}$$

$$\begin{split} &= \frac{1}{n-1} \sum_{j=1}^{n} \left(\begin{bmatrix} X_{j1} - \bar{X}_{1} \\ X_{j2} - \bar{X}_{2} \\ \vdots \\ X_{jp} - \bar{X}_{p} \end{bmatrix} \right) \left(\begin{bmatrix} X_{j1} - \bar{X}_{1} \\ X_{j2} - \bar{X}_{2} \\ \vdots \\ X_{jp} - \bar{X}_{p} \end{bmatrix} \right) \\ &= \frac{1}{n-1} \sum_{j=1}^{n} \left(\begin{bmatrix} X_{j1} - \bar{X}_{1} \\ X_{j2} - \bar{X}_{2} \\ \vdots \\ X_{jp} - \bar{X}_{p} \end{bmatrix} \right) ([X_{j1} - \bar{X}_{1}, X_{j2} - \bar{X}_{2}, \dots, X_{jp} - \bar{X}_{p}]) \\ &= \begin{bmatrix} \frac{\sum_{j=1}^{n} (X_{j1} - \bar{X}_{1})^{2}}{n-1} & \frac{\sum_{j=1}^{n} (X_{j1} - \bar{X}_{1})(X_{j2} - \bar{X}_{2})}{n-1} & \cdots & \frac{\sum_{j=1}^{n} (X_{j1} - \bar{X}_{1})(X_{jp} - \bar{X}_{p})}{n-1} \\ &= \begin{bmatrix} \frac{\sum_{j=1}^{n} (X_{j2} - \bar{X}_{2})(X_{j1} - \bar{X}_{1})}{n-1} & \frac{\sum_{j=1}^{n} (X_{j2} - \bar{X}_{2})^{2}}{n-1} & \cdots & \frac{\sum_{j=1}^{n} (X_{j2} - \bar{X}_{2})(X_{jp} - \bar{X}_{p})}{n-1} \\ &\vdots & \ddots & \vdots \\ &\sum_{j=1}^{n} (X_{jp} - \bar{X}_{p})(X_{j1} - \bar{X}_{1})} & \frac{\sum_{j=1}^{n} (X_{jp} - \bar{X}_{p})(X_{j2} - \bar{X}_{2})}{n-1} & \cdots & \frac{\sum_{j=1}^{n} (X_{jp} - \bar{X}_{p})(X_{jp} - \bar{X}_{p})}{n-1} \\ &= \begin{bmatrix} S_{11} & S_{12} & \cdots & S_{1p} \\ S_{21} & S_{22} & \cdots & S_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ S_{p1} & S_{p2} & \cdots & S_{pp} \end{bmatrix} \end{split}$$

In terms of matrix operations $\underset{(p \times p)}{\mathbf{S}_{\mathbf{X}}}$ can be obtained by

$$\mathbf{S}_{\mathbf{X}} = \frac{1}{n-1} \cdot \left(\mathbf{X}_{(n \times p)} - \frac{1}{n} \cdot \mathbf{1}_{(n \times 1)} \cdot \mathbf{1}_{(1 \times n)} \cdot \mathbf{X}_{(n \times p)} \right)' \cdot \left(\mathbf{X}_{(n \times p)} - \frac{1}{n} \cdot \mathbf{1}_{(n \times 1)} \cdot \mathbf{1}_{(1 \times n)} \cdot \mathbf{X}_{(n \times p)} \right)$$

where

$$\begin{aligned} \frac{1}{n} \cdot \underbrace{\mathbf{1}}_{(n \times 1)} \cdot \underbrace{\mathbf{1}'}_{(1 \times n)} \cdot \underbrace{\mathbf{X}}_{(n \times p)} \\ &= \frac{1}{n} \cdot \begin{bmatrix} 1_1 \\ 1_2 \\ \vdots \\ 1_n \end{bmatrix}_{(n \times 1)} \cdot \begin{bmatrix} 1_1, 1_2, \dots, 1_n \end{bmatrix} \cdot \begin{bmatrix} X_{11} & X_{12} & \cdots & X_{1p} \\ X_{21} & X_{22} & \cdots & X_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ X_{n1} & X_{n2} & \cdots & X_{np} \end{bmatrix} \end{aligned}$$
$$= \begin{bmatrix} 1_{1} \\ 1_{2} \\ \vdots \\ 1_{n} \\ (n \times 1) \end{bmatrix} \cdot \frac{1}{n} \cdot \left[\sum_{j=1}^{n} X_{j_{1}}, \sum_{\substack{j=1 \ (1 \times p)}}^{n} X_{j_{2}}, \dots, \sum_{j=1}^{n} X_{jp} \right]$$

$$= \begin{bmatrix} 1_{1} \\ 1_{2} \\ \vdots \\ 1_{n} \\ (n \times 1) \end{bmatrix} \cdot \left[\frac{1}{n} \sum_{j=1}^{n} X_{j_{1}}, \frac{1}{n} \sum_{\substack{j=1 \ (1 \times p)}}^{n} X_{j_{2}}, \dots, \frac{1}{n} \sum_{\substack{j=1 \ (1 \times p)}}^{n} X_{jp} \right]$$

$$= \begin{bmatrix} 1_{1} \\ 1_{2} \\ \vdots \\ 1_{n} \\ (n \times 1) \end{bmatrix} \cdot \left[\bar{X}_{1}, \bar{X}_{2}, \dots, \bar{X}_{p} \right]$$

$$= \begin{bmatrix} \bar{X}_{1} & \bar{X}_{2} & \dots & \bar{X}_{p} \\ (n \times 1) & \ddots & \vdots \\ \bar{X}_{1} & \bar{X}_{2} & \dots & \bar{X}_{p} \\ \vdots & \vdots & \ddots & \vdots \\ \bar{X}_{1} & \bar{X}_{2} & \dots & \bar{X}_{p} \end{bmatrix}$$

Thus,

$$\begin{split} \mathbf{S}_{\mathbf{X}} &= \frac{1}{n-1} \cdot \begin{pmatrix} \mathbf{X} \\ (n \times p) \end{pmatrix} - \frac{1}{n} \cdot \begin{pmatrix} \mathbf{1} \\ (n \times 1) \end{pmatrix} \cdot \begin{pmatrix} \mathbf{1}' \\ (n \times p) \end{pmatrix} + \begin{pmatrix} \mathbf{X} \\ (n \times p) \end{pmatrix} + \begin{pmatrix} \mathbf{1} \\ (n \wedge p) \end{pmatrix} + \begin{pmatrix} \mathbf{1} \\$$

$$= \begin{bmatrix} \frac{\sum_{j=1}^{n} (X_{j1} - \bar{X}_{1})^{2}}{n-1} & \frac{\sum_{j=1}^{n} (X_{j1} - \bar{X}_{1})(X_{j2} - \bar{X}_{2})}{n-1} & \cdots & \frac{\sum_{j=1}^{n} (X_{j1} - \bar{X}_{1})(X_{jp} - \bar{X}_{p})}{n-1} \\ \frac{\sum_{j=1}^{n} (X_{j2} - \bar{X}_{2})(X_{j1} - \bar{X}_{1})}{n-1} & \frac{\sum_{j=1}^{n} (X_{j2} - \bar{X}_{2})^{2}}{n-1} & \cdots & \frac{\sum_{j=1}^{n} (X_{j2} - \bar{X}_{2})(X_{jp} - \bar{X}_{p})}{n-1} \\ \vdots & \ddots & \vdots \\ \frac{\sum_{j=1}^{n} (X_{jp} - \bar{X}_{p})(X_{j1} - \bar{X}_{1})}{n-1} & \frac{\sum_{j=1}^{n} (X_{jp} - \bar{X}_{p})(X_{j2} - \bar{X}_{2})}{n-1} & \cdots & \frac{\sum_{j=1}^{n} (X_{jp} - \bar{X}_{p})^{2}}{n-1} \end{bmatrix} \\ = \begin{bmatrix} S_{11} & S_{12} & \cdots & S_{1p} \\ S_{21} & S_{22} & \cdots & S_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ S_{p1} & S_{p2} & \cdots & S_{pp} \end{bmatrix}$$

The diagonal elements of the sample variance-covariance matrix are the sample variances

$$S_{kk} = (n-1)^{-1} \sum_{j=1}^{n} (X_{jk} - \bar{X}_k)^2$$

for k = 1, 2, ..., p, i = k where $S_{ii} = S_{kk}$. The off-diagonal elements of the sample

variance-covariance matrix are the sample covariances

$$S_{ik} = (n-1)^{-1} \sum_{j=1}^{n} (X_{ji} - \bar{X}_i) (X_{jk} - \bar{X}_k)$$

for $i, k = 1, 2, ..., p, i \neq k$ where $S_{ik} = S_{ki}$. Furthermore,

$$tr(\mathbf{S}_{\mathbf{X}}) = \sum_{k=1}^{p} S_{kk} = S_{11} + S_{22} + \dots + S_{pp}$$

(total sample variance). Because, $E\left(\mathbf{S}_{\mathbf{X}}\right) = \sum_{(p \times p)}$, one can say $\mathbf{S}_{\mathbf{X}}$ is an unbiased

estimator for the population variance-covariance matrix $\sum_{(p \times p)}$.

4.4.3 Sample Standard Deviation Matrix

Definition 4.4.1 (Sample Standard Deviation Matrix for X). Let random

vectors \mathbf{X}_1 , \mathbf{X}_2 , ..., \mathbf{X}_n constitute a multivariate random sample defined in $(p \times 1)$ $(p \times 1)$ $(p \times 1)$

Definition 4.2.2. Assume the sample standard deviations defined in Definition 4.3.2 exists. Then the $p \times p$ diagonal unrealized sample standard deviation matrix for $\mathbf{X}_{(n \times p)}$ is defined by

$$\mathbf{D}_{(p \times p)}^{1/2} = \begin{bmatrix} \sqrt{S_{11}} & 0 & \cdots & 0 \\ 0 & \sqrt{S_{22}} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sqrt{S_{pp}} \end{bmatrix}_{(p \times p)}$$

with inverse

$$(\mathbf{D}_{(p\times p)}^{1/2})^{-1} = \mathbf{D}_{(p\times p)}^{-1/2} = \begin{bmatrix} \frac{1}{\sqrt{S_{11}}} & 0 & \cdots & 0\\ 0 & \frac{1}{\sqrt{S_{22}}} & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots\\ 0 & 0 & \cdots & \frac{1}{\sqrt{S_{pp}}} \end{bmatrix}$$

$$(p\times p)$$

[3, p. 139].

4.4.4 Sample Correlation Matrix

Theorem 4.4.3 (Sample Correlation Matrix for X). Let random

vectors \mathbf{X}_1 , \mathbf{X}_2 , ..., \mathbf{X}_n constitute a multivariate random sample defined in $(p \times 1)$ $(p \times 1)$ $(p \times 1)$

Definition 4.2.2. Assume the sample variance-covariance matrix $\mathbf{S}_{\mathbf{X}}$ defined in $(p \times p)$

Theorem 4.4.2 exists, and the inverse sample standard deviation matrix defined in

Definition 4.4.1 exists. Then the $p \times p$ *symmetric unrealized* **sample correlation**

matrix for $\underset{(n \times p)}{\mathbf{X}}$ is defined by

 $\mathop{\mathbf{R}}_{(p\times p)}$

$$= \mathbf{D}_{(p \times p)}^{-1/2} \cdot \mathbf{S}_{\mathbf{X}} \cdot \mathbf{D}_{(p \times p)}^{-1/2}$$
$$= \begin{bmatrix} 1 & R_{12} & \cdots & R_{1p} \\ R_{21} & 1 & \cdots & R_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ R_{p1} & R_{p2} & \cdots & 1 \\ & & (p \times p) \end{bmatrix}$$

[3, p. 139].

Proof. Use Definition 2.2.5 and Definition 4.3.4.

$$\begin{split} \mathbf{R}_{(p \times p)} &= \mathbf{D}_{(p \times p)}^{-1/2} \cdot \mathbf{S}_{\mathbf{X}} \cdot \mathbf{D}_{(p \times p)}^{-1/2} \\ &= \begin{bmatrix} \frac{1}{\sqrt{S_{11}}} & 0 & \cdots & 0\\ 0 & \frac{1}{\sqrt{S_{22}}} & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots\\ 0 & 0 & \cdots & \frac{1}{\sqrt{S_{pp}}} \end{bmatrix} \begin{bmatrix} S_{11} & S_{12} & \cdots & S_{1p} \\ S_{21} & S_{22} & \cdots & S_{2p} \\ \vdots & \vdots & \ddots & \vdots\\ S_{p1} & S_{p2} & \cdots & S_{pp} \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{S_{11}}} & 0 & \cdots & 0\\ 0 & \frac{1}{\sqrt{S_{22}}} & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots\\ 0 & 0 & \cdots & \frac{1}{\sqrt{S_{pp}}} \end{bmatrix}_{(p \times p)} \end{split}$$

$$= \begin{bmatrix} \frac{S_{11}}{\sqrt{S_{11}}} & \frac{S_{12}}{\sqrt{S_{11}}} & \cdots & \frac{S_{1p}}{\sqrt{S_{11}}} \\ \frac{S_{21}}{\sqrt{S_{22}}} & \frac{S_{22}}{\sqrt{S_{22}}} & \cdots & \frac{S_{2p}}{\sqrt{S_{22}}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{S_{p1}}{\sqrt{S_{pp}}} & \frac{S_{p2}}{\sqrt{S_{pp}}} & \cdots & \frac{S_{pp}}{\sqrt{S_{pp}}} \end{bmatrix} \cdot \begin{bmatrix} \frac{1}{\sqrt{S_{11}}} & 0 & \cdots & 0 \\ 0 & \frac{1}{\sqrt{S_{22}}} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{1}{\sqrt{S_{pp}}} \end{bmatrix} \\ = \begin{bmatrix} \frac{S_{11}}{\sqrt{S_{11}}\sqrt{S_{11}}} & \frac{S_{12}}{\sqrt{S_{11}}\sqrt{S_{22}}} & \cdots & \frac{S_{1p}}{\sqrt{S_{11}}\sqrt{S_{pp}}} \\ \frac{S_{21}}{\sqrt{S_{22}}\sqrt{S_{11}}} & \frac{S_{22}}{\sqrt{S_{22}}\sqrt{S_{22}}} & \cdots & \frac{S_{2p}}{\sqrt{S_{22}}\sqrt{S_{pp}}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{S_{p1}}{\sqrt{S_{pp}}\sqrt{S_{11}}} & \frac{S_{p2}}{\sqrt{S_{pp}}\sqrt{S_{22}}} & \cdots & \frac{S_{pp}}{\sqrt{S_{pp}}\sqrt{S_{pp}}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{S_{p1}}{\sqrt{S_{pp}}\sqrt{S_{11}}} & \frac{S_{p2}}{\sqrt{S_{pp}}\sqrt{S_{22}}} & \cdots & \frac{S_{pp}}{\sqrt{S_{pp}}\sqrt{S_{pp}}} \\ \end{bmatrix} = \begin{bmatrix} 1 & R_{12} & \cdots & R_{1p} \\ R_{21} & 1 & \cdots & R_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ R_{p1} & R_{p2} & \cdots & 1 \end{bmatrix} \bullet$$

The diagonal elements of the sample correlation matrix are

$$R_{kk} = \frac{S_{kk}}{\sqrt{S_{kk}}\sqrt{S_{kk}}} = \frac{S_{kk}}{S_{kk}} = 1$$

for k = 1, 2, ..., p, i = k where $R_{ii} = R_{kk}$. The off-diagonal elements of the sample correlation matrix are

$$R_{ik} = \frac{S_{ik}}{\sqrt{S_{ii}}\sqrt{S_{kk}}}$$

for $i, k = 1, 2, ..., p, i \neq k$ where $R_{ik} = R_{ki}$.

Furthermore,

$$\operatorname{tr}(\mathbf{R}) = \sum_{k=1}^{p} R_{kk} = 1 + 1 + \dots + 1 = p$$

(number of characteristics). Because, $E\left(\mathbf{R}\atop(p\times p)\right) \neq \mathbf{\rho}$, one can say $\mathbf{R}\atop(p\times p)$ is a biased

estimator for the population correlation matrix $p \atop (p imes p)$. Finally,

$$\mathbf{R}_{(p \times p)} = \mathbf{D}_{(p \times p)}^{-1/2} \cdot \mathbf{S}_{\mathbf{X}} \cdot \mathbf{D}_{(p \times p)}^{-1/2} \Rightarrow \mathbf{S}_{\mathbf{X}} = \mathbf{D}_{(p \times p)}^{1/2} \cdot \mathbf{R}_{(p \times p)} \cdot \mathbf{D}_{(p \times p)}^{1/2}$$

[3, p. 140].

4.5 Sample Mean Vector and Variance-Covariance

Matrix for Linear Combinations of Continuous Random

Variables

4.5.1 Linear Combination

Definition 4.5.1 (Linear Combination of **X**). Let $\underset{(p \times 1)}{\mathbf{c}}$ be a $p \times 1$ vector of constants

defined as

$$\mathbf{c}_{(p\times 1)} = \begin{bmatrix} c_1\\c_2\\\vdots\\c_p\\(p\times 1) \end{bmatrix}$$

and let $\underset{(p \times 1)}{\mathbf{X}}$ be a $p \times 1$ population random vector of continuous random variables

$$\mathbf{X}_{(p \times 1)} = \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_p \end{bmatrix}_{(p \times 1)}$$

Now consider a **linear combination of** $\underset{(n \times p)}{\mathbf{X}}$ *of the form*

$$\mathbf{c}'_{(1\times p)} \cdot \mathbf{X}_{(p\times 1)} = \begin{bmatrix} c_1, c_2, \dots, c_p \end{bmatrix} \cdot \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_p \end{bmatrix} = c_1 X_1 + c_2 X_2 + \dots + c_p X_p$$

$$(p\times 1)$$

whose unrealized quantity on the jth multivariate sample observation is

$$\mathbf{c}'_{(1\times p)} \cdot \mathbf{X}_{j} = \begin{bmatrix} c_1, c_2, \dots, c_p \end{bmatrix} \cdot \begin{bmatrix} X_{j1} \\ X_{j2} \\ \vdots \\ X_{jp} \end{bmatrix} = c_1 X_{j1} + c_2 X_{j2} + \dots + c_p X_{jp}$$

$$(p \times 1)$$

for j = 1,2, ..., *n* [3, p. 140].

4.5.2 Sample Statistics for Linear Combinations

Theorem 4.5.1 (Sample Mean of a Linear Combination of X). Let random

vectors \mathbf{X}_1 , \mathbf{X}_2 , ..., \mathbf{X}_n constitute a multivariate random sample defined in $_{(p\times 1)}^{(p\times 1)}$

Definition 4.2.2. Assume the sample mean vector $\overline{\mathbf{X}}_{(p \times 1)}$ defined in Theorem 4.4.1

exists. Next, consider a linear combination of the form $\mathbf{c}'_{(1 \times p)} \cdot \mathbf{X}_{(p \times 1)}$ with jth

multivariate sample observation
$$\mathbf{c}'_{(1 \times p)} \cdot \mathbf{X}_{j}_{(p \times 1)}$$
 given in Definition 4.5.1. *Then, the*

unrealized sample mean of a linear combination of $\underset{(n \times p)}{\mathbf{X}}$ is defined by

sample mean of
$$\mathbf{c}'_{(1\times p)} \cdot \mathbf{X}_{(p\times 1)} = E\left(\mathbf{c}'_{(1\times p)} \cdot \mathbf{X}_{(p\times 1)}\right) = \mathbf{c}'_{(1\times p)} \cdot \mathbf{\overline{X}}_{(p\times 1)}.$$

[3, p. 140].

Proof. Use Definition 2.1.6, Definition 2.1.11, and Result 2.2.1. (d).

$$\begin{split} E\left(\underbrace{\mathbf{c}'_{(1\times p)} \cdot \underbrace{\mathbf{X}}_{(p\times 1)}}_{(p\times 1)}\right) \\ &= \frac{1}{n} \sum_{j=1}^{n} \underbrace{\mathbf{c}'_{(1\times p)} \cdot \underbrace{\mathbf{X}_{j}}_{(p\times 1)}}_{(p\times 1)} \\ &= \frac{1}{n} \left(\underbrace{\mathbf{c}'_{(1\times p)} \cdot \underbrace{\mathbf{X}_{1}}_{(p\times 1)} + \underbrace{\mathbf{c}'_{(1\times p)} \cdot \underbrace{\mathbf{X}_{2}}_{(p\times 1)} + \dots + \underbrace{\mathbf{c}'_{(1\times p)} \cdot \underbrace{\mathbf{X}_{n}}_{(p\times 1)}}_{(p\times 1)}\right) \\ &= \underbrace{\mathbf{c}'_{(1\times p)} \cdot \left[\frac{1}{n} \left(\underbrace{\mathbf{X}_{1}}_{(p\times 1)} + \underbrace{\mathbf{X}_{2}}_{(p\times 1)} + \dots + \underbrace{\mathbf{X}_{n}}_{(p\times 1)}\right)\right]}_{(p\times 1)} \\ &= \underbrace{\mathbf{c}'_{(1\times p)} \cdot \left[\frac{1}{n} \left(\begin{bmatrix}X_{11}\\X_{12}\\\vdots\\X_{1p}\\(p\times 1)\end{bmatrix} + \begin{bmatrix}X_{21}\\X_{22}\\\vdots\\X_{2p}\\(p\times 1)\end{bmatrix}} + \dots + \begin{bmatrix}X_{n1}\\X_{n2}\\\vdots\\X_{np}\\(p\times 1)\end{bmatrix}\right)\right]}_{(p\times 1)} \\ &= \underbrace{\mathbf{c}'_{(1\times p)} \cdot \left[\frac{1}{n} \begin{bmatrix}X_{11} + X_{21} + \dots + X_{n1}\\X_{12} + X_{22} + \dots + X_{n2}\\\vdots\\X_{1p} + X_{2p} + \dots + X_{np}\\(p\times 1)\end{bmatrix}}\right]}_{(p\times 1)} \end{split}$$



Theorem 4.5.2 (Sample Variance of a Linear Combination of **X**). Let random vectors \mathbf{X}_1 , \mathbf{X}_2 , ..., \mathbf{X}_n constitute a multivariate random sample defined in $(p \times 1)$ $(p \times 1)$ $(p \times 1)$ $(p \times 1)$ Definition 4.2.2. Assume the sample variance-covariance matrix $\mathbf{S}_{\mathbf{X}}$ defined in $(p \times p)$ Theorem 4.4.2 exists. Next consider a linear combination of the form $\mathbf{c'} \cdot \mathbf{X}_{(1 \times p)} \cdot \mathbf{X}_{(1 \times p)}$ with jth multivariate sample observation $\mathbf{c'} \cdot \mathbf{X}_j$ given in Definition 4.5.1. Then the unrealized sample variance of a linear combination of \mathbf{X} is defined by

the unrealized sample variance of a linear combination of $\underset{(n \times p)}{\mathbf{X}}$ is defined by

sample variance of
$$\mathbf{c}'_{(1\times p)} \cdot \mathbf{X}_{(p\times 1)} = \operatorname{var}\left(\mathbf{c}'_{(1\times p)} \cdot \mathbf{X}_{(p\times 1)}\right) = \mathbf{c}'_{(1\times p)} \cdot \mathbf{S}_{\mathbf{X}}_{(p\times p)} \cdot \mathbf{c}_{(p\times 1)}$$

[3, p. 140].

Proof. Use Definition 2.1.11 and Result 2.2.1. (d).

Since,

$$\begin{split} \left(\underbrace{\mathbf{c}'_{(1\times p)} \cdot \mathbf{X}_{j}}_{(p\times 1)} - \underbrace{\mathbf{c}'_{(1\times p)} \cdot \left(\overline{\mathbf{X}}_{j}\right)}_{(p\times 1)} \right)^{2} \\ &= \left(\underbrace{\mathbf{c}'_{(1\times p)} \cdot \left(\mathbf{X}_{j} - \overline{\mathbf{X}}_{(p\times 1)} \right)}_{(1\times p)} \right)^{2} \\ &= \underbrace{\mathbf{c}'_{(1\times p)} \cdot \left(\mathbf{X}_{j} - \overline{\mathbf{X}}_{(p\times 1)} \right) \cdot \underbrace{\mathbf{c}'_{(p\times 1)}}_{(1\times p)} \cdot \left(\mathbf{X}_{j} - \overline{\mathbf{X}}_{(p\times 1)} \right)}_{(p\times 1)} \\ &= \underbrace{\mathbf{c}'_{(1\times p)} \cdot \left(\mathbf{X}_{j} - \overline{\mathbf{X}}_{(p\times 1)} \right) \cdot \left(\mathbf{X}_{j} - \overline{\mathbf{X}}_{(p\times 1)} \right)}_{(p\times 1)} \cdot \underbrace{\mathbf{c}}_{(p\times 1)} \\ &= \underbrace{\mathbf{c}'_{(1\times p)} \cdot \left(\mathbf{X}_{j} - \overline{\mathbf{X}}_{(p\times 1)} \right) \left(\mathbf{X}_{j} - \left(\overline{\mathbf{X}}_{j} \right) \right)' \cdot \underbrace{\mathbf{c}}_{(p\times 1)} \\ &= \underbrace{\mathbf{c}'_{(1\times p)} \cdot \left(\mathbf{X}_{j} - \left(\overline{\mathbf{X}}_{j} \right) \right) \left(\mathbf{X}_{j} - \left(\overline{\mathbf{X}}_{j} \right) \right)' \cdot \underbrace{\mathbf{c}}_{(p\times 1)} \\ &= \underbrace{\mathbf{c}'_{(1\times p)} \cdot \left(\mathbf{X}_{j} - \left(\overline{\mathbf{X}}_{j} \right) \right) \left(\mathbf{X}_{j} - \left(\overline{\mathbf{X}}_{j} \right) \right)' \cdot \underbrace{\mathbf{c}}_{(p\times 1)} \\ &= \underbrace{\mathbf{c}'_{(1\times p)} \cdot \left(\mathbf{X}_{j} - \left(\overline{\mathbf{X}}_{j} \right) \right) \left(\mathbf{X}_{j} - \left(\overline{\mathbf{X}}_{j} \right) \right)' \cdot \underbrace{\mathbf{c}}_{(p\times 1)} \\ &= \underbrace{\mathbf{c}'_{(1\times p)} \cdot \left(\mathbf{X}_{j} - \left(\overline{\mathbf{X}}_{j} \right) \right) \left(\underbrace{\mathbf{X}_{j} - \left(\overline{\mathbf{X}}_{j} \right) \right)' \cdot \underbrace{\mathbf{c}}_{(p\times 1)} \\ &= \underbrace{\mathbf{c}'_{(1\times p)} \cdot \left[\frac{1}{n-1} \sum_{j=1}^{n} \left(\mathbf{X}_{j} - \left(\overline{\mathbf{x}}_{j} \right) \right) \left(\underbrace{\mathbf{X}_{j} - \left(\overline{\mathbf{x}}_{j} \right) \right)' \right) \left(\underbrace{\mathbf{c}}_{(p\times 1)} \right)' \cdot \underbrace{\mathbf{c}}_{(p\times 1)} \end{aligned}$$

$$= \mathbf{c}'_{(1 \times p)} \cdot \mathbf{S}_{\mathbf{X}} \cdot \mathbf{c}_{(p \times p)} \bullet$$

Theorem 4.5.3 (Sample Covariance of Two Linear Combinations of X). Let random

vectors \mathbf{X}_1 , \mathbf{X}_2 , ..., \mathbf{X}_n constitute a multivariate random sample defined in $(p \times 1)$ $(p \times 1)$ $(p \times 1)$

Definition 4.2.2. Assume the sample variance-covariance matrix $\mathbf{S}_{\mathbf{X}}_{(p \times p)}$ defined in

Theorem 4.4.2 *exists. Next consider two linear combinations of the form* $\mathbf{b}'_{(1 \times p)} \cdot \mathbf{X}_{(p \times 1)}$

and $\mathbf{c}'_{(1 \times p)} \cdot \mathbf{X}_{(p \times 1)}$ with jth multivariate sample observations $\mathbf{b}'_{(1 \times p)} \cdot \mathbf{X}_{j}$ and $\mathbf{b}'_{(p \times 1)} \cdot \mathbf{b}'_{(p \times 1)}$

 $\mathbf{c}'_{(1 \times p)} \cdot \mathbf{X}_{j}$, respectively, given in Definition 4.5.1. Then the unrealized sample

covariance of two linear combinations of $\underset{(n \times p)}{\mathbf{X}}$ is defined by

sample covariance of
$$\mathbf{b}'_{(1\times p)} \cdot \mathbf{X}_{(p\times 1)}$$
 and $\mathbf{c}'_{(1\times p)} \cdot \mathbf{X}_{(p\times 1)}$
= $\operatorname{cov}\left(\mathbf{b}'_{(1\times p)} \cdot \mathbf{X}_{(p\times 1)}, \mathbf{c}'_{(1\times p)} \cdot \mathbf{X}_{(p\times 1)}\right) = \mathbf{b}'_{(1\times p)} \cdot \mathbf{S}_{\mathbf{X}}_{(p\times p)} \cdot \mathbf{c}_{(p\times 1)}$

[3, pp. 140-141].

Proof. Use Definition 2.1.11 and Result 2.2.1. (d).

Since,

$$\begin{pmatrix} \mathbf{b}' \cdot \mathbf{X}_{j} - \mathbf{b}' \cdot \overline{\mathbf{X}}_{j} \\ {}_{(1 \times p)} \cdot {}_{(p \times 1)} & {}_{(1 \times p)} \cdot {}_{(p \times 1)} \end{pmatrix} \cdot \begin{pmatrix} \mathbf{c}' \cdot \mathbf{X}_{j} - \mathbf{c}' \cdot \overline{\mathbf{X}}_{j} \\ {}_{(1 \times p)} \cdot {}_{(p \times 1)} & {}_{(p \times 1)} \end{pmatrix} = \begin{pmatrix} \mathbf{b}' \cdot \mathbf{X}_{j} - \mathbf{b}' \cdot \overline{\mathbf{X}}_{j} \\ {}_{(1 \times p)} \cdot {}_{(p \times 1)} & {}_{(1 \times p)} \cdot {}_{(p \times 1)} \end{pmatrix} \cdot \begin{pmatrix} \mathbf{c}' \cdot \mathbf{X}_{j} - \mathbf{c}' \cdot \overline{\mathbf{X}}_{j} \\ {}_{(1 \times p)} \cdot {}_{(p \times 1)} & {}_{(p \times 1)} \end{pmatrix} = \begin{pmatrix} \mathbf{b}' \\ {}_{(1 \times p)} \cdot \begin{pmatrix} \mathbf{X}_{j} - \overline{\mathbf{X}} \\ {}_{(p \times 1)} \end{pmatrix} \cdot \begin{pmatrix} \mathbf{c}' \\ {}_{(1 \times p)} \cdot \begin{pmatrix} \mathbf{X}_{j} - \overline{\mathbf{X}} \\ {}_{(p \times 1)} \end{pmatrix}$$

$$= \mathbf{b}'_{(1\times p)} \cdot \left(\mathbf{X}_{j} - \overline{\mathbf{X}}\right) \cdot \left(\mathbf{X}_{j} - \overline{\mathbf{X}}\right)' \cdot \mathbf{c}_{(p\times 1)}$$

$$= \mathbf{b}'_{(1\times p)} \cdot \left(\mathbf{X}_{j} - \overline{\mathbf{X}}_{(p\times 1)}\right) \left(\mathbf{X}_{j} - \overline{\mathbf{X}}_{(p\times 1)}\right)' \cdot \mathbf{c}_{(p\times 1)}$$

$$\Rightarrow$$

$$\operatorname{cov}\left(\mathbf{b}'_{(1\times p)} \cdot \mathbf{X}_{(p\times 1)}, \mathbf{c}'_{(1\times p)} \cdot \mathbf{X}_{(p\times 1)}\right)$$

$$= \frac{1}{n-1} \sum_{j=1}^{n} \mathbf{b}'_{(1\times p)} \cdot \left(\mathbf{X}_{j} - \overline{\mathbf{X}}_{(p\times 1)}\right) \left(\mathbf{X}_{j} - \overline{\mathbf{X}}_{(p\times 1)}\right)' \cdot \mathbf{c}_{(p\times 1)}$$

$$= \mathbf{b}'_{(1\times p)} \cdot \left[\frac{1}{n-1} \sum_{j=1}^{n} \left(\mathbf{X}_{j} - \overline{\mathbf{X}}_{(p\times 1)}\right) \left(\mathbf{X}_{j} - \overline{\mathbf{X}}_{(p\times 1)}\right)'\right] \cdot \mathbf{c}_{(p\times 1)}$$

$$= \mathbf{b}'_{(1\times p)} \cdot \left[\frac{1}{n-1} \sum_{j=1}^{n} \left(\mathbf{X}_{j} - \overline{\mathbf{X}}_{(p\times 1)}\right) \left(\mathbf{X}_{j} - \overline{\mathbf{X}}_{(p\times 1)}\right)'\right] \cdot \mathbf{c}_{(p\times 1)}$$

4.5.3 q Linear Combinations

Definition 4.5.2 (*q* Linear Combinations of **X**). Let random vectors \mathbf{X}_1 , \mathbf{X}_2 , ..., \mathbf{X}_n ($p \times 1$) ($p \times 1$)

linear combinations of $\underset{(n \times p)}{\mathbf{X}}$ of the *p* population continuous random variables

 X_1, X_2, \dots, X_p with form:

$$Y_{i} = \frac{\mathbf{c}'_{i}}{(1 \times p)} \cdot \frac{\mathbf{X}}{(p \times 1)} = \begin{bmatrix} c_{i1}, c_{i2}, \dots, c_{ip} \end{bmatrix} \cdot \begin{bmatrix} X_{1} \\ X_{2} \\ \vdots \\ X_{p} \end{bmatrix} = c_{i1}X_{1} + c_{i2}X_{2} + \dots + c_{ip}X_{p}$$

for i = 1,2, ..., *q linear combinations*

$$\begin{aligned} Y_1 &= \mathbf{c}_1' \cdot \mathbf{X}_{(p \times 1)} = c_{11}X_1 + c_{12}X_2 + \dots + c_{1p}X_p \\ Y_2 &= \mathbf{c}_2' \cdot \mathbf{X}_{(p \times 1)} = c_{21}X_1 + c_{22}X_2 + \dots + c_{2p}X_p \\ &\vdots & \vdots \\ Y_q &= \mathbf{c}_{q}' \cdot \mathbf{X}_{(p \times 1)} = c_{q1}X_1 + c_{q2}X_2 + \dots + c_{qp}X_p \end{aligned}$$

[3, pp. 143-144] or in matrix notation,

$$\mathbf{Y}_{(q\times1)} = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_q \\ (q\times1) \end{bmatrix} = \begin{bmatrix} \mathbf{c}'_1 & \cdot \mathbf{X}_{(1\times p) \quad (p\times1)} \\ \mathbf{c}'_2 & \cdot \mathbf{X}_{(1\times p) \quad (p\times1)} \\ \vdots \\ \mathbf{c}'_q & \cdot \mathbf{X}_{(1\times p) \quad (p\times1)} \\ (q\times1) \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1p} \\ c_{21} & c_{22} & \cdots & c_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ c_{q1} & c_{q2} & \cdots & c_{qp} \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_p \end{bmatrix} = \mathbf{c}_{(q\times p) \quad (p\times1)} \mathbf{X}_{(p\times1)}$$

where the unrealized quantity on the jth multivariate sample observation,

j = 1, 2, ..., n, on the ith linear combination, i = 1, 2, ..., q, is

$$Y_{ji} = \mathbf{c}'_{i} \cdot \mathbf{X}_{j} = \begin{bmatrix} c_{i1}, c_{i2}, \dots, c_{ip} \end{bmatrix} \cdot \begin{bmatrix} X_{j1} \\ X_{j2} \\ \vdots \\ X_{jp} \end{bmatrix} = c_{i1}X_{j1} + c_{i2}X_{j2} + \dots + c_{ip}X_{jp}.$$

4.5.4 Sample Mean Vector for *q* Linear Combinations

Definition 4.5.3 (Sample Mean Vector for q Linear Combinations of X). Let random

vectors \mathbf{X}_1 , \mathbf{X}_2 , ..., \mathbf{X}_n constitute a multivariate random sample defined in $(p \times 1)$ $(p \times 1)$

Definition 4.2.2. Assume the sample mean vector $\overline{\mathbf{X}}_{(p \times 1)}$ defined in Theorem 4.4.1

exists. Next, consider q linear combinations of the form $Y_i = \mathbf{c}'_i \cdot \mathbf{X}_{(1 \times p)}$ with jth

multivariate sample observation $Y_{ji} = \mathbf{c}'_i \cdot \mathbf{X}_j$ given in Definition 4.5.2. Then the unrealized sample mean vector for q linear combinations of $\mathbf{X}_{(n \times p)}$ is defined by

sample mean vector of $\mathbf{Y}_{(q \times 1)}$

$$\overline{\mathbf{Y}}_{(q\times 1)} = E\left(\mathbf{Y}_{(q\times 1)}\right) = E\left(\mathbf{C}_{(q\times p)} \cdot \mathbf{X}_{(p\times 1)}\right) = \mathbf{C}_{(q\times p)} \cdot \mathbf{\overline{X}}_{(p\times 1)}$$

[3, p. 144].

Thus, the *i*th row of $\mathbf{Y}_{(q \times 1)}$ has unrealized sample mean

$$\overline{Y}_i = E(Y_i) = E\left(\mathbf{c}'_i \cdot \mathbf{X}_{(1 \times p)}\right) = \mathbf{c}'_i \cdot \mathbf{\overline{X}}_{(p \times 1)}$$

for i = 1, 2, ..., q.

4.5.5 Sample Variance-Covariance Matrix for q Linear

Combinations

Definition 4.5.4 (Sample Variance-Covariance Matrix for q Linear Combinations of

X). Let random vectors \mathbf{X}_1 , \mathbf{X}_2 , ..., \mathbf{X}_n constitute a multivariate random sample $(p \times 1)$ $(p \times 1)$ $(p \times 1)$

defined in Definition 4.2.2. Assume the sample variance-covariance matrix

 $\mathbf{S}_{\mathbf{X}}$ defined in Theorem 4.4.2 exists. Next consider q linear combinations of the $(p \times p)$

form
$$Y_i = \mathbf{c}'_i \cdot \mathbf{X}_{(1 \times p)}$$
 with *j*th multivariate sample observation $Y_{ji} = \mathbf{c}'_i \cdot \mathbf{X}_{j}_{(1 \times p)}$ (*p*×1)

given in Definition 4.5.2. Then the unrealized sample variance-covariance matrix for q linear combinations of $\underset{(n \times p)}{\mathbf{X}}$ is defined by

$$\begin{split} \mathbf{S}_{\mathbf{Y}} &= \mathbf{C}_{(q \times p)} \cdot \mathbf{S}_{\mathbf{X}} \cdot \mathbf{C}'_{(p \times p)} \\ &= \begin{bmatrix} \mathbf{c}'_1 \cdot \mathbf{S}_{\mathbf{X}} \cdot \mathbf{c}_1 & \mathbf{c}'_1 \cdot \mathbf{S}_{\mathbf{X}} \cdot \mathbf{c}_2 & \cdots & \mathbf{c}'_1 \cdot \mathbf{S}_{\mathbf{X}} \cdot \mathbf{c}_q \\ (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) \\ \mathbf{c}'_2 \cdot \mathbf{S}_{\mathbf{X}} \cdot \mathbf{c}_1 & \mathbf{c}'_2 \cdot \mathbf{S}_{\mathbf{X}} \cdot \mathbf{c}_2 & \cdots & \mathbf{c}'_2 \cdot \mathbf{S}_{\mathbf{X}} \cdot \mathbf{c}_q \\ (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{c}'_q \cdot \mathbf{S}_{\mathbf{X}} \cdot \mathbf{c}_1 & \mathbf{c}'_q \cdot \mathbf{S}_{\mathbf{X}} \cdot \mathbf{c}_2 & \cdots & \mathbf{c}'_q \cdot \mathbf{S}_{\mathbf{X}} \cdot \mathbf{c}_q \\ (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{c}'_q \cdot \mathbf{S}_{\mathbf{X}} \cdot \mathbf{c}_1 & \mathbf{c}'_q \cdot \mathbf{S}_{\mathbf{X}} \cdot \mathbf{c}_2 & \cdots & \mathbf{c}'_q \cdot \mathbf{S}_{\mathbf{X}} \cdot \mathbf{c}_q \\ (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) \\ (q \times q) & (1 \times p) & (p \times p) & (p \times 1) \end{bmatrix}$$

[3, p. 144]*.*

Thus, the *i*th row of $\underset{(q \times 1)}{\mathbf{Y}}$ has unrealized sample variance

$$\operatorname{var}(Y_i) = \frac{\mathbf{c}'_i \cdot \mathbf{S}_{\mathbf{X}}}{(1 \times p)} \cdot \frac{\mathbf{c}_i}{(p \times p)} \cdot \frac{\mathbf{c}_i}{(p \times 1)}$$

for i = 1, 2, ..., q.

And, the *i*th row and *k*th row of $\mathbf{Y}_{(p \times 1)}$ have unrealized sample covariance

 $\operatorname{cov}(Y_i, Y_k) = \frac{\mathbf{c}'_i}{(1 \times p)} \cdot \frac{\mathbf{S}_{\mathbf{X}}}{(p \times p)} \cdot \frac{\mathbf{c}_k}{(p \times 1)} = \frac{\mathbf{c}'_k}{(1 \times p)} \cdot \frac{\mathbf{S}_{\mathbf{X}}}{(p \times p)} \cdot \frac{\mathbf{c}_i}{(p \times 1)}$

for i, k = 1, 2, ..., q.

4.6 Standardized Random Samples

4.6.1 Standardized Univariate Random Sample

Definition 4.6.1 (Standardized Univariate Random Sample). *Let random variables* X_{jk} for j = 1, 2, ..., n constitute a univariate random sample defined in Definition 4.2.1. Assume the sample mean \overline{X}_k defined in Definition 4.3.1 and sample variance S_{kk} defined in Definition 4.3.2 exist. Then the standardized univariate random sample is defined by

$$Z_{jk} = \frac{X_{jk} - \bar{X}_k}{\sqrt{S_{kk}}}$$

for j = 1, 2, ..., n. Hence, $Z_{1k}, Z_{2k}, ..., Z_{nk}$ constitute standardized univariate random sample of size n.

Definition 4.6.2 (Standardized Sample Characteristic Vector). Let X_k be a sample $_{(n \times 1)}^{(n \times 1)}$ characteristic vector defined in Definition 4.2.3. Assume the sample mean \overline{X}_k defined in Definition 4.3.1 and sample variance S_{kk} defined in Definition 4.3.2 exist. Then the unrealized standardized sample characteristic vector is defined by

$$\mathbf{Z}_{k} = \frac{\mathbf{X}_{k} - \bar{X}_{k} \cdot \mathbf{1}_{(n \times 1)}}{\sqrt{S_{kk}}} = \begin{bmatrix} \frac{X_{1k} - \bar{X}_{k}}{\sqrt{S_{kk}}} \\ \frac{X_{2k} - \bar{X}_{k}}{\sqrt{S_{kk}}} \\ \vdots \\ \frac{X_{nk} - \bar{X}_{k}}{\sqrt{S_{kk}}} \end{bmatrix} = \begin{bmatrix} Z_{1k} \\ Z_{2k} \\ \vdots \\ Z_{nk} \end{bmatrix}_{(n \times 1)}$$

for k = 1, 2, ..., p [3, p. 135].

4.6.2 Standardized Multivariate Random Sample

Theorem 4.6.1 (Standardized Multivariate Random Sample). Let random vectors

$$\mathbf{X}_{j}_{(p\times 1)} = \begin{bmatrix} X_{j1} \\ X_{j2} \\ \vdots \\ X_{jp} \\ (p\times 1) \end{bmatrix}$$

for j = 1, 2, ..., n constitute a multivariate random sample defined in Definition 4.2.2. Assume the sample mean vector $\overline{\mathbf{X}}_{(p \times 1)}$ defined in Theorem 4.4.1 and inverse sample standard deviation matrix $\mathbf{D}_{(p \times p)}^{-1/2}$ defined in Definition 4.4.1 exist. Then the

standardized multivariate random sample is defined by

$$\mathbf{Z}_{j} = \mathbf{D}_{(p \times p)}^{-1/2} \cdot \left(\mathbf{X}_{j} - \mathbf{\overline{X}}_{(p \times 1)} \right) = \begin{bmatrix} \frac{X_{j1} - X_{1}}{\sqrt{S_{11}}} \\ \frac{X_{j2} - \bar{X}_{2}}{\sqrt{S_{22}}} \\ \vdots \\ \frac{X_{jp} - \bar{X}_{p}}{\sqrt{S_{pp}}} \end{bmatrix} = \begin{bmatrix} Z_{j1} \\ Z_{j2} \\ \vdots \\ Z_{jp} \end{bmatrix}$$
(p×1)

for j = 1, 2, ..., n. *Hence, random vectors* \mathbf{Z}_1 , \mathbf{Z}_2 , ..., \mathbf{Z}_n *constitute a standardized* $(p \times 1)$ $(p \times 1)$

multivariate random sample of size n [3, p. 449].

Proof. Use Definition 2.2.5.

$$\mathbf{Z}_{j}_{(p\times 1)} = \mathbf{D}_{(p\times p)}^{-1/2} \cdot \left(\mathbf{X}_{j}_{(p\times 1)} - \overline{\mathbf{X}}_{(p\times 1)} \right)$$

$$= \begin{bmatrix} \frac{1}{\sqrt{S_{11}}} & 0 & \cdots & 0\\ 0 & \frac{1}{\sqrt{S_{22}}} & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots\\ 0 & 0 & \cdots & \frac{1}{\sqrt{S_{pp}}} \end{bmatrix} \cdot \begin{pmatrix} \begin{bmatrix} X_{j1} \\ X_{j2} \\ \vdots \\ X_{jp} \end{bmatrix} - \begin{bmatrix} \bar{X}_1 \\ \bar{X}_2 \\ \vdots \\ \bar{X}_p \end{bmatrix} \\ (p \times 1) \end{pmatrix}$$

$$= \begin{bmatrix} \frac{1}{\sqrt{S_{11}}} & 0 & \cdots & 0\\ 0 & \frac{1}{\sqrt{S_{22}}} & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots\\ 0 & 0 & \cdots & \frac{1}{\sqrt{S_{pp}}} \end{bmatrix} \begin{bmatrix} X_{j1} - \bar{X}_1\\ X_{j2} - \bar{X}_2\\ \vdots\\ X_{jp} - \bar{X}_p \end{bmatrix}_{(p \times 1)}$$

$$= \begin{bmatrix} \frac{X_{j1} - \bar{X}_1}{\sqrt{S_{11}}} \\ \frac{X_{j2} - \bar{X}_2}{\sqrt{S_{22}}} \\ \vdots \\ \frac{X_{jp} - \bar{X}_p}{\sqrt{S_{pp}}} \\ \frac{\sqrt{S_{pp}}}{(p \times 1)} \end{bmatrix}$$

$$= \begin{bmatrix} Z_{j1} \\ Z_{j2} \\ \vdots \\ Z_{jp} \end{bmatrix}$$
for $j = 1, 2, ..., n \blacksquare$

4.6.3 Standardized Multivariate Random Sample Matrix

Definition 4.6.3 (Standardized Multivariate Random Sample Matrix). *A* **standardized multivariate random sample matrix** *is a matrix whose row vectors are transposed unrealized standardized multivariate random sample observations*

$$\mathbf{Z}'_{j} = \begin{bmatrix} Z_{j1}, Z_{j2}, \dots, Z_{jp} \end{bmatrix}$$
(1×p)

for j = 1, 2, ..., n defined in Theorem 4.6.1. In addition, the column vectors of the matrix are unrealized standardized sample variables or characteristic vectors

$$\mathbf{Z}_{k} = \begin{bmatrix} Z_{1k} \\ Z_{2k} \\ \vdots \\ Z_{nk} \end{bmatrix}_{(n \times 1)}$$

for k = 1, 2, ..., p defined in Definition 4.6.2. Let the (j, k)th entry be a standardized continuous random variable Z_{jk} , then the $n \times p$ standardized multivariate random sample matrix $\mathbf{Z}_{(n \times p)} = \{Z_{jk}\}$ is defined by

$$\mathbf{Z}_{(n \times p)} = \begin{bmatrix} Z_{11} & Z_{12} & \cdots & Z_{1k} & \cdots & Z_{1p} \\ Z_{21} & Z_{22} & \cdots & Z_{2k} & \cdots & Z_{2p} \\ \vdots & \vdots & & \vdots & & \vdots \\ Z_{j1} & Z_{j2} & \cdots & Z_{jk} & \cdots & Z_{jp} \\ \vdots & \vdots & & \vdots & & \vdots \\ Z_{n1} & Z_{n2} & \cdots & Z_{nk} & \cdots & Z_{np} \end{bmatrix}$$

$$= \begin{bmatrix} \frac{X_{11} - \overline{X}_1}{\sqrt{S_{11}}} & \frac{X_{12} - \overline{X}_2}{\sqrt{S_{22}}} & \cdots & \frac{X_{1k} - \overline{X}_k}{\sqrt{S_{kk}}} & \cdots & \frac{X_{1p} - \overline{X}_p}{\sqrt{S_{pp}}} \\ \frac{X_{21} - \overline{X}_1}{\sqrt{S_{11}}} & \frac{X_{22} - \overline{X}_2}{\sqrt{S_{22}}} & \cdots & \frac{X_{2k} - \overline{X}_k}{\sqrt{S_{kk}}} & \cdots & \frac{X_{2p} - \overline{X}_p}{\sqrt{S_{pp}}} \\ \vdots & \vdots & & \vdots & & \vdots \\ \frac{X_{j1} - \overline{X}_1}{\sqrt{S_{11}}} & \frac{X_{j2} - \overline{X}_2}{\sqrt{S_{22}}} & \cdots & \frac{X_{jk} - \overline{X}_k}{\sqrt{S_{kk}}} & \cdots & \frac{X_{jp} - \overline{X}_p}{\sqrt{S_{pp}}} \\ \vdots & & \vdots & & \vdots \\ \frac{X_{n1} - \overline{X}_1}{\sqrt{S_{11}}} & \frac{X_{n2} - \overline{X}_2}{\sqrt{S_{22}}} & \cdots & \frac{X_{nk} - \overline{X}_k}{\sqrt{S_{kk}}} & \cdots & \frac{X_{np} - \overline{X}_p}{\sqrt{S_{pp}}} \\ \vdots & & \vdots & & \vdots \\ \frac{X_{n1} - \overline{X}_1}{\sqrt{S_{11}}} & \frac{X_{n2} - \overline{X}_2}{\sqrt{S_{22}}} & \cdots & \frac{X_{nk} - \overline{X}_k}{\sqrt{S_{kk}}} & \cdots & \frac{X_{np} - \overline{X}_p}{\sqrt{S_{pp}}} \end{bmatrix} = \begin{bmatrix} \mathbf{Z}_1' \\ \mathbf{Z}_2' \\ \vdots \\ \mathbf{Z}_1' \\ \vdots \\ \mathbf{Z}_n' \\ \mathbf{Z}_n' \end{bmatrix}$$

for j = 1, 2, ..., n standardized multivariate sample observations and k = 1, 2, ..., pstandardized sample characteristics [3, p. 450].

4.7 Sample Statistics for Standardized Samples

Theorem 4.7.1 (Sample Mean for Z_k). Let Z_{jk} for j = 1, 2, ..., n constitute a standardized univariate random sample defined in Definition 4.6.1. Then the unrealized sample mean for Z_k , \overline{Z}_k , is defined by

$$\bar{Z}_{k} = \frac{1}{n} \sum_{j=1}^{n} Z_{jk} = \frac{1}{n} \sum_{j=1}^{n} \frac{X_{jk} - \bar{X}_{k}}{\sqrt{S_{kk}}} = 0$$

for k = 1, 2, ..., p.

Proof.

 \bar{Z}_k

$$=\frac{1}{n}\sum_{j=1}^{n}Z_{jk}$$

$$= \frac{1}{n} \sum_{j=1}^{n} \frac{X_{jk} - \overline{X}_k}{\sqrt{S_{kk}}}$$
$$= \frac{1}{\sqrt{S_{kk}}} \left[\frac{1}{n} \sum_{j=1}^{n} (X_{jk} - \overline{X}_k) \right]$$
$$= \frac{1}{\sqrt{S_{kk}}} \left[\frac{1}{n} \sum_{j=1}^{n} X_{jk} - \frac{1}{n} \sum_{j=1}^{n} \overline{X}_k \right]$$
$$= \frac{1}{\sqrt{S_{kk}}} \left[\frac{1}{n} \cdot (n \cdot \overline{X}_k) - \frac{1}{n} (n \cdot \overline{X}_k) \right]$$
$$= \frac{1}{\sqrt{S_{kk}}} [\overline{X}_k - \overline{X}_k] = 0 \quad \blacksquare$$

Theorem 4.7.2 (Sample Variance for Z_k). Let Z_{jk} for j = 1, 2, ..., n constitute a standardized univariate random sample defined in Definition 4.6.1. Then the unrealized sample variance for Z_k , $S_{z,kk}$, is defined by

$$S_{z,kk} = \frac{1}{n-1} \sum_{j=1}^{n} (Z_{jk} - \bar{Z}_k)^2 = \frac{1}{n-1} \sum_{j=1}^{n} \left(\frac{X_{jk} - \bar{X}_k}{\sqrt{S_{kk}}} \right)^2 = 1$$

for k = 1, 2, ..., p.

 $S_{z,kk}$

$$= \frac{1}{n-1} \sum_{j=1}^{n} (Z_{jk} - \bar{Z}_k)^2$$
$$= \frac{1}{n-1} \sum_{j=1}^{n} \left(\frac{X_{jk} - \bar{X}_k}{\sqrt{S_{kk}}} - 0 \right)^2$$
$$= \frac{1}{n-1} \sum_{j=1}^{n} \left(\frac{X_{jk} - \bar{X}_k}{\sqrt{S_{kk}}} \right)^2$$

$$= \frac{1}{S_{kk}} \cdot \left[\frac{1}{n-1} \sum_{j=1}^{n} (X_{jk} - \overline{X}_k)^2 \right]$$
$$= \frac{1}{S_{kk}} \cdot S_{kk} = 1 \blacksquare$$

Theorem 4.7.3 (Sample Covariance for Z_i and Z_k). Let $\mathbf{Z}_j = \begin{bmatrix} X_{ji} \\ X_{jk} \\ (2 \times 1) \end{bmatrix}$ for j = 1, 2, ..., n

constitute a two-dimensional characteristic subset of the standardized multivariance random sample defined in Theorem 4.6.1. Assume the sample means \bar{Z}_i, \bar{Z}_k defined in Theorem 4.7.1 and sample variances $S_{z,ii}, S_{z,kk}$ defined in Theorem 4.7.2 exist. Then the unrealized sample covariance for Z_i and $Z_k, S_{z,ik}$, is defined by

$$S_{z,ik} = \frac{1}{n-1} \sum_{j=1}^{n} (Z_{ji} - \bar{Z}_i)(Z_{jk} - \bar{Z}_k) = \frac{1}{n-1} \sum_{j=1}^{n} \left(\frac{X_{ji} - \bar{X}_i}{\sqrt{S_{ii}}}\right) \left(\frac{X_{jk} - \bar{X}_k}{\sqrt{S_{kk}}}\right) = R_{ik}$$

for i, k = 1, 2, ..., p.

Proof.

 $S_{z,ik}$

$$= \frac{1}{n-1} \sum_{j=1}^{n} (Z_{ji} - \overline{Z}_i) (Z_{jk} - \overline{Z}_k)$$
$$= \frac{1}{n-1} \sum_{j=1}^{n} \left(\frac{X_{ji} - \overline{X}_i}{\sqrt{S_{ii}}} - 0 \right) \left(\frac{X_{jk} - \overline{X}_k}{\sqrt{S_{kk}}} - 0 \right)$$
$$= \frac{1}{n-1} \sum_{j=1}^{n} \left(\frac{X_{ji} - \overline{X}_i}{\sqrt{S_{ii}}} \right) \left(\frac{X_{jk} - \overline{X}_k}{\sqrt{S_{kk}}} \right)$$

$$= \frac{1}{\sqrt{S_{ii}}\sqrt{S_{kk}}} \cdot \left[\frac{1}{n-1} \sum_{j=1}^{n} (X_{ji} - \overline{X}_i) (X_{jk} - \overline{X}_k)\right]$$
$$= \frac{1}{\sqrt{S_{ii}}\sqrt{S_{kk}}} \cdot S_{ik} = \frac{S_{ik}}{\sqrt{S_{ii}}\sqrt{S_{kk}}} = R_{ik} \blacksquare$$
Note $S_{z,ik} = S_{z,ki}$, and when $i = k, S_{z,kk} = R_{kk} = 1$.

4.8 Sample Mean Vector and Variance-Covariance

Matrix for Standardized Samples

4.8.1 Sample Mean Vector for Standardized Samples

Theorem 4.8.1 (Sample Mean Vector for **Z**). Let \mathbf{Z}_1 , \mathbf{Z}_2 , ..., \mathbf{Z}_n constitute a standardized multivariate random sample defined in Theorem 4.6.1. Then the $p \times 1$ unrealized sample mean vector for $\mathbf{Z}_{(n \times p)}$ is defined by

$$\bar{\mathbf{Z}}_{(p\times1)} = \frac{1}{n} \sum_{j=1}^{n} \mathbf{Z}_{j} = \frac{1}{n} \cdot \frac{\mathbf{Z}'}{(p\times n)} \cdot \frac{1}{(n\times1)} = \begin{bmatrix} \bar{Z}_{1} \\ \bar{Z}_{2} \\ \vdots \\ \bar{Z}_{p} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ (p\times1) \end{bmatrix} = \mathbf{0}_{(p\times1)}$$

[3, p. 450].

$$\begin{split} \bar{\mathbf{Z}}_{(p \times 1)} \\ &= \frac{1}{n} \sum_{j=1}^{n} \mathbf{Z}_{j} \\ &= \frac{1}{n} \left(\mathbf{Z}_{1} + \mathbf{Z}_{2} + \dots + \mathbf{Z}_{n} \right) \\ &= \frac{1}{n} \left(\begin{bmatrix} Z_{11} \\ Z_{12} \\ \vdots \\ Z_{1p} \end{bmatrix} + \begin{bmatrix} Z_{21} \\ Z_{22} \\ \vdots \\ Z_{2p} \end{bmatrix} + \dots + \begin{bmatrix} Z_{n1} \\ Z_{n2} \\ \vdots \\ Z_{2p} \end{bmatrix} \right) \\ &= \frac{1}{n} \begin{bmatrix} Z_{11} + Z_{21} + \dots + Z_{n1} \\ Z_{12} + Z_{22} + \dots + Z_{n2} \\ \vdots \\ Z_{1p} + Z_{2p} + \dots + Z_{np} \end{bmatrix} \\ &= \frac{1}{n} \begin{bmatrix} \sum_{j=1}^{n} Z_{j1} \\ \vdots \\ \sum_{j=1}^{n} Z_{j2} \\ \vdots \\ \sum_{j=1}^{n} Z_{jp} \end{bmatrix} \\ &= \begin{bmatrix} \frac{1}{n} \sum_{j=1}^{n} Z_{j1} \\ \vdots \\ \sum_{j=1}^{n} Z_{j2} \\ \vdots \\ \vdots \\ \vdots \\ \sum_{j=1}^{n} Z_{j2} \\ \vdots \\ \vdots \\ \frac{1}{n} \sum_{j=1}^{n} Z_{jp} \end{bmatrix} \end{split}$$

$$= \begin{bmatrix} \bar{Z}_1 \\ \bar{Z}_2 \\ \vdots \\ \bar{Z}_p \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} = \mathbf{0}_{(p \times 1)}$$

In terms of matrix operations $\bar{\mathbf{Z}}_{(p imes 1)}$ can be obtained by

$$\begin{split} \bar{\mathbf{Z}}_{(p\times1)} &= \frac{1}{n} \cdot \underbrace{\mathbf{Z}'}_{(p\timesn)} \cdot \underbrace{\mathbf{1}}_{(n\times1)} \\ &= \frac{1}{n} \begin{bmatrix} Z_{11} & Z_{21} & \cdots & Z_{n1} \\ Z_{12} & Z_{22} & \cdots & Z_{n2} \\ \vdots & \vdots & \ddots & \vdots \\ Z_{1p} & Z_{2p} & \cdots & Z_{np} \end{bmatrix} \cdot \begin{bmatrix} 1_1 \\ 1_2 \\ \vdots \\ 1_n \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ Z_{1p} & Z_{2p} & \cdots & Z_{np} \end{bmatrix} \\ &= \frac{1}{n} \begin{bmatrix} Z_{11} + Z_{21} + \cdots + Z_{n1} \\ Z_{12} + Z_{22} + \cdots + Z_{n2} \\ \vdots \\ Z_{1p} + Z_{2p} + \cdots + Z_{np} \end{bmatrix} \\ &= \begin{bmatrix} \sum_{i=1}^{n} Z_{i1} \\ \sum_{i=1}^{n} Z_{j2} \\ \vdots \\ \sum_{i=1}^{n} Z_{jp} \end{bmatrix} = \begin{bmatrix} \frac{1}{n} \sum_{j=1}^{n} Z_{j1} \\ \frac{1}{n} \sum_{j=1}^{n} Z_{j2} \\ \vdots \\ \frac{1}{n} \sum_{i=1}^{n} Z_{jp} \end{bmatrix} \\ &= \begin{bmatrix} \overline{Z}_1 \\ \overline{Z}_2 \\ \vdots \\ \overline{Z}_p \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ \vdots \\ 0 \\ (p\times1) \end{bmatrix} = \begin{bmatrix} 0 \\ (p\times1) \end{bmatrix} \bullet$$

4.8.2 Sample Variance-Covariance Matrix for Standardized Samples

Theorem 4.8.2 (Sample Variance-Covariance Matrix for **Z**). Let \mathbf{Z}_1 , \mathbf{Z}_2 , ..., \mathbf{Z}_n ($p \times 1$) *constitute a standardized multivariate random sample defined in Theorem* 4.6.1. Assume the sample mean vector $\mathbf{\overline{Z}}_{(p \times 1)}$ defined in Theorem 4.8.1 exists.

Then the $p \times p$ symmetric unrealized **sample variance-covariance matrix for** $\underset{(n \times p)}{\mathbf{Z}}$ *is*

$$\begin{split} \mathbf{S}_{\mathbf{Z}} &= \frac{1}{n-1} \sum_{j=1}^{n} \left(\mathbf{Z}_{j} - \bar{\mathbf{Z}}_{(p\times 1)} \right) \left(\mathbf{Z}_{j} - \bar{\mathbf{Z}}_{(p\times 1)} \right)' \\ &= \frac{1}{n-1} \cdot \left(\mathbf{Z}_{(n\times p)} - \frac{1}{n} \cdot \frac{1}{(n\times 1)} \cdot \frac{1'}{(1\times n)} \cdot \frac{\mathbf{Z}_{p}}{(n\times p)} \right)' \cdot \left(\mathbf{Z}_{(n\times p)} - \frac{1}{n} \cdot \frac{1}{(n\times 1)} \cdot \frac{1'}{(1\times n)} \cdot \frac{\mathbf{Z}_{p}}{(n\times p)} \right) \\ &= \frac{1}{n-1} \cdot \left(\mathbf{Z}_{(n\times p)} - \frac{1}{n} \cdot \frac{\mathbf{Z}_{p}}{(n\times 1)} \right)' \cdot \left(\mathbf{Z}_{(n\times p)} - \frac{1}{n} \cdot \frac{\mathbf{Z}_{p}}{(1\times p)} \right) = \frac{1}{n-1} \cdot \mathbf{Z}_{p\times n}' \cdot \mathbf{Z}_{p\times p} = \mathbf{R}_{p\times p} \\ &[3, p. 450]. \end{split}$$

Proof. Use Definition 2.1.4, Definition 2.2.2, Definition 2.2.3, Theorem 4.6.1, Theorem 4.7.2, and Theorem 4.7.3.

$$\begin{split} \mathbf{S}_{\mathbf{Z}} &= \frac{1}{n-1} \sum_{j=1}^{n} \left(\mathbf{Z}_{j} - \bar{\mathbf{Z}}_{(p\times 1)} \right) \left(\mathbf{Z}_{j} - \bar{\mathbf{Z}}_{(p\times 1)} \right)' \\ &= \frac{1}{n-1} \sum_{j=1}^{n} \left(\mathbf{Z}_{j} - \mathbf{0}_{(p\times 1)} \right) \left(\mathbf{Z}_{j} - \mathbf{0}_{(p\times 1)} \right)' \\ &= \frac{1}{n-1} \sum_{j=1}^{n} \left(\mathbf{Z}_{j} - \mathbf{0}_{(p\times 1)} \right) \left(\mathbf{Z}_{j} - \mathbf{0}_{(p\times 1)} \right)' \\ &= \frac{1}{n-1} \sum_{j=1}^{n} \mathbf{Z}_{j} \cdot \mathbf{Z}_{j}' = \frac{1}{n-1} \sum_{j=1}^{n} \left[\frac{Z_{j1}}{Z_{j2}} \right]_{\substack{(Z_{j1}, Z_{j2}, \dots, Z_{jp}] \\ (p\times 1)}} [Z_{j1}, Z_{j2}, \dots, Z_{jp}] \end{split}$$

$$=\frac{1}{n-1}\sum_{j=1}^{n} \left(\begin{bmatrix} \frac{X_{j1} - \bar{X}_{1}}{\sqrt{S_{11}}} \\ \frac{X_{j2} - \bar{X}_{2}}{\sqrt{S_{22}}} \\ \vdots \\ \frac{X_{jp} - \bar{X}_{p}}{\sqrt{S_{pp}}} \end{bmatrix} \begin{bmatrix} \frac{X_{j1} - \bar{X}_{1}}{\sqrt{S_{11}}}, \frac{X_{j2} - \bar{X}_{2}}{\sqrt{S_{22}}}, \dots, \frac{X_{jp} - \bar{X}_{p}}{\sqrt{S_{pp}}} \end{bmatrix} \right)$$

$$= \begin{bmatrix} \frac{\sum_{j=1}^{n} (X_{j1} - \bar{X}_{1})^{2}}{\sqrt{S_{11}}\sqrt{S_{11}}} & \frac{\sum_{j=1}^{n} (X_{j1} - \bar{X}_{1})(X_{j2} - \bar{X}_{2})}{\sqrt{S_{11}}\sqrt{S_{22}}} & \dots & \frac{\sum_{j=1}^{n} (X_{j1} - \bar{X}_{1})(X_{jp} - \bar{X}_{p})}{\sqrt{S_{11}}\sqrt{S_{pp}}} \\ \frac{\sum_{j=1}^{n} (X_{j2} - \bar{X}_{2})(X_{j1} - \bar{X}_{1})}{\sqrt{S_{22}}\sqrt{S_{11}}} & \frac{\sum_{j=1}^{n} (X_{j2} - \bar{X}_{2})^{2}}{\sqrt{S_{22}}\sqrt{S_{22}}} & \dots & \frac{\sum_{j=1}^{n} (X_{j2} - \bar{X}_{2})(X_{jp} - \bar{X}_{p})}{\sqrt{S_{22}}\sqrt{S_{pp}}} \\ \frac{\sqrt{S_{22}}\sqrt{S_{11}}}{n-1} & \frac{\sqrt{S_{22}}\sqrt{S_{22}}}{n-1} & \dots & \frac{N-1}{n-1} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\sum_{j=1}^{n} (X_{jp} - \bar{X}_{p})(X_{j1} - \bar{X}_{1})}{\sqrt{S_{pp}}\sqrt{S_{11}}} & \frac{\sum_{j=1}^{n} (X_{jp} - \bar{X}_{p})(X_{j2} - \bar{X}_{2})}{\sqrt{S_{pp}}\sqrt{S_{22}}} & \dots & \frac{\sum_{j=1}^{n} (X_{jp} - \bar{X}_{p})^{2}}{\sqrt{S_{pp}}\sqrt{S_{pp}}} \\ \frac{\sqrt{S_{pp}}\sqrt{S_{pp}}}{n-1} & \frac{\sqrt{S_{pp}}\sqrt{S_{pp}}}{n-1} & \dots & \frac{N-1}{n-1} \end{bmatrix}$$

$$= \begin{bmatrix} \frac{(n-1)S_{11}}{\sqrt{S_{11}}\sqrt{S_{11}}} & \frac{(n-1)S_{12}}{\sqrt{S_{11}}\sqrt{S_{22}}} & \dots & \frac{(n-1)S_{1p}}{\sqrt{S_{11}}\sqrt{S_{pp}}} \\ \frac{(n-1)S_{21}}{n-1} & \frac{(n-1)S_{22}}{\sqrt{S_{22}}\sqrt{S_{22}}} & \dots & \frac{(n-1)S_{2p}}{\sqrt{S_{22}}\sqrt{S_{pp}}} \\ \frac{(n-1)S_{p1}}{n-1} & \frac{(n-1)S_{p2}}{n-1} & \dots & \frac{(n-1)S_{pp}}{\sqrt{S_{2p}}\sqrt{S_{pp}}} \\ \frac{\sqrt{S_{pp}}\sqrt{S_{11}}}{n-1} & \frac{\sqrt{S_{pp}}\sqrt{S_{22}}}{n-1} & \dots & \frac{\sqrt{S_{pp}}\sqrt{S_{pp}}}{n-1} \end{bmatrix}$$

$$\{S_{ik} = \frac{1}{n-1} \sum_{j=1}^{n} (X_{ji} - \bar{X}_i) (X_{jk} - \bar{X}_k) \Rightarrow (n-1)S_{ik} = \sum_{j=1}^{n} (X_{ji} - \bar{X}_i) (X_{jk} - \bar{X}_k) \text{ for } i, k = 1, 2, \dots, p\}$$

$$= \begin{bmatrix} \frac{S_{11}}{\sqrt{S_{11}}\sqrt{S_{11}}} & \frac{S_{12}}{\sqrt{S_{11}}\sqrt{S_{22}}} & \cdots & \frac{S_{1p}}{\sqrt{S_{11}}\sqrt{S_{pp}}} \\ \frac{S_{21}}{\sqrt{S_{22}}\sqrt{S_{11}}} & \frac{S_{22}}{\sqrt{S_{22}}\sqrt{S_{22}}} & \cdots & \frac{S_{2p}}{\sqrt{S_{22}}\sqrt{S_{pp}}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{S_{p1}}{\sqrt{S_{pp}}\sqrt{S_{11}}} & \frac{S_{p2}}{\sqrt{S_{pp}}\sqrt{S_{22}}} & \cdots & \frac{S_{pp}}{\sqrt{S_{pp}}\sqrt{S_{pp}}} \end{bmatrix}$$
$$= \begin{bmatrix} 1 & R_{12} & \cdots & R_{1p} \\ R_{21} & 1 & \cdots & R_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ R_{p1} & R_{p2} & \cdots & 1 \end{bmatrix}$$
$$= \begin{bmatrix} \mathbf{R} \\ (p \times p) \end{bmatrix}$$

In terms of matrix operations $\mathbf{S_Z}_{(p\times p)} = \underset{(p\times p)}{\mathbf{R}}$ can be obtained by

$$\mathbf{S}_{\mathbf{Z}} = \frac{1}{n-1} \cdot \left(\mathbf{Z}_{(n \times p)} - \frac{1}{n} \cdot \mathbf{1}_{(n \times 1)} \cdot \mathbf{1}_{(1 \times n)} \cdot \mathbf{Z}_{(n \times p)} \right)' \cdot \left(\mathbf{Z}_{(n \times p)} - \frac{1}{n} \cdot \mathbf{1}_{(n \times 1)} \cdot \mathbf{1}_{(1 \times n)} \cdot \mathbf{Z}_{(n \times p)} \right)$$

where

$$\begin{split} \frac{1}{n} \cdot \underbrace{\mathbf{1}}_{(n\times1)} \cdot \underbrace{\mathbf{1}'}_{(1\times n)} \cdot \underbrace{\mathbf{Z}}_{(n\times p)} \\ &= \frac{1}{n} \cdot \begin{bmatrix} 1_1 \\ 1_2 \\ \vdots \\ 1_n \end{bmatrix} \cdot \begin{bmatrix} 1_1, 1_2, \dots, 1_n \end{bmatrix} \cdot \begin{bmatrix} Z_{11} & Z_{12} & \cdots & Z_{1p} \\ Z_{21} & Z_{22} & \cdots & Z_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ Z_{n1} & Z_{n2} & \cdots & Z_{np} \end{bmatrix} \\ &= \begin{bmatrix} 1_1 \\ 1_2 \\ \vdots \\ 1_n \\ (n\times1) \end{bmatrix} \cdot \frac{1}{n} \cdot \begin{bmatrix} \sum_{j=1}^n Z_{j1}, \sum_{j=1}^n Z_{j2}, \dots, \sum_{j=1}^n Z_{jp} \end{bmatrix} \\ &= \begin{bmatrix} 1_1 \\ 1_2 \\ \vdots \\ 1_n \end{bmatrix} \cdot \frac{1}{n} \cdot \begin{bmatrix} \sum_{j=1}^n X_{j1} - \overline{X}_1 \\ (1\times p) \end{bmatrix} \cdot \sum_{j=1}^n Z_{jp} \end{bmatrix}$$

$$= \begin{bmatrix} 1_{1} \\ 1_{2} \\ \vdots \\ 1_{n} \\ (n \times 1) \end{bmatrix} \cdot \begin{bmatrix} \frac{1}{n} \sum_{j=1}^{n} \frac{X_{j1} - \bar{X}_{1}}{\sqrt{S_{11}}}, \frac{1}{n} \sum_{j=1}^{n} \frac{X_{j2} - \bar{X}_{2}}{\sqrt{S_{22}}}, \dots, \frac{1}{n} \sum_{j=1}^{n} \frac{X_{jp} - \bar{X}_{p}}{\sqrt{S_{pp}}} \end{bmatrix}$$

$$= \begin{bmatrix} 1_{1} \\ 1_{2} \\ \vdots \\ 1_{n} \\ (n \times 1) \end{bmatrix} \cdot \begin{bmatrix} \bar{Z}_{1}, \bar{Z}_{2}, \dots, \bar{Z}_{p} \end{bmatrix}$$

$$= \begin{bmatrix} 1 \\ n \times 1 \end{bmatrix} \cdot \begin{bmatrix} \bar{Z}_{1}, \bar{Z}_{2}, \dots, \bar{Z}_{p} \\ (1 \times p) \end{bmatrix}$$

$$= \begin{bmatrix} \bar{Z}_{1} & \bar{Z}_{2} & \cdots & \bar{Z}_{p} \\ \bar{Z}_{1} & \bar{Z}_{2} & \cdots & \bar{Z}_{p} \\ \vdots & \vdots & \ddots & \vdots \\ \bar{Z}_{1} & \bar{Z}_{2} & \cdots & \bar{Z}_{p} \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \\ (n \times p) \end{bmatrix} = \begin{bmatrix} 0 \\ (n \times p) \end{bmatrix}$$

Thus,

$$\begin{split} \mathbf{S}_{\mathbf{Z}} &= \frac{1}{n-1} \cdot \begin{pmatrix} \mathbf{Z} & -\frac{1}{n} \cdot \mathbf{1} & \mathbf{1}' & \mathbf{Z} \\ (n \times p) & -\frac{1}{n} \cdot \mathbf{1}' & (\mathbf{1} \times n) \cdot (\mathbf{1} \times n) \end{pmatrix}' \cdot \begin{pmatrix} \mathbf{Z} & -\frac{1}{n} \cdot \mathbf{1} & \mathbf{1}' & \mathbf{Z} \\ (n \times p) & -\frac{1}{n} \cdot \left(\mathbf{X} & -\frac{1}{n} \cdot \mathbf{X} & \mathbf{X} \\ (n \times p) & -\frac{1}{n} \cdot \left(\mathbf{X} & -\frac{1}{n} \cdot \mathbf{X} & \mathbf{X} \\ (n \times p) & -\frac{1}{n-1} \cdot \left(\mathbf{X} & -\frac{1}{n-1} \cdot \mathbf{X} & \mathbf{X} \\ (n \times p) & -\frac{1}{n-1} \cdot \mathbf{X} & \mathbf{X} \\ &= \frac{1}{n-1} \cdot \begin{pmatrix} \mathbf{Z} & -\mathbf{0} \\ (n \times p) & -\frac{1}{(n \times p)} \end{pmatrix}' \cdot \begin{pmatrix} \mathbf{Z} & -\mathbf{0} \\ (n \times p) & -\frac{1}{(n \times p)} \end{pmatrix} \\ &= \frac{1}{n-1} \cdot \begin{bmatrix} \mathbf{Z}_{11} & \mathbf{Z}_{21} & \cdots & \mathbf{Z}_{n1} \\ \mathbf{Z}_{12} & \mathbf{Z}_{22} & \cdots & \mathbf{Z}_{n2} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{Z}_{1p} & \mathbf{Z}_{2p} & \cdots & \mathbf{Z}_{np} \end{bmatrix} \begin{bmatrix} \mathbf{Z}_{11} & \mathbf{Z}_{12} & \cdots & \mathbf{Z}_{1p} \\ \mathbf{Z}_{11} & \mathbf{Z}_{12} & \cdots & \mathbf{Z}_{np} \\ \mathbf{Z}_{11} & \mathbf{Z}_{12} & \cdots & \mathbf{Z}_{np} \\ \mathbf{Z}_{n1} & \mathbf{Z}_{n2} & \cdots & \mathbf{Z}_{np} \end{bmatrix} \end{split}$$

$$= \frac{1}{n-1} \begin{bmatrix} \frac{X_{11} - \bar{X}_1}{\sqrt{S_{11}}} & \frac{X_{21} - \bar{X}_1}{\sqrt{S_{11}}} & \cdots & \frac{X_{n1} - \bar{X}_1}{\sqrt{S_{11}}} \\ \frac{X_{12} - \bar{X}_2}{\sqrt{S_{22}}} & \frac{X_{22} - \bar{X}_2}{\sqrt{S_{22}}} & \cdots & \frac{X_{n2} - \bar{X}_2}{\sqrt{S_{22}}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{X_{1p} - \bar{X}_p}{\sqrt{S_{pp}}} & \frac{X_{2p} - \bar{X}_p}{\sqrt{S_{pp}}} & \cdots & \frac{X_{np} - \bar{X}_p}{\sqrt{S_{pp}}} \end{bmatrix} \begin{bmatrix} \frac{X_{11} - \bar{X}_1}{\sqrt{S_{11}}} & \frac{X_{12} - \bar{X}_2}{\sqrt{S_{22}}} & \cdots & \frac{X_{1p} - \bar{X}_p}{\sqrt{S_{pp}}} \\ \frac{X_{21} - \bar{X}_1}{\sqrt{S_{22}}} & \frac{X_{22} - \bar{X}_2}{\sqrt{S_{22}}} & \cdots & \frac{X_{2p} - \bar{X}_p}{\sqrt{S_{pp}}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{X_{1p} - \bar{X}_p}{\sqrt{S_{pp}}} & \frac{X_{2p} - \bar{X}_p}{\sqrt{S_{pp}}} & \cdots & \frac{X_{np} - \bar{X}_p}{\sqrt{S_{pp}}} \end{bmatrix} \begin{bmatrix} \frac{X_{11} - \bar{X}_1}{\sqrt{S_{11}}} & \frac{X_{12} - \bar{X}_2}{\sqrt{S_{22}}} & \cdots & \frac{X_{1p} - \bar{X}_p}{\sqrt{S_{pp}}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{X_{n1} - \bar{X}_1}{\sqrt{S_{11}}} & \frac{X_{n2} - \bar{X}_2}{\sqrt{S_{22}}} & \cdots & \frac{X_{np} - \bar{X}_p}{\sqrt{S_{pp}}} \end{bmatrix}$$

$$= \begin{bmatrix} \frac{\sum_{j=1}^{n} \left(X_{j1} - \bar{X}_{1}\right)^{2}}{\sqrt{S_{11}}\sqrt{S_{11}}} & \frac{\sum_{j=1}^{n} \left(X_{j1} - \bar{X}_{1}\right) \left(X_{j2} - \bar{X}_{2}\right)}{\sqrt{S_{11}}\sqrt{S_{22}}} & \dots & \frac{\sum_{j=1}^{n} \left(X_{j1} - \bar{X}_{1}\right) \left(X_{jp} - \bar{X}_{p}\right)}{\sqrt{S_{11}}\sqrt{S_{pp}}} \\ \frac{\sum_{j=1}^{n} \left(X_{j2} - \bar{X}_{2}\right) \left(X_{j1} - \bar{X}_{1}\right)}{\sqrt{S_{22}}\sqrt{S_{11}}} & \frac{\sum_{j=1}^{n} \left(X_{j2} - \bar{X}_{2}\right)^{2}}{\sqrt{S_{22}}\sqrt{S_{22}}} & \dots & \frac{\sum_{j=1}^{n} \left(X_{j2} - \bar{X}_{2}\right) \left(X_{jp} - \bar{X}_{p}\right)}{\sqrt{S_{22}}\sqrt{S_{pp}}} \\ \frac{\sqrt{S_{22}}\sqrt{S_{11}}}{n-1} & \frac{\sqrt{S_{22}}\sqrt{S_{22}}}{n-1} & \dots & \frac{N-1}{n-1} \\ \vdots & \ddots & \vdots \\ \frac{\sum_{j=1}^{n} \left(X_{jp} - \bar{X}_{p}\right) \left(X_{j1} - \bar{X}_{1}\right)}{\sqrt{S_{pp}}\sqrt{S_{11}}} & \frac{\sum_{j=1}^{n} \left(X_{jp} - \bar{X}_{p}\right) \left(X_{j2} - \bar{X}_{2}\right)}{\sqrt{S_{pp}}\sqrt{S_{22}}} & \dots & \frac{\sum_{j=1}^{n} \left(X_{jp} - \bar{X}_{p}\right)^{2}}{\sqrt{S_{pp}}\sqrt{S_{pp}}} \\ \frac{\sqrt{S_{pp}}\sqrt{S_{pp}}}{n-1} & \frac{\sqrt{S_{pp}}\sqrt{S_{pp}}}{n-1} & \dots & \frac{\sum_{j=1}^{n} \left(X_{jp} - \bar{X}_{p}\right)^{2}}{\sqrt{S_{pp}}\sqrt{S_{pp}}} \\ \frac{\sqrt{S_{pp}}\sqrt{S_{pp}}}{n-1} & \frac{\sqrt{S_{pp}}\sqrt{S_{pp}}}{n-1} & \dots & \frac{\sum_{j=1}^{n} \left(X_{jp} - \bar{X}_{p}\right)^{2}}{\sqrt{S_{pp}}\sqrt{S_{pp}}} \\ \frac{\sqrt{S_{pp}}\sqrt{S_{pp}}}{n-1} & \frac{\sqrt{S_{pp}}\sqrt{S_{pp}}}{n-1} & \dots & \frac{\sqrt{S_{pp}}\sqrt{S_{pp}}}{n-1} \end{bmatrix}$$

$$\{S_{ik} = \frac{1}{n-1} \sum_{j=1}^{n} (X_{ji} - \bar{X}_{i}) (X_{jk} - \bar{X}_{k}) \Rightarrow (n-1)S_{ik} = \sum_{j=1}^{n} (X_{ji} - \bar{X}_{i}) (X_{jk} - \bar{X}_{k}) \text{ for } i, k = 1, 2, ..., p \}$$

$$= \begin{bmatrix} \frac{(n-1)S_{11}}{\sqrt{S_{11}}\sqrt{S_{11}}} & \frac{(n-1)S_{12}}{\sqrt{S_{11}}\sqrt{S_{22}}} & \dots & \frac{(n-1)S_{1p}}{\sqrt{S_{11}}\sqrt{S_{pp}}} \\ \frac{(n-1)S_{21}}{\sqrt{S_{22}}\sqrt{S_{11}}} & \frac{(n-1)S_{22}}{\sqrt{S_{22}}\sqrt{S_{22}}} & \dots & \frac{\sqrt{S_{22}}\sqrt{S_{pp}}}{n-1} \\ \frac{(n-1)S_{p1}}{\sqrt{S_{pp}}\sqrt{S_{11}}} & \frac{(n-1)S_{p2}}{\sqrt{S_{pp}}\sqrt{S_{22}}} & \dots & \frac{(n-1)S_{pp}}{n-1} \\ \frac{\sqrt{S_{pp}}\sqrt{S_{11}}}{n-1} & \frac{\sqrt{S_{pp}}\sqrt{S_{22}}}{n-1} & \dots & \frac{\sqrt{S_{pp}}\sqrt{S_{pp}}}{n-1} \end{bmatrix}$$

$$= \begin{bmatrix} \frac{S_{11}}{\sqrt{S_{11}}\sqrt{S_{11}}} & \frac{S_{12}}{\sqrt{S_{11}}\sqrt{S_{22}}} & \cdots & \frac{S_{1p}}{\sqrt{S_{11}}\sqrt{S_{pp}}} \\ \frac{S_{21}}{\sqrt{S_{22}}\sqrt{S_{11}}} & \frac{S_{22}}{\sqrt{S_{22}}\sqrt{S_{22}}} & \cdots & \frac{S_{2p}}{\sqrt{S_{22}}\sqrt{S_{pp}}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{S_{p1}}{\sqrt{S_{pp}}\sqrt{S_{11}}} & \frac{S_{p2}}{\sqrt{S_{pp}}\sqrt{S_{22}}} & \cdots & \frac{S_{pp}}{\sqrt{S_{pp}}\sqrt{S_{pp}}} \end{bmatrix}$$
$$= \begin{bmatrix} 1 & R_{12} & \cdots & R_{1p} \\ R_{21} & 1 & \cdots & R_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ R_{p1} & R_{p2} & \cdots & 1 \end{bmatrix}$$
$$= \begin{bmatrix} \mathbf{R}_{(p \times p)} \bullet$$

Thus, the sample variance-covariance matrix $\mathbf{S}_{\mathbf{Z}}_{(p \times p)}$ derived from matrix $\mathbf{Z}_{(n \times p)}$,

is equivalent to the sample correlation matrix $\mathbf{R}_{(p \times p)}$, derived from $\mathbf{X}_{(n \times p)}$. That is,

 $\mathbf{S}_{\mathbf{Z}} = \mathbf{R}_{(p \times p)}$. The diagonal elements of $\mathbf{S}_{\mathbf{Z}} = \mathbf{R}_{(p \times p)}$ are the sample variances

$$S_{z,kk} = \frac{S_{kk}}{\sqrt{S_{kk}}\sqrt{S_{kk}}} = \frac{S_{kk}}{S_{kk}} = R_{kk} = 1$$

for k = 1, 2, ..., p, i = k where $S_{z,ii} = S_{z,kk}$. The off-diagonal elements of

 $\mathbf{S}_{\mathbf{Z}} = \mathbf{R}_{(p \times p)}$ are the sample covariances

$$S_{z,ik} = \frac{S_{ik}}{\sqrt{S_{ii}}\sqrt{S_{kk}}} = R_{ik}$$

for $i, k = 1, 2, ..., p, i \neq k$ where $S_{z,ik} = S_{z,ki}$.

Furthermore,

$$tr(\mathbf{R}) = \sum_{k=1}^{p} R_{kk} = tr(\mathbf{S}_{\mathbf{Z}}) = \sum_{k=1}^{p} S_{z,kk} = 1 + 1 + \dots + 1 = p$$

(total standardized sample variance).

4.9 Sample Mean Vector and Variance-Covariance Matrix for Linear Combinations of Standardized Samples

4.9.1 Linear Combination of Standardized Samples

Definition 4.9.1 (Linear Combination of **Z**). Let $\underset{(p \times 1)}{\mathbf{c}}$ be a $p \times 1$ vector of constants defined as

$$\mathbf{c}_{(p\times 1)} = \begin{bmatrix} c_1\\c_2\\\vdots\\c_p\\(p\times 1) \end{bmatrix}$$

and let $\sum_{(p \times 1)} be a p \times 1$ population random vector of standardized continuous random variables

$$\mathbf{Z}_{(p \times 1)} = \begin{bmatrix} Z_1 \\ Z_2 \\ \vdots \\ Z_p \end{bmatrix}_{(p \times 1)}$$

Now consider a **linear combination of** $\underset{(n \times p)}{\mathbf{Z}}$ *with form*

$$\mathbf{c}'_{(1\times p)} \cdot \mathbf{Z}_{(p\times 1)} = \begin{bmatrix} c_1, c_2, \dots, c_p \end{bmatrix} \cdot \begin{bmatrix} Z_1 \\ Z_2 \\ \vdots \\ Z_p \end{bmatrix} = c_1 Z_1 + c_2 Z_2 + \dots + c_p Z_p$$

$$(p\times 1)$$

whose unrealized quantity on the jth standardized multivariate sample observation

is

$$\mathbf{c}'_{(1\times p)} \cdot \mathbf{Z}_{j} = \begin{bmatrix} c_1, c_2, \dots, c_p \end{bmatrix} \cdot \begin{bmatrix} Z_{j_1} \\ Z_{j_2} \\ \vdots \\ Z_{jp} \end{bmatrix} = c_1 Z_{j_1} + c_2 Z_{j_2} + \dots + c_p Z_{jp}$$

for j = 1, 2, ..., n.

4.9.2 Sample Statistics for Linear Combinations of

Standardized Samples

Definition 4.9.2 (Sample Mean for a Linear Combination of **Z**). Let \mathbf{Z}_{1} , \mathbf{Z}_{2} , ..., \mathbf{Z}_{n} ($p \times 1$) ($p \times 1$) ($p \times 1$) constitute a standardized multivariate random sample defined in Theorem 4.6.1. Assume the sample mean vector $\mathbf{Z}_{(p \times 1)}$ defined in Theorem 4.8.1 exists. Next, consider a linear combination of the form $\mathbf{c'}_{(1 \times p)} \cdot \mathbf{Z}_{(p \times 1)}$ with jth standardized multivariate sample observation $\mathbf{c'}_{(1 \times p)} \cdot \mathbf{Z}_{j}$ given in Definition 4.9.1. Then the unrealized sample mean for a linear combination of $\mathbf{Z}_{(n \times p)}$ is defined by

sample mean of
$$\mathbf{c}'_{(1\times p)} \cdot \mathbf{Z}_{(p\times 1)} = E\left(\mathbf{c}'_{(1\times p)} \cdot \mathbf{Z}_{(p\times 1)}\right) = \mathbf{c}'_{(1\times p)} \cdot \mathbf{\bar{Z}}_{(p\times 1)} = 0$$

Definition 4.9.3 (Sample Variance for a Linear Combination of Z). Let

 \mathbf{Z}_1 , \mathbf{Z}_2 , ..., \mathbf{Z}_n constitute a standardized multivariate random sample defined in $(p \times 1)$ $(p \times 1)$ $(p \times 1)$

Theorem 4.6.1. *Assume the sample variance-covariance matrix* $\mathbf{S}_{\mathbf{Z}} = \mathbf{R}_{(p \times p)}$ *defined*

in Theorem 4.8.2 *exists. Next, consider a linear combination of the form* $\mathbf{c}'_{(1 \times p)} \cdot \mathbf{Z}_{(p \times 1)}$

with jth standardized multivariate sample observation $\mathbf{c}'_{(1 \times p)} \cdot \mathbf{Z}_{j}$ given in

Definition 4.9.1. Then the unrealized sample variance for a linear combination of $\sum_{(n \times p)} is$ defined by

sample variance of
$$\underset{(1\times p)}{c'}\cdot\underset{(p\times 1)}{Z}$$

$$= \operatorname{var}\left(\mathbf{c}'_{(1\times p)} \cdot \mathbf{Z}_{(p\times 1)} \right) = \mathbf{c}'_{(1\times p)} \cdot \mathbf{S}_{\mathbf{Z}}_{(p\times p)} \cdot \mathbf{c}_{(p\times 1)} = \mathbf{c}'_{(1\times p)} \cdot \mathbf{R}_{(p\times p)} \cdot \mathbf{c}_{(p\times 1)}.$$

Definition 4.9.4 (Sample Covariance for Two Linear Combinations of Z). Let

 \mathbf{Z}_1 , \mathbf{Z}_2 , ..., \mathbf{Z}_n constitute a standardized multivariate random sample defined in $(p \times 1)$ $(p \times 1)$

Theorem 4.6.1. *Assume the sample variance-covariance matrix* $\mathbf{S}_{\mathbf{Z}} = \mathbf{R}_{(p \times p)}$ *defined*

in Theorem 4.8.2 exists. Next, consider two linear combinations of the form

 $\begin{array}{l} \mathbf{b}' \\ {}_{(1 \times p)} \cdot \mathbf{Z}_{p \times 1} \end{array} and \begin{array}{l} \mathbf{c}' \\ {}_{(1 \times p)} \cdot \mathbf{Z}_{p \times 1} \end{array} with jth \text{ standardized multivariate sample observations} \\ \\ \mathbf{b}' \\ {}_{(1 \times p)} \cdot \mathbf{Z}_{j} and \begin{array}{l} \mathbf{c}' \\ {}_{(1 \times p)} \cdot \mathbf{Z}_{j} \end{array}, respectively, given in \end{array}$

Definition 4.9.1. Then the unrealized sample covariance for two linear combinations

of $\underset{(n \times p)}{\mathbf{Z}}$ is defined by

sample covariance of
$$\mathbf{b}'_{(1\times p)} \cdot \mathbf{Z}_{(p\times 1)}$$
 and $\mathbf{c}'_{(1\times p)} \cdot \mathbf{Z}_{(p\times 1)}$

$$\operatorname{cov}\left(\mathbf{b}'_{(1\times p)} \cdot \mathbf{Z}_{(p\times 1)}, \mathbf{c}'_{(1\times p)} \cdot \mathbf{Z}_{(p\times 1)}\right)$$

$$= \mathbf{b}'_{(1\times p)} \cdot \mathbf{S}_{\mathbf{Z}} \cdot \mathbf{c}_{(p\times 1)} = \mathbf{c}'_{(1\times p)} \cdot \mathbf{S}_{\mathbf{Z}} \cdot \mathbf{b}_{(p\times 1)}$$

$$= \mathbf{b}'_{(1\times p)} \cdot \mathbf{R}_{(p\times p)} \cdot \mathbf{c}_{(p\times 1)} = \mathbf{c}'_{(1\times p)} \cdot \mathbf{R}_{(p\times p)} \cdot \mathbf{b}_{(p\times 1)}.$$

4.9.3 q Linear Combinations of Standardized Samples

Definition 4.9.5 (*q* Linear Combinations of **Z**). Let \mathbf{Z}_1 , \mathbf{Z}_2 , ..., \mathbf{Z}_n constitute a $(p \times 1)$ $(p \times 1)$ $(p \times 1)$

standardized multivariate random sample defined in Theorem 4.6.1. Now consider q linear combinations of $\underset{(n \times p)}{\mathbf{Z}}$ of the form:

$$Y_{i} = \frac{\mathbf{c}'_{i}}{(1 \times p)} \cdot \frac{\mathbf{Z}}{(p \times 1)} = \begin{bmatrix} c_{i1}, c_{i2}, \dots, c_{ip} \end{bmatrix} \cdot \begin{bmatrix} Z_{1} \\ Z_{2} \\ \vdots \\ Z_{p} \end{bmatrix} = c_{i1}Z_{1} + c_{i2}Z_{2} + \dots + c_{ip}Z_{p}$$

for i = 1,2, ..., *q linear combinations*

$$Y_{1} = \mathbf{c}_{1}' \cdot \mathbf{Z}_{(p \times 1)} = c_{11}Z_{1} + c_{12}Z_{2} + \dots + c_{1p}Z_{p}$$
$$Y_{2} = \mathbf{c}_{2}' \cdot \mathbf{Z}_{(p \times 1)} = c_{21}Z_{1} + c_{22}Z_{2} + \dots + c_{2p}Z_{p}$$
$$\vdots \qquad \vdots$$
$$Y_{q} = \mathbf{c}_{q}' \cdot \mathbf{Z}_{(p \times 1)} = c_{q1}Z_{1} + c_{q2}Z_{2} + \dots + c_{qp}Z_{p}$$

or in matrix notation,

$$\mathbf{Y}_{(q\times1)} = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_q \end{bmatrix} = \begin{bmatrix} \mathbf{c}_1' & \mathbf{Z} \\ (1\times p) & (p\times1) \\ \mathbf{c}_2' & \mathbf{Z} \\ (1\times p) & (p\times1) \\ \vdots \\ \mathbf{c}_q' & \mathbf{Z} \\ (1\times p) & (p\times1) \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1p} \\ c_{21} & c_{22} & \cdots & c_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ c_{q1} & c_{q2} & \cdots & c_{qp} \end{bmatrix} \begin{bmatrix} Z_1 \\ Z_2 \\ \vdots \\ Z_p \\ (p\times1) \end{bmatrix} = \mathbf{C} \cdot \mathbf{Z}$$

where the unrealized quantity on the jth standardized multivariate sample observation, j = 1, 2, ..., n, and ith linear combination, i = 1, 2, ..., q, is

$$Y_{ji} = \mathbf{c}'_{i} \cdot \mathbf{Z}_{j} = \begin{bmatrix} c_{i1}, c_{i2}, \dots, c_{ip} \end{bmatrix} \cdot \begin{bmatrix} Z_{j1} \\ Z_{j2} \\ \vdots \\ Z_{jp} \end{bmatrix} = c_{i1}Z_{j1} + c_{i2}Z_{j2} + \dots + c_{ip}Z_{jp}.$$

4.9.4 Sample Mean Vector for q Linear Combinations of

Standardized Samples

Definition 4.9.6 (Sample Mean Vector for *q* Linear Combinations of **Z**). Let

 \mathbf{Z}_1 , \mathbf{Z}_2 , ... , \mathbf{Z}_n constitute a standardized multivariate random sample defined in $_{(p\times 1)}$

Theorem 4.6.1. *Assume the sample mean vector* $\bar{\mathbf{Z}}_{(p \times 1)} = \mathbf{0}_{(p \times 1)}$ *defined*
in Theorem 4.8.1 exists. Next, consider q linear combinations of $\mathbf{Z}_{(n \times p)}$ of the form

 $Y_i = \mathbf{c}'_i \cdot \mathbf{Z}_{(1 \times p)}$ with *j*th standardized multivariate sample observation

 $Y_{ji} = \mathbf{c}'_i \cdot \mathbf{Z}_j$ given in Definition 4.9.5. Then the unrealized sample mean vector

for *q* linear combinations of $\underset{(n \times p)}{\mathbf{Z}}$ is defined by

sample mean vector of $\mathbf{Y}_{(q \times 1)}$

$$\overline{\mathbf{Y}}_{(q\times 1)} = E\left(\mathbf{Y}_{(q\times 1)}\right) = E\left(\mathbf{C}_{(q\times p)} \cdot \mathbf{Z}_{(p\times 1)}\right) = \mathbf{C}_{(q\times p)} \cdot \overline{\mathbf{Z}}_{(p\times 1)} = \mathbf{0}_{(q\times 1)}.$$

Thus, the *i*th row of $\mathbf{Y}_{(q \times 1)}$ has unrealized sample mean

$$\bar{Y}_i = E(Y_i) = E\left(\mathbf{c}'_i \cdot \mathbf{Z}_{(1 \times p)}\right) = \mathbf{c}'_i \cdot \mathbf{\bar{Z}}_{(1 \times p)} = \mathbf{0}$$

for i = 1, 2, ..., q.

4.9.5 Sample Variance-Covariance Matrix for *q* Linear

Combinations of Standardized Samples

Definition 4.9.7 (Sample Variance-Covariance Matrix for q Linear Combinations of

Z). Let \mathbf{Z}_1 , \mathbf{Z}_2 , ..., \mathbf{Z}_n constitute a standardized multivariate random sample $(p \times 1)$ $(p \times 1)$ $(p \times 1)$

defined in Theorem 4.6.1. Assume the sample variance-covariance matrix

 $\mathbf{S}_{\mathbf{Z}} = \mathbf{R}_{(p \times p)}$ defined in Theorem 4.8.2 exists. Next, consider q linear combinations of

 $\sum_{(n \times p)} of the form Y_i = \mathbf{c}'_i \cdot \mathbf{Z}_{(n \times p)} with jth standardized multivariate sample$

observation $Y_{ji} = \mathbf{c}'_i \cdot \mathbf{Z}_j$ given in Definition 4.9.5. Then the unrealized

symmetric standardized sample variance-covariance matrix for q linear

combinations of $\underset{(n \times p)}{\textbf{Z}}$ is defined by

$$\begin{split} \mathbf{S}_{\mathbf{Y}} &= \mathbf{C}_{(q \times p)} \cdot \mathbf{S}_{\mathbf{Z}} \cdot \mathbf{C}'_{(p \times p)} \\ &= \begin{bmatrix} \mathbf{C}'_1 \cdot \mathbf{S}_{\mathbf{Z}} \cdot \mathbf{C}_1 & \mathbf{C}'_1 \cdot \mathbf{S}_{\mathbf{Z}} \cdot \mathbf{C}_2 & \cdots & \mathbf{C}'_1 \cdot \mathbf{S}_{\mathbf{Z}} \cdot \mathbf{C}_q \\ (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) \\ \mathbf{C}'_2 \cdot \mathbf{S}_{\mathbf{Z}} \cdot \mathbf{C}_1 & \mathbf{C}'_2 \cdot \mathbf{S}_{\mathbf{Z}} \cdot \mathbf{C}_2 & \cdots & \mathbf{C}'_2 \cdot \mathbf{S}_{\mathbf{Z}} \cdot \mathbf{C}_q \\ (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{C}'_q \cdot \mathbf{S}_{\mathbf{Z}} \cdot \mathbf{C}_1 & \mathbf{C}'_q \cdot \mathbf{S}_{\mathbf{Z}} \cdot \mathbf{C}_2 & \cdots & \mathbf{C}'_q \cdot \mathbf{S}_{\mathbf{Z}} \cdot \mathbf{C}_q \\ (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) \\ \end{bmatrix} \end{split}$$

$$= \begin{bmatrix} \mathbf{C}'_1 \cdot \mathbf{R} \cdot \mathbf{C}_1 & \mathbf{C}'_1 \cdot \mathbf{R} \cdot \mathbf{C}_2 & \cdots & \mathbf{C}'_1 \cdot \mathbf{R} \cdot \mathbf{C}_q \\ (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) \\ \mathbf{C}'_2 \cdot \mathbf{R} \cdot \mathbf{C}_1 & \mathbf{C}'_2 \cdot \mathbf{R} \cdot \mathbf{C}_2 & \cdots & \mathbf{C}'_2 \cdot \mathbf{R} \cdot \mathbf{C}_q \\ (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{C}'_q \cdot \mathbf{R} \cdot \mathbf{C}_1 & \mathbf{C}'_q \cdot \mathbf{R} \cdot \mathbf{C}_2 & \cdots & \mathbf{C}'_q \cdot \mathbf{R} \cdot \mathbf{C}_q \\ (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{C}'_q \cdot \mathbf{R} \cdot \mathbf{C}_1 & \mathbf{C}'_q \cdot \mathbf{R} \cdot \mathbf{C}_2 & \cdots & \mathbf{C}'_q \cdot \mathbf{R} \cdot \mathbf{C}_q \\ (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{C}'_q \cdot \mathbf{R} \cdot \mathbf{C}_1 & \mathbf{C}'_q \cdot \mathbf{R} \cdot \mathbf{C}_2 & \cdots & \mathbf{C}'_q \cdot \mathbf{R} \cdot \mathbf{C}_q \\ (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times 1) & (1 \times p) & (p \times p) & (p \times p) & (p \times 1) \end{bmatrix}$$

Thus, the *i*th row of $\underset{(q \times 1)}{\mathbf{Y}}$ has unrealized sample variance

$$\operatorname{var}(Y_i) = \underset{(1 \times p)}{\mathbf{c}'_i} \cdot \underset{(p \times p)}{\mathbf{S}_{\mathbf{Z}}} \cdot \underset{(p \times 1)}{\mathbf{c}_i} = \underset{(1 \times p)}{\mathbf{c}'_i} \cdot \underset{(p \times p)}{\mathbf{R}} \cdot \underset{(p \times 1)}{\mathbf{c}_i}$$

for i = 1, 2, ..., q.

And, the *i*th row and *k*th row of $\mathbf{Y}_{(q \times 1)}$ has unrealized sample covariance

 $\operatorname{cov}(Y_i, Y_k)$

$$= \mathbf{c}'_{i} \cdot \mathbf{S}_{\mathbf{Z}} \cdot \mathbf{c}_{k} = \mathbf{c}'_{k} \cdot \mathbf{S}_{\mathbf{Z}} \cdot \mathbf{c}_{i}$$
$$= \mathbf{c}'_{i} \cdot \mathbf{S}_{\mathbf{Z}} \cdot \mathbf{c}_{i}$$
$$(p \times p) \cdot (p \times p) \cdot \mathbf{c}_{k} = \mathbf{c}'_{k} \cdot \mathbf{R} \cdot \mathbf{c}_{i}$$
$$= \mathbf{c}'_{i} \cdot \mathbf{R} \cdot \mathbf{c}_{k} = \mathbf{c}'_{k} \cdot \mathbf{R} \cdot \mathbf{c}_{i}$$
$$(p \times p) \cdot \mathbf{c}_{i}$$
$$(p \times p) \cdot \mathbf{c}_{i}$$

for i, k = 1, 2, ..., q.

Chapter 5

Principal Components Analysis

5.1 Introduction

A principal components analysis is concerned with explaining the variancecovariance (or correlation) structure of a set of variables through a few linear combinations of these variables. Its general objectives are (1) data reduction and (2) interpretation [3, p. 430].

5.2 Population Principal Components

Algebraically, population principal components are particular linear combinations of the p population continuous random variables $X_1, X_2, ..., X_p$. Geometrically, these linear combinations represent the selection of a new coordinate system obtained by rotating the original system with $X_1, X_2, ..., X_p$ as the coordinate axes. The new axes represent the directions with maximum variability and provide a simpler and more parsimonious description of the covariance (or correlation) structure.

As we shall see, principal components depend solely on the covariance matrix $\sum_{(p \times p)} or$ the correlation matrix $\sum_{(p \times p)} = \rho_{(p \times p)}$. Their development does *not* require a multivariate normal assumption. On the other hand, principal components derived for multivariate normal populations have useful interpretations in terms of the constant density ellipsoids. Further, inferences can be made from the sample components when the population is multivariate normal [3, pp. 430-431].

Let

$$\mathbf{X}_{(p\times 1)} = \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_p \end{bmatrix}_{(p\times 1)}$$

be a population random vector for continuous random variables defined in Definition 3.2.1. Assume the corresponding population variance-covariance matrix

$$\sum_{(p \times p)} \begin{bmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1p} \\ \sigma_{21} & \sigma_{22} & \cdots & \sigma_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{p1} & \sigma_{p2} & \cdots & \sigma_{pp} \end{bmatrix}$$

defined in Theorem 3.2.1 is positive definite with eigenvalue and normalizedeigenvector pairs

$$\begin{pmatrix} \lambda_1, \mathbf{e}_1 \\ (p \times 1) \end{pmatrix}, \begin{pmatrix} \lambda_2, \mathbf{e}_2 \\ (p \times 1) \end{pmatrix}, \dots, \begin{pmatrix} \lambda_i, \mathbf{e}_i \\ (p \times 1) \end{pmatrix}, \dots, \begin{pmatrix} \lambda_p, \mathbf{e}_p \\ (p \times 1) \end{pmatrix}$$

where $\lambda_1 > \lambda_2 > \cdots > \lambda_p > 0$. That is, the λ_i 's are positive and distinct. One should be aware that a population variance-covariance matrix is in general positive semidefinite [6, p. 200]. But some books still assume positive definite population variance-covariance matrix in their treatment of PCA [6, p. 206]. The purpose of the assumption here is due to the fact that the proof for Theorem 5.2.1 (*i*th Population Principal Component) uses Theorem 2.2.1 (Maximization of Quadratic Forms for Points on the Unit Sphere) where $\mathbf{B}_{(p \times p)} = \sum_{(p \times p)} \mathbf{i}$ is positive definite. To clarify, we assume $\lambda_i > 0, i = 1, ..., p$ based on $\sum_{(p \times p)} \mathbf{b}$ being positive definite. However, the assumption of the λ_i 's being distinct ensures the \mathbf{e}_i 's are mutually orthogonal. In general, the λ_i 's can be repeated but then the associated eigenvectors need to be chosen to be orthogonal [3, p. 432].

Therefore, for the remainder of the paper we will assume all populations variance-covariance and correlation matrices will be positive definite and the λ_i 's are positive and distinct, including for the sample cases. It is our belief that these assumptions do not detract from the general concept of principal components analysis and are also seen quite often in applications. Our rationalization comes from the fact that, the variance-covariance matrix of a multivariate probability distribution is positive definite unless one variable is an *exact* linear function of the others [7].

Moving on, let the orthogonal matrix with columns being the normalized eigenvectors be

$$\mathbf{E}_{(p \times p)} = \begin{bmatrix} e_{11} & e_{12} & \cdots & e_{1p} \\ e_{21} & e_{22} & \cdots & e_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ e_{p1} & e_{p2} & \cdots & e_{pp} \end{bmatrix} = \begin{bmatrix} \mathbf{e}_1 & \vdots & \mathbf{e}_2 & \vdots & \cdots & \vdots & \mathbf{e}_p \\ (p \times 1) & (p \times p) & (p \times p) \end{bmatrix}$$

given in Definition 2.2.18. From Definition 3.3.2 consider q = p linear combinations, Y_i , of the p population continuous random variables $X_1, X_2, ..., X_p$ with arbitrary coefficients:

$$Y_{1} = \frac{\mathbf{c}_{1}'}{(1 \times p)} \cdot \frac{\mathbf{X}}{(p \times 1)} = c_{11}X_{1} + c_{12}X_{2} + \dots + c_{1p}X_{p}$$
$$Y_{2} = \frac{\mathbf{c}_{2}'}{(1 \times p)} \cdot \frac{\mathbf{X}}{(p \times 1)} = c_{21}X_{1} + c_{22}X_{2} + \dots + c_{2p}X_{p}$$
$$\vdots \qquad \vdots$$
$$Y_{p} = \frac{\mathbf{c}_{p}'}{(1 \times p)} \cdot \frac{\mathbf{X}}{(p \times 1)} = c_{p1}X_{1} + c_{p2}X_{2} + \dots + c_{pp}X_{p}$$

or in matrix notation,

$$\mathbf{Y}_{(p\times1)} = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_p \\ (p\times1) \end{bmatrix} = \begin{bmatrix} \mathbf{c}'_1 & \mathbf{X} \\ (1\times p) & (p\times1) \\ \mathbf{c}'_2 & \mathbf{X} \\ (1\times p) & (p\times1) \\ \vdots \\ \mathbf{c}'_p & \mathbf{X} \\ (1\times p) & (p\times1) \end{bmatrix} = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1p} \\ c_{21} & c_{22} & \cdots & c_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ c_{p1} & c_{p2} & \cdots & c_{pp} \\ (p\times p) & (p\times1) \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_p \end{bmatrix} = \mathbf{C} & \mathbf{X} \\ (p\times p) & (p\times1) \end{bmatrix}$$

Using Theorem 3.3.5, we obtain

$$\operatorname{Var}(Y_i) = \operatorname{Var}\left(\mathbf{c}'_i \cdot \mathbf{X}_{(1 \times p)} \right) = \mathbf{c}'_i \cdot \sum_{(1 \times p)} \mathbf{c}'_{(p \times p)} \cdot \mathbf{c}_i_{(p \times 1)}$$

for i = 1, 2, ..., p and

$$\operatorname{Cov}(Y_i, Y_k) = \operatorname{Cov}\left(\mathbf{c}'_i \cdot \mathbf{X}_{(1 \times p)}, \mathbf{c}'_{(1 \times p)} \cdot \mathbf{X}_{(p \times 1)}\right) = \mathbf{c}'_i \cdot \sum_{(1 \times p)} \mathbf{c}'_{(p \times p)} \cdot \mathbf{c}'_{(p \times 1)}$$

for $i, k = 1, 2, ..., p, i \neq k$.

The population principal components are those uncorrelated linear combinations

 Y_1, Y_2, \dots, Y_p whose population variances are as large as possible [3, p. 431].

The first population principal component is the linear combination with

 $\operatorname{Var}(Y_1) = \operatorname{Var}\left(\mathbf{c}'_1 \cdot \mathbf{X}_{(1 \times p)} \right) = \mathbf{c}'_1 \cdot \sum_{(1 \times p)} \mathbf{c}'_{(p \times p)} \cdot \mathbf{c}_1_{(p \times 1)}.$ It is clear that $\operatorname{Var}(Y_1)$ can be

increased by multiplying any $\mathop{\mathbf{c_1}}_{(p\times 1)}$ by some constant. To eliminate this

maximum variance among all linear combinations. That is, it maximizes

indeterminacy, it is convenient to restrict attention to coefficient vectors of unit length. We therefore define

First population principal component = linear combination $Y_1 = \mathbf{c}'_1 \cdot \mathbf{X}_{(1 \times p)}$ that maximizes

$$\operatorname{Var}\left(\underset{(1\times p)}{\mathbf{c}_{1}'}\cdot\underset{(p\times 1)}{\mathbf{X}}\right)\operatorname{subject}\operatorname{to}\underset{(1\times p)}{\mathbf{c}_{1}'}\cdot\underset{(p\times 1)}{\mathbf{c}_{1}}=1$$

Second population principal component = linear combination $Y_2 = \mathbf{c}'_2 \cdot \mathbf{X}_{(1 \times p)}$ that maximizes $\operatorname{Var}\left(\mathbf{c}'_2 \cdot \mathbf{X}_{(1 \times p)}\right)$ subject to $\mathbf{c}'_2 \cdot \mathbf{c}_2_{(1 \times p)} = 1$ and

$$\operatorname{Cov}\left(\mathbf{c}_{1}' \cdot \mathbf{X}_{(p \times 1)}, \mathbf{c}_{2}' \cdot \mathbf{X}_{(p \times 1)}\right) = 0$$

And the *i*th step,

*i*th population principal component = linear combination $Y_i = \mathbf{c}'_i \cdot \mathbf{X}_{(1 \times p)}$ that maximizes

$$\operatorname{Var}\left(\begin{array}{c} \mathbf{c}'_{i} \cdot \mathbf{X}\\ (1 \times p) \end{array}\right) \text{ subject to } \begin{array}{c} \mathbf{c}'_{i} \cdot \mathbf{c}_{i} \\ (1 \times p) \end{array} = 1 \text{ and}$$
$$\operatorname{Cov}\left(\begin{array}{c} \mathbf{c}'_{i} \cdot \mathbf{X}\\ (1 \times p) \end{array}\right) \left(\begin{array}{c} \mathbf{c}'_{k} \\ (p \times 1) \end{array}\right) \left(\begin{array}{c} \mathbf{c}'_{k} \\ (p \times 1) \end{array}\right) \left(\begin{array}{c} \mathbf{c}'_{k} \\ (p \times 1) \end{array}\right) = 0 \quad \text{for} \quad k < i$$

[3, p. 431].

Theorem 5.2.1 (*i*th Population Principal Component). Let $\underset{(p \times 1)}{\mathbf{X}}$ be a population random vector for continuous random variables defined in Definition 3.2.1 *with associated positive-definite variance-covariance matrix* $\underset{(p \times p)}{\sum}$ defined in Theorem

3.2.1. Let $\sum_{(p \times p)}$ have eigenvalue and normalized-eigenvector pairs $\left(\lambda_{i}, \mathbf{e}_{i}\right)$

i = 1, 2, ..., p where $\lambda_1 > \lambda_2 > \cdots > \lambda_p > 0$. Then the unrealized **i**th population principal component *is given by*

$$Y_i = \underbrace{\mathbf{e}'_i}_{(1 \times p)} \cdot \underbrace{\mathbf{X}}_{(p \times 1)} = \begin{bmatrix} e_{1i}, e_{2i}, \dots, e_{pi} \end{bmatrix} \cdot \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_p \end{bmatrix} = e_{1i}X_1 + e_{2i}X_2 + \dots + e_{pi}X_p$$

for *i* = 1,2, ..., *p*, with unrealized population variance and covariance

$$\operatorname{Var}(Y_i) = \frac{\mathbf{e}'_i}{(1 \times p)} \cdot \sum_{(p \times p)} \cdot \frac{\mathbf{e}_i}{(p \times 1)} = \lambda_i$$

for i = 1,2, ..., *p and*

$$\operatorname{Cov}(Y_i, Y_k) = \frac{\mathbf{e}'_i}{(1 \times p)} \cdot \sum_{(p \times p)} \cdot \frac{\mathbf{e}_k}{(p \times 1)} = 0$$

for $i, k = 1, 2, ..., p, i \neq k$ [3, p. 432].

Proof.

Using Definition 2.2.18, Theorem 3.3.5

$$\operatorname{Var}(Y_i) = \frac{\mathbf{e}'_i}{(1 \times p)} \cdot \sum_{\substack{(p \times p)}} \cdot \frac{\mathbf{e}_i}{(p \times 1)}$$
$$= \frac{\mathbf{e}'_i}{(1 \times p)} \cdot \left(\sum_{\substack{(p \times p)}} \cdot \frac{\mathbf{e}_i}{(p \times 1)} \right)$$
$$= \frac{\mathbf{e}'_i}{(1 \times p)} \cdot \left(\lambda_i \cdot \mathbf{e}_i \right)$$
$$= \lambda_i \cdot \frac{\mathbf{e}'_i}{(1 \times p)} \cdot \frac{\mathbf{e}_i}{(p \times 1)}$$
$$= \lambda_i \cdot 1 = \lambda_i, i = 1, 2, \dots, p$$

Similarly,

$$Cov(Y_i, Y_k) = \frac{\mathbf{e}'_i}{(1 \times p)} \cdot \sum_{\substack{(p \times p)}} \cdot \frac{\mathbf{e}_k}{(p \times p)}$$
$$= \frac{\mathbf{e}'_i}{(1 \times p)} \cdot \left(\sum_{\substack{(p \times p)}} \cdot \frac{\mathbf{e}_k}{(p \times 1)} \right)$$
$$= \frac{\mathbf{e}'_i}{(1 \times p)} \cdot \left(\lambda_k \cdot \mathbf{e}_k \right)$$
$$= \lambda_k \cdot \frac{\mathbf{e}'_i}{(1 \times p)} \cdot \frac{\mathbf{e}_k}{(p \times 1)}$$
$$= \lambda_k \cdot 0 = 0, i, k = 1, 2, \dots, p, i \neq k.$$

Next, we know from the first part of Theorem 2.2.1, with $\mathbf{B}_{(p \times p)} = \sum_{(p \times p)} \mathbf{K}_{\mathbf{X}}$, that

$$\max_{\substack{\mathbf{c} \neq \mathbf{0} \\ (p \times 1) \neq (p \times 1)}} \frac{\mathbf{c}' \cdot \sum_{\substack{(p \times p) \\ (1 \times p) \neq (p \times 1)}} \mathbf{c}}{\mathbf{c}' \cdot \mathbf{c}} = \lambda_1 \qquad \left(\text{attained when } \mathbf{c} = \mathbf{e}_1 \right)$$

By Definition 2.2.18 $\mathbf{e}'_1 \cdot \mathbf{e}_1 = 1$ since the eigenvectors are normalized. Thus, $(1 \times p) \cdot (p \times 1) = 1$

$$\max_{\substack{\mathbf{c} \\ (p\times1) \neq (p\times1)}} \frac{\mathbf{c}' \cdot \sum_{(1\times p)} \cdot \mathbf{c}_{(p\times1)}}{\mathbf{c}' \cdot \mathbf{c}_{(1\times p)} \cdot (p\times1)} = \lambda_1 = \frac{\mathbf{e}'_1 \cdot \sum_{\mathbf{X}} \cdot \mathbf{e}_1}{(1\times p) \cdot (p\times p) \cdot (p\times1)} = \mathbf{e}'_1 \cdot \sum_{(1\times p)} \mathbf{e}'_1 \cdot \mathbf{e}_1$$

Similarly, using the second part of Theorem 2.2.1 we get

$$\max_{\substack{\mathbf{c} \\ (p \times 1)^{\perp} (p \times 1) (p \times 1)}} \max_{\substack{\mathbf{e}_2, \dots, \mathbf{e}_k \\ (p \times 1)^{\perp} (p \times 1)}} \frac{\mathbf{c}' \cdot \sum_{\substack{(1 \times p)}} \mathbf{c}_{\substack{(p \times p)}} \cdot \mathbf{c}_{\substack{(p \times 1)}}}{\mathbf{c}' \cdot \mathbf{c}_{\substack{(1 \times p)}}} = \lambda_{k+1}, \qquad k = 1, 2, \dots, p-1$$

For the choice $\mathbf{c}_{(p \times 1)} = \mathbf{e}_{k+1}$, with $\mathbf{e}'_{k+1} \cdot \mathbf{e}_i_{(p \times 1)} = 0$,

for i = 1, 2, ..., k and k = 1, 2, ..., p - 1,

$$\frac{\mathbf{e}'_{k+1} \cdot \sum_{\substack{\mathbf{X} \\ (1 \times p) \\ (1 \times p) \\ (1 \times p) \\ (1 \times p) \\ (p \times 1)}}{\mathbf{e}'_{k+1} \cdot \mathbf{e}_{k+1} \\ (1 \times p) \\ (p \times 1)}} = \mathbf{e}'_{k+1} \cdot \sum_{\substack{\mathbf{X} \\ (1 \times p) \\ (p \times p) \\ (p \times 1)}} \cdot \mathbf{e}_{k+1} \\ (p \times 1)} = \operatorname{Var}(Y_{k+1}) \blacksquare$$

From above, the principal components are uncorrelated and have variances equal to the eigenvalues of $\sum_{(p \times p)} [3, p. 432]$.

Thus, the population principal components, Y_i , are given by

$$Y_{1} = \mathbf{e}_{1}' \cdot \mathbf{X}_{(p \times 1)} = e_{11}X_{1} + e_{21}X_{2} + \dots + e_{p1}X_{p}$$
$$Y_{2} = \mathbf{e}_{2}' \cdot \mathbf{X}_{(p \times 1)} = e_{12}X_{1} + e_{22}X_{2} + \dots + e_{p2}X_{p}$$
$$\vdots \qquad \vdots$$
$$Y_{p} = \mathbf{e}_{p}' \cdot \mathbf{X}_{(p \times 1)} = e_{1p}X_{1} + e_{2p}X_{2} + \dots + e_{pp}X_{p}$$

or in matrix notation,

$$\mathbf{Y}_{(p \times 1)} = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_p \\ (p \times 1) \end{bmatrix} = \begin{bmatrix} \mathbf{e}'_1 & \cdot \mathbf{X} \\ \stackrel{(1 \times p) \quad (p \times 1)}{\mathbf{e}'_2 & \cdot \mathbf{X}} \\ \stackrel{(1 \times p) \quad (p \times 1)}{\vdots} \\ \mathbf{e}'_p & \cdot \mathbf{X} \\ \stackrel{(1 \times p) \quad (p \times 1)}{(p \times 1)} \end{bmatrix} = \begin{bmatrix} e_{11} & e_{21} & \cdots & e_{p1} \\ e_{12} & e_{22} & \cdots & e_{p2} \\ \vdots & \vdots & \ddots & \vdots \\ e_{1p} & e_{2p} & \cdots & e_{pp} \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_p \end{bmatrix} = \mathbf{E}' \cdot \mathbf{X} \\ \stackrel{(p \times p) \quad (p \times 1)}{(p \times 1)}$$

Theorem 5.2.2 (Total Population Variance). Let $\underset{(p \times 1)}{X}$ be a population random vector

for continuous random variables defined in Definition 3.2.1 with associated

positive-definite variance-covariance matrix $\sum_{(p \times p)} defined$ *in Theorem 3.2.1. Let*

$$\sum_{(p \times p)} have \ eigenvalue \ and \ normalized - eigenvector \ pairs\left(\lambda_i, \mathbf{e}_i\right) i = 1, 2, \dots, p$$

where $\lambda_1 > \lambda_2 > \dots > \lambda_p > 0$. Let $Y_1 = \mathbf{e}'_1 \cdot \mathbf{X}_{(p \times 1)}, Y_2 = \mathbf{e}'_2 \cdot \mathbf{X}_{(p \times 1)}, \dots, Y_p = \mathbf{e}'_p \cdot \mathbf{Y}_{(1 \times p)}$.

 $\underset{(p\times 1)}{\mathbf{X}}$ be the population principal components. Then the total population variance

$$\sigma_{11} + \sigma_{22} + \dots + \sigma_{pp} = \sum_{i=1}^{p} \sigma_{ii} = \lambda_1 + \lambda_2 + \dots + \lambda_p = \sum_{i=1}^{p} \operatorname{Var}(Y_i).$$

Proof.

From Definition 2.2.14,

$$\operatorname{tr}(\Sigma_{\mathbf{X}}) = \sum_{i=1}^{p} \sigma_{ii} = \sigma_{11} + \sigma_{22} + \dots + \sigma_{pp}.$$

Using a direct result of Result 2.2.8 with $\mathbf{A}_{(p \times p)} = \sum_{(p \times p)} \mathbf{X}_{p}$, we can write

$$\sum_{(p \times p)} = \underbrace{\mathbf{E}}_{(p \times p)} \cdot \underbrace{\mathbf{\Lambda}}_{(p \times p)} \cdot \underbrace{\mathbf{E}'}_{(p \times p)}$$

where $\bigwedge_{(p \times p)}$ is the diagonal matrix of eigenvalues and $\underset{(p \times p)}{\mathbf{E}}$ is the orthogonal matrix

with columns being the normalized eigenvectors.

Using Result 2.2.6 (b) and orthogonality of $\mathop{\mathbf{E}}_{(p\times p)}$, we have

$$\operatorname{tr}(\underline{\Sigma}_{\mathbf{X}}) = \operatorname{tr}(\mathbf{E} \cdot \mathbf{\Lambda} \cdot \mathbf{E}') = \operatorname{tr}(\mathbf{\Lambda} \cdot \mathbf{E}' \cdot \mathbf{E}) = \operatorname{tr}(\mathbf{\Lambda} \cdot \mathbf{I}) = \operatorname{tr}(\mathbf{\Lambda}) = \lambda_1 + \lambda_2 + \dots + \lambda_p$$

Thus,

$$\sum_{i=1}^{p} \sigma_{ii} = \operatorname{tr}(\Sigma_{\mathbf{X}}) = \operatorname{tr}(\mathbf{\Lambda}) = \sum_{i=1}^{p} \operatorname{Var}(Y_{i}) \blacksquare$$

Hence,

Total population variance =
$$\sigma_{11} + \sigma_{22} + \dots + \sigma_{pp}$$

$$=\lambda_1+\lambda_2+\cdots+\lambda_p.$$

Consequently,

$$\begin{pmatrix} \text{Proportion of total} \\ \text{population variance} \\ \text{due to } i \text{th population} \\ \text{principal component} \end{pmatrix} = \frac{\lambda_i}{\lambda_1 + \lambda_2 + \dots + \lambda_p} \qquad i = 1, 2, \dots, p$$

$$\begin{pmatrix} \text{Proportion of total population} \\ \text{variance due to the first } k \text{ population} \\ \text{principal components} \end{pmatrix} = \frac{\sum_{i=1}^{k} \lambda_i}{\lambda_1 + \lambda_2 + \dots + \lambda_p} \qquad k < p.$$

If most (for instance, 80 to 90%) of the total population variance, for large p, can be attributed to the first one, two, or three components, then these components can "replace" the original p variables without much loss of information.

Each component of the coefficient vector $\mathbf{e}'_i = [e_{1i}, e_{2i}, \dots, e_{ki}, \dots, e_{pi}]$ also merits inspection. The magnitude of e_{ki} measures the importance of the *k*th variable to the *i*th principal component, irrespective of the other variables [3, pp. 432-433].

and

5.3 Population Principal Components for Standardized Continuous Random Variables

The population principal components derived from a standardized population random vector for continuous random variables $\mathbf{Z}_{(p \times 1)}$ may be obtained from the normalized eigenvectors of the correlation matrix $\sum_{(p \times p)} \sum_{(p \times p)} \rho$. All our previous results apply, with some simplifications, since the variance of each Z_i is unity. We shall continue to use the notation Y_i to refer to the *i*th population principal component and $\left(\lambda_{i}, \mathbf{e}_{i} \atop (n \times 1)\right)$ for the eigenvalue and normalized-eigenvector pair from either $\sum_{(p \times p)} = \rho \operatorname{or} \sum_{(p \times p)} \sum_{(p \times p)}$ general, not the same as the ones derived from $\sum_{(p \times p)} \sum_{(p \times p)} \sum_{(p \times p)} [3, p. 437].$ **Theorem 5.3.1** (*i*th Population Principal Component of **Z**). Let $\underset{(p \times 1)}{\mathbf{Z}}$ be a standardized population random vector for continuous random variables defined in Definition 3.4.1. with associated positive-definite variance-covariance matrix $\sum_{(p \times p)} = \rho \text{ defined in Theorem 3.4.4. Let } \sum_{(p \times p)} = \rho \text{ have eigenvalue and }$ *normalized-eigenvector pairs* $\left(\lambda_{i}, \underbrace{\mathbf{e}_{i}}_{(p \times 1)}\right), i = 1, 2, ..., p \text{ where } \lambda_{1} > \lambda_{2} > \cdots > \lambda_{p} > 0.$ Then the unrealized **i**th population principal component of $\sum_{(p \times 1)} is given by$

$$Y_{i} = \mathbf{e}_{i}' \cdot \mathbf{Z}_{(p \times 1)} = \mathbf{e}_{i}' \cdot \left(\mathbf{V}_{(p \times p)}^{1/2}\right)^{-1} \cdot \left(\mathbf{X}_{(p \times 1)} - \mathbf{\mu}_{\mathbf{X}}\right)$$
$$= \mathbf{e}_{i}' \cdot \mathbf{V}_{(p \times p)}^{-1/2} \cdot \left(\mathbf{X}_{(p \times 1)} - \mathbf{\mu}_{\mathbf{X}}\right)$$
$$= \left[e_{1i}, e_{2i}, \dots, e_{pi}\right] \cdot \begin{bmatrix} Z_{1} \\ Z_{2} \\ \vdots \\ Z_{p} \end{bmatrix} = e_{1i}Z_{1} + e_{2i}Z_{2} + \dots + e_{pi}Z_{p}$$

for i = 1,2, ..., p with unrealized population variance and covariance,

 $\operatorname{Var}(Y_i) = \mathbf{e}'_i \cdot \sum_{(p \times p)} \mathbf{e}_i = \mathbf{e}'_i \cdot \mathbf{\rho} \cdot \mathbf{e}_i = \lambda_i$

for i = 1,2, ..., *p and*

$$\operatorname{Cov}(Y_i, Y_k) = \frac{\mathbf{e}'_i}{(1 \times p)} \cdot \sum_{(p \times p)} \cdot \frac{\mathbf{e}_k}{(p \times 1)} = \frac{\mathbf{e}'_i}{(1 \times p)} \cdot \frac{\boldsymbol{\rho}}{(p \times p)} \cdot \frac{\mathbf{e}_k}{(p \times 1)} = 0$$

for $i, k = 1, 2, ..., p, i \neq k$ [3, p. 437].

Proof.

Follows from Theorem 5.2.1 with Z_1, Z_2, \dots, Z_p in place of X_1, X_2, \dots, X_p and

 $\sum_{(p \times p)} = \frac{\rho}{(p \times p)} \text{ in place of } \sum_{(p \times p)} \blacksquare$

Thus, the population principal components of $\mathbf{Z}_{(p \times 1)}$, Y_i , are given by

$$Y_{1} = \mathbf{e}_{1}' \cdot \mathbf{Z}_{(p \times 1)} = e_{11}Z_{1} + e_{21}Z_{2} + \dots + e_{p1}Z_{p}$$

$$Y_{2} = \mathbf{e}_{2}' \cdot \mathbf{Z}_{(p \times 1)} = e_{12}Z_{1} + e_{22}Z_{2} + \dots + e_{p2}Z_{p}$$

$$\vdots \qquad \vdots$$

$$Y_{p} = \mathbf{e}_{p}' \cdot \mathbf{Z}_{(p \times 1)} = e_{1p}Z_{1} + e_{2p}Z_{2} + \dots + e_{pp}Z_{p}$$

or in matrix notation,

$$\begin{split} \mathbf{Y}_{(p\times1)} &= \begin{bmatrix} \mathbf{Y}_1\\ \mathbf{Y}_2\\ \vdots\\ \mathbf{Y}_p\\ (p\times1) \end{bmatrix} = \begin{bmatrix} \mathbf{e}_1' & \cdot \mathbf{Z}\\ (1\times p) & (p\times1)\\ \mathbf{e}_2' & \cdot \mathbf{Z}\\ (1\times p) & (p\times1)\\ \vdots\\ \mathbf{e}_p' & \cdot \mathbf{Z}\\ (1\times p) & (p\times1) \end{bmatrix} = \begin{bmatrix} e_{11} & e_{21} & \cdots & e_{p1}\\ e_{12} & e_{22} & \cdots & e_{p2}\\ \vdots & \vdots & \ddots & \vdots\\ e_{1p} & e_{2p} & \cdots & e_{pp} \end{bmatrix} \begin{bmatrix} Z_1\\ Z_2\\ \vdots\\ Z_p\\ (p\times1) \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{E}_{(p\times p)}' & \mathbf{Z}\\ (p\times1) \end{bmatrix} = \begin{bmatrix} \mathbf{E}_{(p\times p)}' & (\mathbf{V}_{(p\times1)}^{1/2})^{-1} \cdot \left(\mathbf{X}\\ (p\times1) - \mathbf{\mu}_{\mathbf{X}} \right) \right) \\ &= \begin{bmatrix} \mathbf{E}_{(p\times p)}' & \mathbf{V}_{(p\times p)}^{-1/2} \cdot \left(\mathbf{X}\\ (p\times1) - \mathbf{\mu}_{\mathbf{X}} \right) \right) \end{split}$$

Theorem 5.3.2 (Total Standardized Population Variance). Let $\sum_{(p \times 1)} be a standardized$ population random vector for continuous random variables defined in Definition 3.4.1. with associated positive-definite standardized variance-covariance matrix $\sum_{(p \times p)} = \bigcap_{(p \times p)} defined in Theorem 3.4.4. Let \sum_{(p \times p)} = \bigcap_{(p \times p)} have eigenvalue and$ normalized-eigenvector pairs $\left(\lambda_i, \mathbf{e}_i\right), i = 1, 2, ..., p$ where $\lambda_1 > \lambda_2 > \cdots > \lambda_p > 0$.

$$Let Y_1 = \mathbf{e}'_1 \cdot \mathbf{Z}_{(1 \times p)}, Y_2 = \mathbf{e}'_2 \cdot \mathbf{Z}_{(1 \times p)}, \dots, Y_p = \mathbf{e}'_p \cdot \mathbf{Z}_{(p \times 1)} be the population principal$$

components of $\mathbf{Z}_{(p \times 1)}$ *.*

Then the total standardized population variance

$$\sum_{i=1}^{p} \operatorname{Var}(Z_i) = \sum_{i=1}^{p} \operatorname{Var}(Y_i) = p.$$

Proof.

Follows from Theorem 5.2.2 with Z_1, Z_2, \dots, Z_p in place of X_1, X_2, \dots, X_p and

 $\sum_{(p \times p)} = \frac{\rho}{(p \times p)} \text{ in place of } \sum_{(p \times p)} \blacksquare$

Hence,

Total standardized population variance = $1 + 1 + \dots + 1$

$$= p$$
$$= \lambda_1 + \lambda_2 + \dots + \lambda_p.$$

$$\begin{pmatrix} Proportion of total \\ standardized population variance \\ due to ith population principal \\ component of Z \\ (p \times 1) \end{pmatrix} = \frac{\lambda_i}{p} \qquad i = 1, 2, ..., p$$

and

$$\begin{pmatrix} \text{Proportion of total standardized} \\ \text{population variance due to the first } k \\ \text{population principal components of } \mathbf{Z}_{(p \times 1)} \end{pmatrix} = \frac{\sum_{i=1}^{k} \lambda_i}{p} \qquad k < p$$

[3, p. 437].

5.4 Sample Principal Components

Theorem 5.4.1 (ith Sample Principal Component). Let random vectors

 \mathbf{X}_1 , \mathbf{X}_2 , ..., \mathbf{X}_n constitute a multivariate random sample defined in Definition $(p \times 1)$ $(p \times 1)$

4.2.2 with associated positive-definite sample variance-covariance matrix $\mathbf{S}_{\mathbf{X}}_{(p \times p)}$

defined in Theorem 4.4.2. Let $S_{\mathbf{X}}_{(p \times p)}$ have sample eigenvalue and normalized-

eigenvector pairs $\left(\hat{\lambda}_{i}, \hat{\mathbf{e}}_{i}\right)$, i = 1, 2, ..., p where $\hat{\lambda}_{1} > \hat{\lambda}_{2} > \cdots > \hat{\lambda}_{p} > 0$. Then the

unrealized ith sample principal component is of the form

$$\hat{Y}_{i} = \hat{\mathbf{e}}'_{i} \cdot \sum_{\substack{(p \times 1) \\ (1 \times p)}} \mathbf{X}_{i} = \begin{bmatrix} \hat{e}_{1i}, \hat{e}_{2i}, \dots, \hat{e}_{pi} \end{bmatrix} \cdot \begin{bmatrix} X_{1} \\ X_{2} \\ \vdots \\ X_{p} \end{bmatrix} = \hat{e}_{1i}X_{1} + \hat{e}_{2i}X_{2} + \dots + \hat{e}_{pi}X_{p}$$

for *i* = 1,2, ..., *p* with unrealized quantity on the *j*th multivariate sample observation

٦7

$$\hat{Y}_{ji} = \hat{\mathbf{e}}'_{i} \cdot \mathbf{X}_{j} = \begin{bmatrix} \hat{e}_{1i}, \hat{e}_{2i}, \dots, \hat{e}_{pi} \end{bmatrix} \cdot \begin{bmatrix} X_{j1} \\ X_{j2} \\ \vdots \\ X_{jp} \end{bmatrix} = \hat{e}_{1i}X_{j1} + \hat{e}_{2i}X_{j2} + \dots + \hat{e}_{pi}X_{jp}$$

for j = 1,2, ..., *n*, *with unrealized sample variance and covariance*

$$\operatorname{var}(\hat{Y}_i) = \hat{\mathbf{e}}'_i \cdot \mathbf{S}_{\mathbf{X}} \cdot \hat{\mathbf{e}}_i = \hat{\lambda}_i$$
$$(1 \times p) \cdot (p \times p) \cdot (p \times 1) = \hat{\lambda}_i$$

for *i* = 1,2, ..., *p* and

$$\operatorname{cov}(\hat{Y}_i, \hat{Y}_k) = \hat{\mathbf{e}}'_i \cdot \mathbf{S}_{\mathbf{X}} \cdot \hat{\mathbf{e}}_k = 0$$
$$(1 \times p) \cdot (p \times p) \cdot (p \times 1) = 0$$

One can write the *n* sample principal components in matrix notation

$$\begin{split} \hat{\mathbf{Y}}_{(n\times p)} &= \underbrace{\mathbf{X}}_{(n\times p)} \cdot \widehat{\mathbf{E}}_{(p\times p)} \\ &= \begin{bmatrix} X_{11} & X_{12} & \cdots & X_{1p} \\ X_{21} & X_{22} & \cdots & X_{2p} \\ \vdots & \vdots & & \vdots \\ X_{j1} & X_{j2} & \cdots & X_{jp} \\ \vdots & \vdots & & \vdots \\ X_{n1} & X_{n2} & \cdots & X_{np} \end{bmatrix} \begin{bmatrix} \hat{e}_{11} & \hat{e}_{12} & \cdots & \hat{e}_{1i} & \cdots & \hat{e}_{1p} \\ \hat{e}_{21} & \hat{e}_{22} & \cdots & \hat{e}_{2i} & \cdots & \hat{e}_{2p} \\ \vdots & \vdots & & \vdots & & \vdots \\ \hat{e}_{p1} & \hat{e}_{p2} & \cdots & \hat{e}_{pi} & \cdots & \hat{e}_{pp} \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{X}_{1}' \cdot \widehat{\mathbf{e}}_{1} & \mathbf{X}_{1}' \cdot \widehat{\mathbf{e}}_{2} & \cdots & \mathbf{X}_{1}' \cdot \widehat{\mathbf{e}}_{i} & \cdots & \mathbf{X}_{1}' \cdot \widehat{\mathbf{e}}_{p} \\ (n\times p) & (p\times 1) & (1\times p) & (p\times 1) & (1\times p) & (p\times 1) \\ \mathbf{X}_{2}' \cdot \widehat{\mathbf{e}}_{1} & \mathbf{X}_{2}' \cdot \widehat{\mathbf{e}}_{2} & \cdots & \mathbf{X}_{2}' \cdot \widehat{\mathbf{e}}_{i} & \cdots & \mathbf{X}_{2}' \cdot \widehat{\mathbf{e}}_{p} \\ (1\times p) & (p\times 1) & (1\times p) & (p\times 1) & (1\times p) & (p\times 1) & (1\times p) & (p\times 1) \\ \vdots & \vdots & \vdots & & \vdots \\ \mathbf{X}_{j}' \cdot \widehat{\mathbf{e}}_{1} & \mathbf{X}_{j}' \cdot \widehat{\mathbf{e}}_{2} & \cdots & \mathbf{X}_{j}' \cdot \widehat{\mathbf{e}}_{i} & \cdots & \mathbf{X}_{j}' \cdot \widehat{\mathbf{e}}_{p} \\ (1\times p) & (p\times 1) & (1\times p) & (p\times 1) & (1\times p) & (p\times 1) & (1\times p) & (p\times 1) \\ \vdots & \vdots & & \vdots & & \vdots \\ \mathbf{X}_{n}' \cdot \widehat{\mathbf{e}}_{1} & \mathbf{X}_{n}' \cdot \widehat{\mathbf{e}}_{2} & \cdots & \mathbf{X}_{n}' \cdot \widehat{\mathbf{e}}_{i} & \cdots & \mathbf{X}_{n}' \cdot \widehat{\mathbf{e}}_{p} \\ (1\times p) & (p\times 1) & (1\times p) & (p\times 1) & (1\times p) & (p\times 1) & (1\times p) & (p\times 1) \\ \vdots & & \vdots & & \vdots & & \vdots \\ \mathbf{X}_{n}' \cdot \widehat{\mathbf{e}}_{1} & \mathbf{X}_{n}' \cdot \widehat{\mathbf{e}}_{2} & \cdots & \mathbf{X}_{n}' \cdot \widehat{\mathbf{e}}_{i} & \cdots & \mathbf{X}_{n}' \cdot \widehat{\mathbf{e}}_{p} \\ (1\times p) & (p\times 1) & (1\times p) & (p\times 1) & (1\times p) & (p\times 1) & (1\times p) & (p\times 1) \\ \vdots & & & \vdots & & & \vdots \\ \mathbf{X}_{n}' \cdot \widehat{\mathbf{e}}_{1} & \mathbf{X}_{n}' \cdot \widehat{\mathbf{e}}_{2} & \cdots & \mathbf{X}_{n}' \cdot \widehat{\mathbf{e}}_{i} & \cdots & \mathbf{X}_{n}' \cdot \widehat{\mathbf{e}}_{p} \\ (1\times p) & (p\times 1) & (1\times p) & (p\times 1) & (1\times p) & (p\times 1) \\ \end{array}$$

Using Definition 2.1.11 inner (dot) product of two vectors

$$= \begin{bmatrix} \hat{\mathbf{e}}'_{1} \cdot \mathbf{X}_{1} & \hat{\mathbf{e}}'_{2} \cdot \mathbf{X}_{1} & \cdots & \hat{\mathbf{e}}'_{i} \cdot \mathbf{X}_{1} & \cdots & \hat{\mathbf{e}}'_{p} \cdot \mathbf{X}_{1} \\ (1 \times p) & (p \times 1) & (1 \times p) & (p \times 1) & (1 \times p) & (p \times 1) & (1 \times p) & (p \times 1) \\ \hat{\mathbf{e}}'_{1} \cdot \mathbf{X}_{2} & \hat{\mathbf{e}}'_{2} \cdot \mathbf{X}_{2} & \cdots & \hat{\mathbf{e}}'_{i} \cdot \mathbf{X}_{2} & \cdots & \hat{\mathbf{e}}'_{p} \cdot \mathbf{X}_{2} \\ (1 \times p) & (p \times 1) & (1 \times p) & (p \times 1) & (1 \times p) & (p \times 1) & (1 \times p) & (p \times 1) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \hat{\mathbf{e}}'_{1} \cdot \mathbf{X}_{j} & \hat{\mathbf{e}}'_{2} \cdot \mathbf{X}_{j} & \cdots & \hat{\mathbf{e}}'_{i} \cdot \mathbf{X}_{j} & \cdots & \hat{\mathbf{e}}'_{p} \cdot \mathbf{X}_{j} \\ (1 \times p) & (p \times 1) & (1 \times p) & (p \times 1) & (1 \times p) & (p \times 1) & (1 \times p) & (p \times 1) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \hat{\mathbf{e}}'_{1} \cdot \mathbf{X}_{n} & \hat{\mathbf{e}}'_{2} \cdot \mathbf{X}_{n} & \cdots & \hat{\mathbf{e}}'_{i} \cdot \mathbf{X}_{n} & \cdots & \hat{\mathbf{e}}'_{p} \cdot \mathbf{X}_{n} \\ (1 \times p) & (p \times 1) & (1 \times p) & (p \times 1) & (1 \times p) & (p \times 1) & (1 \times p) & (p \times 1) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \hat{\mathbf{e}}'_{1} \cdot \mathbf{X}_{n} & \hat{\mathbf{e}}'_{2} \cdot \mathbf{X}_{n} & \cdots & \hat{\mathbf{e}}'_{i} \cdot \mathbf{X}_{n} & \cdots & \hat{\mathbf{e}}'_{p} \cdot \mathbf{X}_{n} \\ (1 \times p) & (p \times 1) & (1 \times p) & (p \times 1) & (1 \times p) & (p \times 1) \\ (n \times p) & (n \times p) &$$

$$\mathbf{x}'_{(1\times n)} \cdot \mathbf{y}_{(n\times 1)} = \mathbf{y}'_{(1\times n)} \cdot \mathbf{x}_{(n\times 1)} \Rightarrow$$

$$= \begin{bmatrix} \hat{Y}_{11} & \hat{Y}_{12} & \cdots & \hat{Y}_{1i} & \cdots & \hat{Y}_{1p} \\ \hat{Y}_{21} & \hat{Y}_{22} & \cdots & \hat{Y}_{2i} & \cdots & \hat{Y}_{2p} \\ \vdots & \vdots & & \vdots & & \vdots \\ \hat{Y}_{j1} & \hat{Y}_{j2} & \cdots & \hat{Y}_{ji} & \cdots & \hat{Y}_{jp} \\ \vdots & \vdots & & \vdots & & \vdots \\ \hat{Y}_{n1} & \hat{Y}_{n2} & \cdots & \hat{Y}_{ni} & \cdots & \hat{Y}_{np} \end{bmatrix}$$

Theorem 5.4.2 (Total Sample Variance). *Let random vectors* \mathbf{X}_1 , \mathbf{X}_2 , ..., \mathbf{X}_n ($p \times 1$) ($p \times 1$), $(p \times 1)$, $(p \times 1)$

constitute a multivariate random sample defined in Definition 4.2.2. with associated positive-definite sample variance-covariance matrix $\mathbf{S}_{\mathbf{X}}$ defined in Theorem 4.4.2.

Let $\mathbf{S}_{\mathbf{X}}$ have sample eigenvalue and normalized-eigenvector pairs $\left(\hat{\lambda}_{i}, \hat{\mathbf{e}}_{i}\right)$,

i = 1, 2, ..., p where $\hat{\lambda}_1 > \hat{\lambda}_2 > \cdots > \hat{\lambda}_p > 0$. Let the unrealized sample principal

components be of the form $\hat{Y}_i = \hat{\mathbf{e}}'_i \cdot \mathbf{X}_{(1 \times p)}$ *with jth multivariate sample*

observation $\hat{Y}_{ji} = \hat{\mathbf{e}}'_i \cdot \mathbf{X}_j$. Then the **total sample variance** $(1 \times p) \cdot (p \times 1)$.

$$S_{11} + S_{22} + \dots + S_{pp} = \sum_{i=1}^{p} S_{ii} = \hat{\lambda}_1 + \hat{\lambda}_2 + \dots + \hat{\lambda}_p = \sum_{i=1}^{p} \operatorname{Var}(\hat{Y}_i).$$

Consequently,

$$\begin{pmatrix} \text{Proportion of total} \\ \text{sample variance} \\ \text{due to } i\text{th sample} \\ \text{principal component} \end{pmatrix} = \frac{\hat{\lambda}_i}{\hat{\lambda}_1 + \hat{\lambda}_2 + \dots + \hat{\lambda}_p} \qquad i = 1, 2, \dots, p$$

[3, p. 442] and

$$\begin{pmatrix} \text{Proportion of total sample} \\ \text{variance due to the first } k \\ \text{sample principal components} \end{pmatrix} = \frac{\sum_{i=1}^{k} \hat{\lambda}_i}{\hat{\lambda}_1 + \hat{\lambda}_2 + \dots + \hat{\lambda}_p} \qquad k$$

We shall denote the sample principal components by $\hat{Y}_1, \hat{Y}_2, ..., \hat{Y}_p$, irrespective of whether they are obtained from $\mathbf{S}_{\mathbf{X}}$ or $\mathbf{S}_{\mathbf{Z}} = \mathbf{R}_{(p \times p)}$. The components constructed from $\mathbf{S}_{\mathbf{X}}$ and $\mathbf{S}_{\mathbf{Z}} = \mathbf{R}_{(p \times p)}$ are not the same, in general, but it will be clear from the context which matrix is being used, and the single notation \hat{Y}_i is convenient. It is also convenient to label the component coefficient vectors $\hat{\mathbf{e}}_i$ and the component $\hat{\lambda}_i$ for both situations [3, p. 443].

5.5 Sample Principal Components for Standardized

Samples

Theorem 5.5.1 (*i*th Sample Principal Component of Z). Let random

vectors \mathbf{Z}_1 , \mathbf{Z}_2 , ..., \mathbf{Z}_n constitute a standardized multivariate random sample $(p \times 1)$ $(p \times 1)$

defined in Theorem 4.6.1. with associated positive-definite sample variance-

covariance matrix $\mathbf{S}_{\mathbf{Z}} = \mathbf{R}_{(p \times p)}$ *defined in Theorem 4.8.2. Let* $\mathbf{S}_{\mathbf{Z}} = \mathbf{R}_{(p \times p)}$ *have*

sample eigenvalue and normalized-eigenvector pairs $\left(\hat{\lambda}_{i}, \hat{\mathbf{e}}_{i}, \mathbf{e}_{i}, \mathbf{e}_{i}$

 $\hat{\lambda}_1 > \hat{\lambda}_2 > \cdots > \hat{\lambda}_p > 0$. Then the unrealized **i**th sample principal component of

 $\sum_{(n \times p)}$ is of the form

$$\hat{Y}_{i} = \hat{\mathbf{e}}'_{i} \cdot \mathbf{Z}_{(1 \times p)} \cdot \begin{bmatrix} \hat{e}_{1i}, \hat{e}_{2i}, \dots, \hat{e}_{pi} \end{bmatrix} \cdot \begin{bmatrix} Z_{1} \\ Z_{2} \\ \vdots \\ Z_{p} \end{bmatrix} = \hat{e}_{1i} Z_{1} + \hat{e}_{2i} Z_{2} + \dots + \hat{e}_{pi} Z_{p}$$

for i = 1, 2, ..., p with unrealized quantity on the jth standardized multivariate

sample observation

$$\hat{Y}_{ji} = \hat{\mathbf{e}}'_{i} \cdot \mathbf{Z}_{j}_{(1 \times p)} = \begin{bmatrix} \hat{e}_{1i}, \hat{e}_{2i}, \dots, \hat{e}_{pi} \end{bmatrix} \cdot \begin{bmatrix} Z_{j1} \\ Z_{j2} \\ \vdots \\ Z_{jp} \end{bmatrix} = \hat{e}_{1i} Z_{j1} + \hat{e}_{2i} Z_{j2} + \dots + \hat{e}_{pi} Z_{jp}$$

$$(p \times 1)$$

for j = 1,2, ..., *n with unrealized sample variance and covariance*

$$\operatorname{var}(\hat{Y}_{i}) = \hat{\mathbf{e}}_{i}' \cdot \mathbf{S}_{\mathbf{Z}} \cdot \hat{\mathbf{e}}_{i} = \hat{\mathbf{e}}_{i}' \cdot \mathbf{R} \cdot \hat{\mathbf{e}}_{i} = \hat{\lambda}_{i}$$

for i = 1, 2, ..., p *and*

$$\operatorname{cov}(\hat{Y}_i, \hat{Y}_k) = \frac{\hat{\mathbf{e}}'_i}{(1 \times p)} \cdot \frac{\mathbf{S}_{\mathbf{Z}}}{(p \times p)} \cdot \frac{\hat{\mathbf{e}}_k}{(p \times 1)} = \frac{\hat{\mathbf{e}}'_i}{(1 \times p)} \cdot \frac{\mathbf{R}}{(p \times p)} \cdot \frac{\hat{\mathbf{e}}_k}{(p \times 1)} = 0$$

for $i, k = 1, 2, ..., p, i \neq k$ *given in Definition 4.9.7* [3, p. 451].

One can write the *n* sample principal components of $\sum_{(n \times p)} in$ matrix notation

$$\begin{split} \hat{\mathbf{Y}}_{(n \times p)} &= \mathbf{Z} & \hat{\mathbf{E}}_{(n \times p)} \cdot \hat{\mathbf{E}}_{(p \times p)} \\ &= \begin{bmatrix} Z_{11} & Z_{12} & \cdots & Z_{1p} \\ Z_{21} & Z_{22} & \cdots & Z_{2p} \\ \vdots & \vdots & & \vdots \\ Z_{j1} & Z_{j2} & \cdots & Z_{jp} \\ \vdots & \vdots & & \vdots \\ Z_{n1} & Z_{n2} & \cdots & Z_{np} \end{bmatrix} \begin{bmatrix} \hat{e}_{11} & \hat{e}_{12} & \cdots & \hat{e}_{1i} & \cdots & \hat{e}_{1p} \\ \hat{e}_{21} & \hat{e}_{22} & \cdots & \hat{e}_{2i} & \cdots & \hat{e}_{2p} \\ \vdots & \vdots & & \vdots & & \vdots \\ \hat{e}_{p1} & \hat{e}_{p2} & \cdots & \hat{e}_{pi} & \cdots & \hat{e}_{pp} \end{bmatrix} \\ &= \begin{bmatrix} \hat{Y}_{11} & \hat{Y}_{12} & \cdots & \hat{Y}_{1i} & \cdots & \hat{Y}_{1p} \\ \hat{Y}_{21} & \hat{Y}_{22} & \cdots & \hat{Y}_{2i} & \cdots & \hat{Y}_{2p} \\ \vdots & \vdots & & \vdots & & \vdots \\ \hat{Y}_{j1} & \hat{Y}_{j2} & \cdots & \hat{Y}_{ji} & \cdots & \hat{Y}_{jp} \\ \vdots & \vdots & & \vdots & & \vdots \\ \hat{Y}_{n1} & \hat{Y}_{n2} & \cdots & \hat{Y}_{ni} & \cdots & \hat{Y}_{np} \end{bmatrix} \end{split}$$

Theorem 5.5.2 (Total Standardized Sample Variance). Let random

vectors \mathbf{Z}_1 , \mathbf{Z}_2 , ..., \mathbf{Z}_n constitute a standardized multivariate random sample $(p \times 1)$ $(p \times 1)$ $(p \times 1)$

defined in Theorem 4.6.1 with associated positive-definite standardized sample

variance-covariance matrix $\mathbf{S}_{\mathbf{Z}} = \mathbf{R}_{(p \times p)}$ *defined in Theorem 4.8.2. Let* $\mathbf{S}_{\mathbf{Z}} = \mathbf{R}_{(p \times p)}$

have sample eigenvalue and normalized-eigenvector pairs $(\hat{\lambda}_i, \hat{\mathbf{e}}_i)$, i = 1, 2, ..., p

where $\hat{\lambda}_1 > \hat{\lambda}_2 > \cdots > \hat{\lambda}_p > 0$. Let the unrealized sample principal components of $\sum_{(n \times p)} be of the form \hat{Y}_i = \hat{\mathbf{e}}'_i \cdot \sum_{(p \times 1)} with jth standardized multivariate sample$

observations $\hat{Y}_{ji} = \hat{\mathbf{e}}'_i \cdot \mathbf{Z}_j$. (1×p) (p×1)

Then the total standardized sample variance

$$1 + 1 + \dots + 1 = p = \sum_{i=1}^{p} S_{z,ii} = \sum_{i=1}^{p} R_{ii} = \hat{\lambda}_1 + \hat{\lambda}_2 + \dots + \hat{\lambda}_p = \sum_{i=1}^{p} \operatorname{Var}(\hat{Y}_i).$$

Consequently,

$$\begin{pmatrix} Proportion of total \\ standardized sample variance \\ due to the ith sample principal \\ component of $\begin{array}{c} \mathbf{Z} \\ (n \times p) \end{array} \end{pmatrix} = \frac{\hat{\lambda}_i}{p} \qquad i = 1, 2, \dots, p$$$

and

$$\begin{pmatrix} \text{Proportion of total standardized} \\ \text{sample variance due to the first } k \\ \text{sample principal components of } \sum_{(n \times p)}^{\mathbf{Z}} \end{pmatrix} = \frac{\sum_{i=1}^{k} \hat{\lambda}_{i}}{p} \qquad k < p.$$

A rule of thumb suggests retaining only those components whose variances $\hat{\lambda}_i$ are greater than unity or, equivalently, only those components which, individually, explain at least a proportion 1/p of the total variance. This rule does not have a great deal of theoretical support, however, and it should not be applied blindly. Also, a scree plot is useful for selecting the appropriate number of components [3, p. 451].

Chapter 6

Results and Discussion

6.1 R Programming Language

Analysis of data is conducted using R version 3.6.2 (2019-12-12) -- "Dark and

Stormy Night". R is an open source software for statistical computing and graphics.

The latest version can be downloaded at R: The R Project for Statistical Computing

website <u>https://www.r-project.org/</u>.

6.2 Univariate Distribution Analysis

6.2.1 Descriptives for US Crime 2018

	vars	n	sd	min	q1	median	mean	q3	max	range
MURDER	1	327	5.45	0	1.95	3.9	5.11	6.25	60.9	60.9
RAPE	2	327	26.36	13	33.15	44.8	50.99	62.2	200.1	187.1
ROBBERY	3	327	61.93	1.2	33.3	55.5	70.52	87.95	473.2	472
ASSAULT	4	327	182.33	30.2	152.1	233.6	270.11	323.7	1477.8	1447.6
BURGLARY	5	327	233.2	87.3	264.1	393.9	435.9	557.1	1576.1	1488.8
LARCENY	6	327	734.07	488.5	1282	1657.2	1748.4	2045.1	8558.1	8069.6
VEHICLE	7	327	160.58	13.7	103	166.7	215.18	281.4	970.9	957.2

Table 6.2.1: Descriptives for US Crime 2018

Table 6.2.1 gives the descriptives for 327 US metropolitan statistical areas in 2018 for violent crime and property crime per 100,000 residents.

6.2.2 Distributions of US Crime 2018

6.2.2.1 Murder Distribution



Figure 6.2.1: Murder Distribution Plots

Based on the density and histogram in Figure 6.2.1, the distribution of Murder looks right skewed. The lower left plot in Figure 6.2.1 is a Normal QQ-Plot for Murder that shows a clear lack of normality. One can use Shapiro-Wilk test for normality with $\alpha = 0.1$ to confirm this assertion. That is,

- H_0 : Population Distribution for Murder is Normal
- H_1 : Population Distribution for Murder is not Normal

$W = 0.61016; p - value \approx 0$

Thus, as expected, one rejects H_0 . There is sufficient evidence to say that the population distribution of Murder is not normally distributed. However, the distribution of Murder could be lognormal. The lower right plot in Figure 6.2.1 is a Lognormal QQ-Plot for Murder that shows a clear potential of lognormality, along with the density and histogram. One can use the same Shapiro-Wilk test to test for lognormality by a simple log transformation on \mathbf{X}_{Murder} . Indeed, this is due to the (327×1) fact that $X_i \sim \text{Lognormal} \Rightarrow \log(X_i) \sim \text{Normal} [8].$

 H_0 : Population Distribution for Murder is Lognormal H_1 : Population Distribution for Murder is not Lognormal p - value doesn't exist

The p – value doesn't exist because seven metropolitan statistical areas have murder rates of 0. As a result, the transformation from X_{Murder} to $log\left(X_{Murder}\right)$ cannot be completed and the Shapiro-Wilk test will not compute a p – value. Nevertheless, using the Lognormal QQ-Plot one can cautiously assume the population distribution of Murder is approximately lognormal.

It has been found that all the outliers of Murder are located at the upper end of the distribution. These metropolitan statistical areas correspond places with extremely high murder rates per 100,000 residents. Furthermore, it may be of interest to see the areas in the lowest 2.5% of the Murder distribution for 2018.



Figure 6.2.2: Murder Outliers and Lower 2.5% of Sample

The left plot in Figure 6.2.2 shows the seven metropolitan statistical areas with a murder rate of 0. The right plot in Figure 6.2.2 highlights three areas with radically high murder rates per 100,000; namely, St Louis (60.9), Detroit (38.9), and New Orleans (37.1).



Figure 6.2.3: Rape Distribution Plots

Based on the density and histogram in Figure 6.2.3, the distribution of rape looks right skewed with several outliers. Next, one uses Shapiro-Wilk test to test for normality and lognormality with $\alpha = 0.1$.

 H_0 : Population Distribution for Rape is Normal

 H_1 : Population Distribution for Rape is not Normal

 $W = 0.85053; p - value \approx 0$

 H_0 : Population Distribution for Rape is Lognormal

 H_1 : Population Distribution for Rape is not Lognormal

$$p - value \approx 0.164$$

One rejects H_0 for normality and fails to reject H_0 for lognormality. Yet the Lognormal QQ-Plot appears to contradict the hypothesis test result. Thus, more work should be done to resolve this inconsistency. However, learning the true distribution of rape is not of major interest, so one can move on.



Figure 6.2.4: Rape Outliers and Lower 2.5% of Sample

The right plot in Figure 6.2.4 focuses one's attention to four areas with extremely high rape rates per 100,000; specifically, Anchorage (200.1), Myrtle Beach (190), New Orleans (171.8), and Detroit (147.2). Anchorage has long time been known for

its high rape rates. The question of interest is why? Some have posed that it is related to the high male-to-female ratio. Others have said it is due to the long winters and physical isolation of individuals. While others have stated that the issue is established upon patriarchy and capitalism, which objectifies and commodifies women as the property of men [9]. Whereas, Myrtle Beach and New Orleans are vacation and party destinations which could lead to increased sexual assault. Finally, remember, that Detroit and New Orleans also had dangerously high Murder rates. One should pay attention to these metropolitan statistical areas that repeatedly show up in the high-ranking crime category.



6.2.2.3 Robbery Distribution

Figure 6.2.5: Robbery Distribution Plots

Based on the density and histogram in Figure 6.2.5, the distribution of Robbery looks right skewed with several outliers. Next, one uses Shapiro-Wilk test for testing normality and lognormality with $\alpha = 0.1$.

 H_0 : Population Distribution for Robbery is Normal

 H_1 : Population Distribution for Robbery is not Normal

 $W = 0.73871; p - value \cong 0$

 H_0 : Population Distribution for Robbery is Lognormal

 H_1 : Population Distribution for Robbery is not Lognormal

 $p - value \cong 0$

One rejects H_0 for normality and lognormality.



Figure 6.2.6: Robbery Outliers and Lower 2.5% of Sample

The right plot in Figure 6.2.6 has some repeatedly high-ranking metropolitan statistical areas for crime, in general, and in robbery as well. The names one hasn't seen yet in the univariate outliers list are Houston, Albuquerque, Stockton, and San Francisco.



6.2.2.4 Assault Distribution

Figure 6.2.7: Assault Distribution Plots

Based on the density and histogram in Figure 6.2.7, the distribution of Assault looks right skewed with several outliers. Next, one uses Shapiro-Wilk test for normality and lognormality with $\alpha = 0.1$.

 H_0 : Population Distribution for Assault is Normal

*H*₁ : Population Distribution for Assault is not Normal

 $W = 0.8109; p - value \approx 0$

*H*⁰ : Population Distribution for Assault is Lognormal

 H_1 : Population Distribution for Assault is not Lognormal

 $p - value \cong 0.3352$

One rejects H_0 for normality and fails to reject H_0 for lognormality. Similar to rape, the Lognormal QQ-Plot for assault, appears to contradict the hypothesis test result.



Figure 6.2.8: Assault Outliers and Lower 2.5% of Sample

The right plot in Figure 6.2.8 features four areas with drastically higher assault rates per 100,000. Detroit (1477.8), St Louis (1165.6), Little Rock (1130.5), and Farmington (1006.4). Interesting, two metropolitan statistical areas are in New Mexico: Farmington and Albuquerque. Similarly, three metropolitan statistical

areas are in Texas: Lubbock, Odessa, and Houston. Immediately we can see many of these outliers have been seen in previous plots.

6.2.2.5 Burglary Distribution



Figure 6.2.9: Burglary Distribution Plots

Based on the density and histogram in Figure 6.2.9, the distribution of Burglary looks right skewed with several outliers. Next, one uses Shapiro-Wilk test for normality and lognormality with $\alpha = 0.1$.
H_0 : Population Distribution for Burglary is Normal

*H*₁ : Population Distribution for Burglary is not Normal

 $W = 0.90035; p - value \approx 0$

 H_0 : Population Distribution for Burglary is Lognormal

 H_1 : Population Distribution for Burglary is not Lognormal

 $p - value \cong 0.4335$

One rejects H_0 for normality and fails to reject H_0 for lognormality. Similar to rape and assault, the Lognormal QQ-Plot for burglary, appears to contradict the hypothesis test result.



Figure 6.2.10: Burglary Outliers and Lower 2.5% of Sample

The right plot in Figure 6.2.10 contains two areas with larger burglary rates per 100,000: Lake Charles (1576.1) and Hot Springs (1421.6). What is noteworthy is these areas have not shown up on any other of the other outlier plots.



Figure 6.2.11: Larceny Distribution Plots

Based on the density and histogram in Figure 6.2.11, the distribution of Larceny looks right skewed with several outliers. Next, one uses Shapiro-Wilk test for normality and lognormality with $\alpha = 0.1$.

 H_0 : Population Distribution for Larceny is Normal

 H_1 : Population Distribution for Larceny is not Normal

 $W = 0.90035; p - value \approx 0$

 H_0 : Population Distribution for Larceny is Lognormal

 H_1 : Population Distribution for Larceny is not Lognormal

$$p - value \approx 0.04679$$

One rejects H_0 for normality and lognormality.



Figure 6.2.12: Larceny Outliers and Lower 2.5% of Sample

The right plot in Figure 6.2.12 has one extreme crime area that stands out compared to the other outliers. Myrtle Beach's (8558.1) larceny crime rate is almost double any other of the outliers. Theft of person property is often higher in tourist destinations. It is surprising that Las Vegas is not one of the high-raking areas for this type of crime.



Figure 6.2.13: Vehicle Distribution Plots

Based on the density and histogram in Figure 6.2.11, the distribution of Vehicle looks right skewed with several outliers. Next, one uses Shapiro-Wilk test for normality and lognormality with $\alpha = 0.1$.

 H_0 : Population Distribution for Vehicle is Normal

 H_1 : Population Distribution for Vehicle is not Normal

 $W = 0.8565; p - value \approx 0$

 H_0 : Population Distribution for Vehicle is Lognormal

*H*₁ : Population Distribution for Vehicle is not LogNormal

$$p - value \approx 0.3367$$

One rejects H_0 for normality and fails to reject H_0 for lognormality.



Figure 6.2.14: Vehicle Outliers and Lower 2.5% of Sample

The right plot in Figure 6.2.14 does not show any metropolitan statistical areas where vehicle theft stands out significantly more than others.

6.3 Bivariate Distribution Analysis

6.3.1 Correlation Matrix for US Crime 2018



Figure 6.3.1: Correlation Matrix for US Crime 2018

In Figure 6.3.1, one can see strong positive sample correlation between murder and robbery, murder and assault, robbery and assault, burglary and larceny, and robbery and vehicular theft.



6.3.2 Contour-Scatter Matrix

Figure 6.3.2: Contour-Scatter Matrix

The upper diagonal of Figure 6.3.2 displays scatterplots for the seven US Crime 2018 characteristics (variables). One can see there is a dense cloud on the lower-left part of most of the scatterplots linked to areas where pairs of characteristics have lower or medium crime rates. In contrast, one can see less dense scatter in the upper-right of the scatterplots related to those areas where high to extremely crime rates exist.

The lower diagonal of Figure 6.3.2 displays contour plots where the 2-d density is colored with a lighter color for more dense regions and the 2-d density is colored darker for less dense regions. Specifically, the contour plots are a nice way to visualize the bivariate densities in two dimensions instead of in three dimensions. Here, with the contour plots, one can see the densest regions for each pair of variables, unlike in the upper diagonal where it is obscured by the larger number of dots scattered in close proximity.

6.4 Multivariate Distribution Analysis

6.4.1 Testing Multivariate Normality

Using the generalization of Shapiro-Wilk test (Villasenor-Alva and Gonzalez-Estrada 2009) for multivariate normality one can test

 H_0 : Population Distribution is Multivariate Normal H_1 : Population Distribution is not Multivariate Normal $W = 0.8513; p - value \cong 0$

Consequently, one rejects H_0 . There is sufficient evidence to say that the US Crime population distribution is not multivariate normal.

6.5 Sample PCA for Standardized US Crime 2018

When individual sample characteristics have vastly different ranges they are routinely standardized before running a principal components analysis [3, p. 439]. Otherwise the characteristics with the largest ranges will dominate the first few sample principal components. Hence, the first step in the principal components analysis is to standardize the US Crime 2018 data.

6.5.1 Descriptives for Standardized US Crime 2018

	vars	n	sd	min	q1	median	mean	q3	max	range
MURDER	1	327	1	-0.94	-0.58	-0.22	0	0.21	10.24	11.18
RAPE	2	327	1	-1.44	-0.68	-0.23	0	0.43	5.66	7.1
ROBBERY	3	327	1	-1.12	-0.60	-0.24	0	0.28	6.5	7.62
ASSAULT	4	327	1	-1.32	-0.65	-0.2	0	0.29	6.62	7.94
BURGLARY	5	327	1	-1.49	-0.74	-0.18	0	0.52	4.89	6.38
LARCENY	6	327	1	-1.72	-0.64	-0.12	0	0.40	9.28	10.99
VEHICLE	7	327	1	-1.25	-0.70	-0.3	0	0.41	4.71	5.96

 Table 6.5.1: Descriptives for Standardized US Crime 2018

In Table 6.5.1, one can see that all sample means are 0 and all sample standard deviations are 1. Further, the respective ranges are comparable in size. Now, elements in the standardized multivariate random sample matrix that are positive will be above the sample mean and elements that are negative will be below the sample mean.

6.5.2 Sample PCA for Standardized US Crime 2018

6.5.2.1 Explained Standardized Sample Variance by Principal

Component for US Crime 2018

Table 6.5.2: Explained Standardized Sample Variance by Principal Component

	y1	y2	y3	y4	y5	y6	y7
Eigenvalues	4.4138	0.7695	0.6568	0.4503	0.3226	0.2163	0.1707
% of Variance	63.05%	10.99%	9.38%	6.43%	4.61%	3.09%	2.44%
Cumulative %	63.05%	74.05%	83.43%	89.86%	94.47%	97.56%	100.00%

The first row of Table 6.5.2 displays the standardized sample variances for each of the sample principal components $(var(\hat{Y}_i) = \hat{\lambda}_i \text{ for } i = 1, ..., 7)$. The second row provides the percent of standardized sample variance due to the *i*th sample principal component $(\frac{\hat{\lambda}_i}{7} \cdot 100\%, i = 1, ..., 7)$. Finally, the third row shows the percent of standardized sample variance due to the first *k*th sample principal component $(\frac{\sum_{i=1}^k \hat{\lambda}_i}{7}, k \leq 7)$. One can see that the first three sample principal components account for 83.43% of the total standardized variation in the sample from US Crime 2018. Figure 6.5.1 gives us a way to visualize the relation between the standardized sample principal components and their percentages of explained standardized sample variance.





6.5.2.2 Sample Principal Components for Standardized Crime 2018

	y1	y2	y3	y4	y5	y6	y7
MURDER	-0.3704	0.4863	-0.3767	0.197	-0.3164	-0.4799	0.3394
RAPE	-0.2849	-0.7469	-0.5518	0.0673	-0.0436	-0.1588	-0.1574
ROBBERY	-0.4087	0.3387	-0.131	-0.3787	-0.1169	0.1769	-0.7161
ASSAULT	-0.4144	0.0985	-0.1529	0.4294	0.2454	0.709	0.2191
BURGLARY	-0.372	-0.0847	0.555	0.5261	0.1749	-0.3644	-0.3261
LARCENY	-0.3792	-0.2702	0.4419	-0.2523	-0.6337	0.193	0.2923
VEHICLE	-0.4009	-0.0313	0.0989	-0.5371	0.6261	-0.2013	0.328

Table 6.5.3: Sample Principal Components for Standardized Crime 2018

Given that the first three sample principal components yield 83.43% of the total standardized variation in the sample, there is no need to use the other four sample components in one's analysis.

attempted to be explained in the context of the subject matter. To demonstrate, sample principal component \hat{y}_1 has eigenvector components of roughly equal magnitudes. Thus, \hat{y}_1 can be considered a *general crime component*. If one was to explain a metropolitan statistical area's crime rate with one value, then \hat{y}_{j1} would be it. Most importantly because \hat{y}_1 maximizes the standardized sample variance $var(\hat{y}_1)$ subject to $\hat{\mathbf{e}}'_1 \cdot \hat{\mathbf{e}}_1 = 1$ and $cov(\hat{y}_1, \hat{y}_k) = 0, k = 2, ..., 7$. Notice that all the eigenvector components are negative; accordingly, an area with larger crime rates

would have a very negative value (in general).

$$\hat{y}_1 = -0.37 x_{\text{MURDER}} - 0.28 x_{\text{RAPE}} - 0.41 x_{\text{ROBBERY}} - 0.41 x_{\text{ASSAULT}}$$

 $-0.37 x_{\text{BURGLARY}} - 0.37 x_{\text{LARCENY}} - 0.40 x_{\text{VEHICLE}}$

with *j*th observation

$$\hat{y}_{j1} = -0.37x_{j,\text{MURDER}} - 0.28x_{j,\text{RAPE}} - 0.41x_{j,\text{ROBBERY}} - 0.41x_{j,\text{ASSAULT}} - 0.37x_{j,\text{BURGLARY}} - 0.37x_{j,\text{LARCENY}} - 0.40x_{j,\text{VEHICLE}}$$

Sample principal component \hat{y}_2 has largest eigenvector component magnitudes on murder and rape. Therefore, \hat{y}_2 could be deemed a *heinous crime component*. If the area has a much larger murder rate, then rape rate, \hat{y}_{j2} will likely stand out in the positive direction. If the area has a much larger rape rate, then murder rate, \hat{y}_{j2} will likely stand out in the negative direction. If the area has approximately equal values, then \hat{y}_{j2} will likely not stand out in either direction.

$$\hat{y}_2 = 0.49x_{\text{MURDER}} - 0.75x_{\text{RAPE}} + 0.34x_{\text{ROBBERY}} + 0.10x_{\text{ASSAULT}}$$

$$-0.08x_{\text{BURGLARY}} - 0.27x_{\text{LARCENY}} - 0.03x_{\text{VEHICLE}}$$

with *j*th observation

$$\hat{y}_{j2} = 0.49x_{j,\text{MURDER}} - 0.75x_{j,\text{RAPE}} + 0.34x_{j,\text{ROBBERY}} + 0.10x_{j,\text{ASSAULT}}$$

 $-0.08x_{j,\text{BURGLARY}} - 0.27x_{j,\text{LARCENY}} - 0.03x_{j,\text{VEHICLE}}$

Sample principal component \hat{y}_3 has negative eigenvector components for violent crime and positive eigenvector components for property crime. Immediately, \hat{y}_3 can be thought of as a *crime type component*. That is, areas with particularly negative \hat{y}_{j3} values will often have larger violent crime relative to property crime. Conversely, areas with larger property crime relative to violent crime will have more positive \hat{y}_{j3} values.

$$\hat{y}_3 = -0.38x_{\text{MURDER}} - 0.55x_{\text{RAPE}} - 0.13x_{\text{ROBBERY}} - 0.15x_{\text{ASSAULT}}$$
$$+0.56x_{\text{BURGLARY}} + 0.44x_{\text{LARCENY}} + 0.10x_{\text{VEHICLE}}$$

with *j*th observation

$$\hat{y}_{j3} = -0.38x_{j,\text{MURDER}} - 0.55x_{j,\text{RAPE}} - 0.13x_{j,\text{ROBBERY}} - 0.15x_{j,\text{ASSAULT}} + 0.56x_{j,\text{BURGLARY}} + 0.44x_{j,\text{LARCENY}} + 0.10x_{j,\text{VEHICLE}}$$

Note that explaining these principal components is not a perfect science and caution should be exercised when interpreting the \hat{y}_i 's in context of the data. Figure 6.5.2 gives a graphical interpretation of how the standardized characteristics contributed to the first three sample principal component derived from the US Crime 2018 data.





In Figure 6.5.2, the percent contribution of the kth standardized characteristic to the *i*th sample principal component is calculated as

Sample Contribution_{*ki*} =
$$\hat{e}_{ki}^2 \cdot 100\%$$

for k, i = 1, 2, ..., p because $\hat{\mathbf{e}}'_i \cdot \hat{\mathbf{e}}_i \cdot 100\% = 1 \cdot 100\% = 100\%$. Hence, \hat{e}^2_{ki} is the proportion contribution of the *k*th standardized characteristic to the *i*th sample principal component. To clarify further, $\hat{\mathbf{e}}'_i \cdot \hat{\mathbf{e}}_i$ represents the squared length or magnitude of the vector $\hat{\mathbf{e}}_i$ so $\hat{e}^2_{ki} = \hat{e}_{ki} \cdot \hat{e}_{ki}$ is the part that the standardized characteristic z_k that contributes to magnitude of, or squared length of, $\hat{\mathbf{e}}_i$.

6.5.2.3 Correlation Matrix for Sample Principal Components and

Standardized Crime 2018 Characteristics



Figure 6.5.3: Correlation Matrix for Sample Principal Components and Standardized Crime 2018 Characteristics

The upper-right triangle of Figure 6.5.3 displays the sample correlations between the (Standardized) US Crime 2018 characteristics as seen in Figure 6.3.1.The bottom-left triangle of Figure 6.5.3 shows the sample principal components are indeed uncorrelated because $cov(\hat{y}_i, \hat{y}_k) = 0 \forall i \neq k$.

In the right-bottom square of Figure 6.5.3, the correlations between the sample principal components and the standardized US Crime 2018 characteristics, can be seen. The interpretation of these sample correlations can lead to similar

interpretations as looking at \hat{e}_{ki} directly, but with some data, this is not true [3, p. 434]. For Figure 6.5.3, the correlations between sample principal components and the standardized US Crime 2018 characteristics match the original interpretations of the \hat{e}_{ki} 's.

To illustrate, the eigenvector components of \hat{y}_1 are all negative and nearly the same magnitude. Analogously, the correlations between the eigenvector components of \hat{y}_1 and the standardized US Crime 2018 characteristics are all strong negatively correlated. For the eigenvector components of \hat{y}_2 and the standardized US Crime 2018 characteristics, one can see a strong negative correlation between the standardized rape characteristic and its respective eigenvector component. In the same way, the standardized murder characteristic is positively correlated with its eigenvector counterpart. Principal component \hat{y}_3 has negative correlations with the violent crime characteristics and positive correlations with the property crime characteristics. Henceforth, the correlation structure between the sample principal components and the standardized US Crime 2018 characteristics agree with the signs and magnitudes of the \hat{e}_{ki} 's.

6.5.2.4 Scatterplots for Sample Principal Components from



Standardized US Crime 2018

Figure 6.5.4: Scatterplot for $\hat{y}_2 \sim \hat{y}_1$

Figure 6.5.4 plots sample principal components

$$\hat{y}_{j2} = 0.49x_{j,\text{MURDER}} - 0.75x_{j,\text{RAPE}} + 0.34x_{j,\text{ROBBERY}} + 0.10x_{j,\text{ASSAULT}}$$

 $-0.08x_{j,\text{BURGLARY}} - 0.27x_{j,\text{LARCENY}} - 0.03x_{j,\text{VEHICLE}}$

by

$$\hat{y}_{j1} = -0.37x_{j,\text{MURDER}} - 0.28x_{j,\text{RAPE}} - 0.41x_{j,\text{ROBBERY}} - 0.41x_{j,\text{ASSAULT}} - 0.37x_{j,\text{BURGLARY}} - 0.37x_{j,\text{LARCENY}} - 0.40x_{j,\text{VEHICLE}}$$

for j = 1, 2, ..., 327.

From Figure 6.5.4, metropolitan statistical areas to the far left in the \hat{y}_1 direction are those places with very extreme crimes rates on one or more characteristics. Specifically, because \hat{y}_1 has all negative eigenvector components, areas with large crime rates will have sample principle components scores far to the left. Thus, St. Louis, Detroit, New Orleans, Little Rock, Anchorage, and Myrtle Beach can be put into the severe crime category based on the *general crime component* \hat{y}_1 .

Next, from Figure 6.5.4, metropolitan statistical areas in the upper region of \hat{y}_2 dimension are going to have high murder rates relative to rape rates. These areas include St. Louis, Chicago, and Baltimore (see also Figure 6.6.2 for Murder Outliers). At the same time, metropolitan statistical areas in the lower region of \hat{y}_2 are going to have high rape rates relative to murder rates. These areas include Myrtle Beach and Anchorage (see also Figure 6.6.3 for Rape Outliers). After all, \hat{y}_2 is the *heinous crime component*, which is dominated by the negative eigenvector component for rape and the positive eigenvector component for murder.

There are also cases where areas had large murder and rape rates that ended up in the center region of \hat{y}_2 . These areas include Detroit, New Orleans, and Little Rock (see Figure 6.6.2-6.6.3). Finally, areas that had smaller crimes rates would end up center around ($\hat{y}_1 = 0, \hat{y}_2 = 0$).



Figure 6.5.5: Scatterplot for $\hat{y}_3 \sim \hat{y}_1$

Figure 6.5.5 plots sample principal components

 $\hat{y}_{j3} = -0.38x_{j,\text{MURDER}} - 0.55x_{j,\text{RAPE}} - 0.13x_{j,\text{ROBBERY}} - 0.15x_{j,\text{ASSAULT}} + 0.56x_{j,\text{BURGLARY}} + 0.44x_{j,\text{LARCENY}} + 0.10x_{j,\text{VEHICLE}}$

by

$$\hat{y}_{j1} = -0.37x_{j,\text{MURDER}} - 0.28x_{j,\text{RAPE}} - 0.41x_{j,\text{ROBBERY}} - 0.41x_{j,\text{ASSAULT}} - 0.37x_{j,\text{BURGLARY}} - 0.37x_{j,\text{LARCENY}} - 0.40x_{j,\text{VEHICLE}}$$

for j = 1, 2, ..., 327.

From Figure 6.5.5, metropolitan statistical areas in the upper region of \hat{y}_3 have serious crime rates related to one or more violent crimes relative to property crimes. One the other hand, metropolitan statistical areas in the lower region of \hat{y}_3 have significant crime rates related to one or more property crimes relative to violent crimes. That is, \hat{y}_3 has negative eigenvector components for violent crime and positive eigenvector components for property crime. Specifically, for violent crime \hat{y}_3 is most weighted towards murder and rape. While, property crime is most weighted towards burglary and larceny. This is the *crime type component*.

Lake Charles has the largest value on \hat{y}_3 . It is interesting because the area only came up once in the outliers for burglary where it had the largest number of burglaries (1576.1) per 100, 000 in the nation (see Figure 6.2.10). Otherwise, Lake Charles has not shown up on one's radar.

Myrtle Beach is interesting because it has large crime rates for all characteristics except for murder. Thus, it is tough to say whether Myrtle Beach is worse with respect to violent crime or property crime based on its \hat{y}_{j3} value. In short, \hat{y}_3 has neutralized the effect for Myrtle Beach.

St. Louis, Detroit, and New Orleans have high crime rates on most of the characteristics, but violent crime is most pronounced in \hat{y}_3 . Most notably, St. Louis has the largest murder rate of 60.9, Detroit has the second highest murder rate at 38.9, and New Orleans has the third highest murder rate at 37.1. New Orleans ranks third in rape at 171.8 and Detroit ranks fourth at 147.2. St Louis leads in robbery with 473.2, Detroit takes fourth with 344, and New Orleans in sixth with 307.5. Detroit is in first for assault with 1477.8 and St. Louis is in second with 1165.6.



Figure 6.5.6: Scatterplot for $\hat{y}_3 \sim \hat{y}_2$

Figure 6.5.6 plots sample principal components

 $\hat{y}_{j3} = -0.38x_{j,\text{MURDER}} - 0.55x_{j,\text{RAPE}} - 0.13x_{j,\text{ROBBERY}} - 0.15x_{j,\text{ASSAULT}} + 0.56x_{j,\text{BURGLARY}} + 0.44x_{j,\text{LARCENY}} + 0.10x_{j,\text{VEHICLE}}$

by

$$\hat{y}_{j2} = 0.49x_{j,\text{MURDER}} - 0.75x_{j,\text{RAPE}} + 0.34x_{j,\text{ROBBERY}} + 0.10x_{j,\text{ASSAULT}}$$

$$-0.08x_{j,\text{BURGLARY}} - 0.27x_{j,\text{LARCENY}} - 0.03x_{j,\text{VEHICLE}}$$

for j = 1, 2, ..., 327.

6.6 k-Means Clustering Method

The *k*-Means clustering algorithm is used to partition a set of *n* unclassified multivariate sample observations \mathbf{x}_1 , \mathbf{x}_2 , ..., \mathbf{x}_n into *k* clusters or groups using $(p \times 1)$ $(p \times 1)$ $(p \times 1)$ $(p \times 1)$ into *k* clusters or groups using a distance metric, most commonly, Euclidean distance. Note that the number of clusters *k* must be specified in advance, which there are various numerical processes to help, analytically, specify this parameter [10, p. 532]. Because the *k*-Means algorithm, by default, uses the Euclidean distance metric it suffers from certain deficiencies based on the number of calculations it must make and the size of those calculations. Respectively, the *k*-Means algorithm runs slower and has trouble finding reasonable clusters in the same proximity when:

(1) *n* and *p* are large.

(2) The ranges of the $x_1, x_2, ..., x_p$ are large and/or when the ranges of

 x_1, x_2, \dots, x_p are largely different from each other.

One solution to solve the range dilemma is to standardize the sample and use \mathbf{z}_1 , \mathbf{z}_2 , ..., \mathbf{z}_n as the inputs into the *k*-Means algorithm. However, this solution $(p \times 1)$ $(p \times 1)$ $(p \times 1)$ as the number of characteristics *p* being large. To address this issue, one can subset *p* variables in some meaningful way and continue with the *k*-Means analysis; but it is in generally difficult to make the decision of which characteristics to keep and which to lose. However, another option exists to solve both problems simultaneously. Specifically, one can use the first two or three sample principal components from the standardized sample provided that they account for a large proportion of the variability in \mathbf{z}_1 , \mathbf{z}_2 , ..., \mathbf{z}_n .

For the US Crime 2018 data, we will use the standardized sample and the first three sample principal components derived from the standardized sample as inputs into the *k*-Means algorithm to compare. One can then see how similar or different the two inputs behave with respect to the *k*-Means cluster assignments.

6.6.1 Choosing *k*

One black-box method for choosing the appropriate *k* for several clustering methods is found in the R package NbClust. NbClust provides 30 indices for determining the relevant number of clusters and proposes to users the best clustering scheme from the different results obtained by varying all combinations of number of clusters, distance measures, and clustering methods. It can simultaneously compute all the indices and determine the number of clusters in a single function call [11].



Figure 6.6.1: NbClust, Black-Box Method, k-Means

In Table 6.6.1, the optimal number of clusters is found to be k = 3 for both inputs,

Standardized Crime 2018 and \hat{y}_1 , \hat{y}_2 , \hat{y}_3 .

6.6.2 *k*-Means, *k* = 3

6.6.2.1 *k*-Means, k = 3, Cluster Sizes

k=3, k-Means, Standardized Crime 2018, Cluster Size						
cluster	1	2	3			
size	11	116	200			
k=3, k-Means, y1, y2, y3, Cluster Size						
cluster	1	2	3			
size	6	109	212			

Table 6.6.1: k-Means, k = 3, Cluster Sizes

6.6.2.2 *k*-Means, k = 3, Differences in Cluster Assignments

k-Means, k=3, Differences in Cluster Assignments						
	y1, y2, y3					
	cluster	1	2	3		
Standardized	1	6	5	0		
Crime 2018	2	0	104	12		
	3	0	0	200		

Table 6.6.2: *k*-Means, *k* = 3, Differences in Cluster Assignments

From Table 6.6.2, one can see that 5 metropolitan statistical areas were assigned to cluster 1 using Standardized Crime 2018 and the same 5 metropolitan statistical areas where assigned to cluster 2 using \hat{y}_1 , \hat{y}_2 , \hat{y}_3 . Similarly, one can see that the same 12 metropolitan statistical areas were assigned to cluster 2 using Standardized Crime 2018 and cluster 3 using \hat{y}_1 , \hat{y}_2 , \hat{y}_3 . Table 6.6.3 presents the specific metropolitan statistical areas assigned to different clusters. These cases are usually located near the border's edges of the cluster regions.

k-Means, k=3, Differences in Assignments for Cluster 1 and 2						
Metropolitan Statistical Area	Standardized Crime 2018	y1, y2, y3				
Albuquerque	1	2				
Chicago	1	2				
Houston	1	2				
Memphis	1	2				
Nashville	1	2				
k-Means, k=3, Differences in Assignments for Cluster 2 and 3						
Metropolitan Statistical Area	Standardized Crime 2018	y1, y2, y3				
Brunswick	2	3				
Charleston	2	3				
Columbus_OH	2	3				
Dayton	2	3				
Jackson_MI	2	3				
Lexington	2	3				
Orlando	2	3				
Reno	2	3				
Saginaw	2	3				
Salem	2	3				
San_Jose	2	3				
Honolulu	2	3				

Table 6.6.3: *k*-Means, k = 3, Differences in Cluster Assignments for Clusters 1,2,3

6.6.2.3 k-Means, k = 3, Sample Cluster Mean Vectors

Table 6.6.4: *k*-Means, *k* = 3, Sample Cluster Mean Vectors

k-Means, k=3, Standardized Crime 2018, Sample Cluster Mean Vectors									
cluster	MURDER	RAPE	ROBBERY	ASSAULT	BURGLARY	LARCENY	VEHICLE		
1	22.7091	111.782	312.0364	865.3273	811.3909	3678.564	681.509		
2	6.79483	58.5103	97.13879	362.3543	616.0431	2158.441	318.863		
3	3.1705	43.2875	41.8045	183.871	310.761	1404.355	129.394		
	k-Means, k=3, y1, y2, y3, Sample Cluster Mean Vectors								
cluster	MURDER	RAPE	ROBBERY	ASSAULT	BURGLARY	LARCENY	VEHICLE		
1	29.55	153.2	316.9833	1010	925.7	4402.25	819.6		
2	7.30917	58.8835	108.5073	391.3817	640.5229	2227.902	332.594		
3	3.29293	44.0415	44.01981	186.8175	316.8269	1426.697	137.703		
Original Sample Mean Vector for Crime 2018									
	MURDER	RAPE	ROBBERY	ASSAULT	BURGLARY	LARCENY	VEHICLE		
	5.11	50.99	70.52	270.11	435.9	1748.36	215.18		

Table 6.6.4 reveals that cluster 3 has smaller sample mean components for both inputs compared to the original sample mean vector for Crime 2018. Thus, cluster 3 can be labeled the *below average crime cluster*. Cluster 2 has larger sample mean components for both inputs compared to the original sample mean vector for Crime 2018. Hence, cluster 2 can be labeled the *above average crime cluster*. Cluster 1 has much larger sample mean components for both inputs compared to their respective cluster 2's, cluster 3's, and the original sample mean vector for Crime 2018. Correspondingly, cluster 1 can be labeled the *extreme crime cluster*. At the same time, one should notice that the sample means with input $\hat{y}_1, \hat{y}_2, \hat{y}_3$ are higher than the samples means with input Standardized Crime 2018. The reason will be evident once we plot the cluster assignments on the $\hat{y}_1, \hat{y}_2, \hat{y}_3$ and the original dimensions.



6.6.2.4 *k*-Means, k = 3, Scatterplots on \hat{y}_1 , \hat{y}_2 , \hat{y}_3

k-Means, k=3, Input Standardized Crime 2018, Plotted on y1, y2, y3



be farther to the right in the \hat{y}_1 dimension. Likewise, cluster 3 the *below average crime cluster*, is farther to the right then clusters 1 and 2 given its smaller mean vector components.



k-Means, k=3, Input y1, y2, y3, Plotted on y1, y2, y3

Figure 6.6.2: *k*-Means, k = 3, Input $\hat{y}_1, \hat{y}_2, \hat{y}_3$, Plotted on $\hat{y}_1, \hat{y}_2, \hat{y}_3$

Referring to Figure 6.6.2, the clusters with input \hat{y}_1 , \hat{y}_2 , \hat{y}_3 do not look remarkably different from clusters in Figure 6.6.1, with input Standardized Crime 2018. Except that cluster 1, the *extreme crime cluster*, has lost five metropolitan statistical areas, Albuquerque, Chicago, Houston, Memphis, and Nashville which have been absorbed into cluster 2 the *above average crime cluster*. These areas have large crime rates but not as extreme as St. Louis, Detroit, New Orleans, Little Rock, Anchorage, and Myrtle Beach with respect to the point estimate \hat{y}_{j1} . That is why the cluster mean vector components are larger for the *extreme crime cluster* with input \hat{y}_1 , \hat{y}_2 , \hat{y}_3 compared to the *extreme crime cluster* with input Standardized Crime 2018. Lastly, one should mention that Albuquerque, Chicago, Houston, Memphis, and Nashville are on the boundary of clusters 1 and 2 for both inputs; consequently, being assigned to either cluster does not seem unreasonable.

6.6.2.5 *k*-Means, k = 3, Scatterplots on Original Crime 2018 Dimensions

Another method of visualizing the *k*-Means, k = 3, cluster assignments for inputs Standardized Crime 2018 and $\hat{y}_1, \hat{y}_2, \hat{y}_3$ is to plot them using a scatterplot matrix on the original Crime 2018 dimensions.



Figure 6.6.3: *k*-Means, *k* = 3, Input Standardized Crime 2018, Original Crime 2018

Looking at Figure 6.6.3, k-Means, k = 3, Input Standardized Crime 2018, one can see the densities for each cluster on each characteristic. Cluster 1's distributions are all shifted farthest to the right giving it the largest sample mean on each characteristic. Next, cluster 2 has the second largest sample means based on the position of the densities. Afterward, cluster 3 has the smallest sample means based upon the same reasoning. One can also gather the same insight by looking at the boxplots located on the right side of Figure 6.6.3. In short, these results match the graphical interpretations given in Figure 6.6.1.



Figure 6.6.4: *k*-Means, k = 3, Input $\hat{y}_1, \hat{y}_2, \hat{y}_3$, Original Crime 2018

Results from Figure 6.6.4 are analogous to results from Figure 6.6.3.

6.7 Hierarchical Clustering Methods

In a hierarchical clustering algorithm, the data are *not* partitioned into a particular number of clusters at a single step. Instead the clustering consists of a series of partitions, which may run from a single cluster containing all *n* individuals, to *n* clusters each containing a single individual. Hierarchical clustering techniques may be subdivided into *agglomerativ*e methods, which proceed by a series of successive fusions of the *n* individuals into groups, and *divisive* methods, which separate the *n* individuals successively into smaller groups [12, p. 71].

6.7.1 Agglomerate Clustering Methods

Agglomerative clustering is the most common type of hierarchical clustering used to group objects in clusters based on their similarity. It works in a "bottom-up" manner. That is, each object is initially considered as a single-element cluster (leaf). At each step of the algorithm, the two clusters that are most similar are combined into a new bigger cluster (nodes). This procedure is iterated until all points are members of just one single big cluster (root). The result is a tree-based representation of the fusion of the objects, named a dendrogram [11]. For our analysis, we will focus solely on two agglomerative clustering methods Average and Ward.

6.7.1.1 Average and Ward's Method

Average and Ward's Method can use a Euclidean Distance Matrix $\mathbf{D}_{(n \times n)}$ as an initial input into the algorithm. Then each method defines a linkage function that takes the distance information $\mathbf{D}_{(n \times n)}$ and groups pairs of objects into clusters based on some type of similarity criterion. Next, these newly formed clusters are linked to each other to make bigger clusters. This process is iterated until all the objects in the original data set are linked together into a dendrogram.

- Average Linkage Function defines similarity between two clusters as the average distance between the elements in one cluster and the elements in the other cluster.
- Ward's Linkage Function minimizes the total within-cluster variance. At each step the pair of clusters with minimum between-cluster distance are merged.
 Note that, at each stage of the clustering process the two clusters, that have the smallest linkage distance, are linked together [11].

6.7.1.1.1 Euclidean Distance Matrix $D_{Z}_{(n \times n)}$ for Standardized Sample $Z_{(n \times p)}$

$$\mathbf{D}_{\mathbf{Z}}_{(n\times n)} = \begin{bmatrix} d_{11} & d_{12} & \cdots & d_{1n} \\ d_{21} & d_{22} & \cdots & d_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ d_{p1} & d_{p2} & \cdots & d_{nn} \end{bmatrix} = \begin{bmatrix} 0 & d_{12} & \cdots & d_{1n} \\ d_{21} & 0 & \cdots & d_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ d_{p1} & d_{p2} & \cdots & 0 \end{bmatrix}$$

where

$$d_{jl} = d\left(\sum_{(p \times 1)}^{p}, \sum_{(p \times 1)}^{p}\right) = \sqrt{\sum_{k=1}^{p} (z_{jk} - z_{lk})^2}$$
$$= \sqrt{(z_{j1} - z_{l1})^2 + (z_{j2} - z_{l2})^2 + \dots + (z_{jp} - z_{lp})^2}$$

for j, l = 1, 2, ..., n.

6.7.1.1.2 Euclidean Distance Matrix $D_{\hat{Y}}_{(n \times n)}$ for Sample Principal Components $\hat{Y}_{(n \times p)}$

$$\mathbf{D}_{\hat{\mathbf{Y}}}_{(n\times n)} = \begin{bmatrix} \hat{d}_{11} & \hat{d}_{12} & \cdots & \hat{d}_{1n} \\ \hat{d}_{21} & \hat{d}_{22} & \cdots & \hat{d}_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \hat{d}_{p1} & \hat{d}_{p2} & \cdots & \hat{d}_{nn} \end{bmatrix} = \begin{bmatrix} 0 & \hat{d}_{12} & \cdots & \hat{d}_{1n} \\ \hat{d}_{21} & 0 & \cdots & \hat{d}_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \hat{d}_{p1} & \hat{d}_{p2} & \cdots & 0 \end{bmatrix}$$

where

$$\hat{d}_{jl} = d\left(\hat{\mathbf{y}}_{j}, \hat{\mathbf{y}}_{l}\right) = \sqrt{\sum_{k=1}^{p} (\hat{y}_{jk} - \hat{y}_{lk})^{2}}$$
$$= \sqrt{(\hat{y}_{j1} - \hat{y}_{l1})^{2} + (\hat{y}_{j2} - \hat{y}_{l2})^{2} + \dots + (\hat{y}_{jp} - \hat{y}_{lp})^{2}}$$

for j, l = 1, 2, ..., n.

6.7.1.1.3 Average and Ward's Clustering Pseudo-Code

- Prepare the sample data.
- Compute the Euclidean distance matrix $\mathbf{D}_{(n \times n)}$.
- Use linkage function to group objects into dendrogram based on $\mathbf{D}_{(n \times n)}$.
- * Determine where to partition the dendrogram branches, creating k clusters

[11].

6.7.2 Euclidean Distance Matrices

6.7.2.1 Euclidean Distance Matrix for Standardized US Crime 2018

Table 6.7.1: Euclidean Distance Matrix for Standardized US Crime 2018, First Five

Euclidean Distance Matrix for Standardized US Crime 2018, First Five Observations								
	Abilene	Akron	Albany_GA	Albany_NY	Albuquerque			
Abilene	0	0.8	3.1	1.9	6.1			
Akron	0.8	0	3.2	1.3	6.3			
Albany_GA	3.1	3.2	0	4.2	4.6			
Albany_NY	1.9	1.3	4.2	0	7.3			
Albuquerque	6.1	6.3	4.6	7.3	0			

Observations

6.7.2.2 Euclidean Distance Matrix for $\hat{y}_1, \hat{y}_2, \hat{y}_3$

Table 6.7.2: Euclidean Distance Matrix for $\hat{y}_1, \hat{y}_2, \hat{y}_3$, First Five Observations

Euclidean Distance Matrix for y1, y2, y3, First Five Observations								
	Abilene Akron Albany_GA Albany_NY Albuquerque							
Abilene	0	0.7	2.8	1.5	5.5			
Akron	0.7	0	2.9	1.1	5.8			
Albany_GA	2.8	2.9	0	3.8	3.4			
Albany_NY	1.5	1.1	3.8	0	6.9			
Albuquerque	5.5	5.8	3.4	6.9	0			

6.7.3 Wards Method

6.7.3.1 Choosing *k*



Figure 6.7.1: NbClust, Black-Box Method, Ward
In Figure 6.7.1, the optimal number of clusters is found to be k = 3 for both inputs,

Standardized US Crime 2018 and \hat{y}_1 , \hat{y}_2 , \hat{y}_3 . We will continue our analysis with k = 3.

6.7.3.2 Ward, k = 3

6.7.3.2.1 Ward, k = 3, Cluster Sizes

k=3, Ward, Standardized Crime 2018, Cluster Size							
cluster 1 2 3							
size	11	48	268				
k=3, Ward, y1, y2, y3, Cluster Size							
cluster	1	2	3				
size	6	72	249				

Table 6.7.3: Ward, k = 3, Cluster Sizes

Interestingly, one can see that the cluster sizes for Wards algorithm, in Table 6.7.3, match the cluster sizes in the *k*-Means algorithm, for cluster 1 (Table 6.6.1). That is, cluster 1 has 11 members for *k*-Means and Wards methods, with respect to input standardized Crime 2018. In the same way, cluster 1 has 6 members for *k*-Means and Wards methods, with respect to input $\hat{y}_1, \hat{y}_2, \hat{y}_3$.Yet, the other assignments for Wards are not the same as for *k*-Means. At first glance, it looks like Wards method produces larger cluster 3's and smaller cluster 2's then in the *k*-Means analysis.

6.7.3.2.2 Ward, k = 3, Difference in Cluster Assignments

Table 6.7.4: Ward, k = 3, Differences in Cluster Assignments

Ward, k=3, Differences in Cluster Assignments							
	y1, y2, y3						
	cluster	1	2	3			
Standardized	1	6	5	0			
Crime 2018	2	0	45	3			
	3	0	22	246			

Ward, k=3, Differences in Assignments for Cluster 1 and 2					
Metropolitan Statistical Area	Standardized Crime 2018	y1, y2, y3			
Albuquerque	1	2			
Chicago	1	2			
Houston	1	2			
Memphis	1	2			
Nashville	1	2			
Ward, k=3, Differences	in Assignments for Cluster 2	and 3			
Metropolitan Statistical Area	Standardized Crime 2018	y1, y2, y3			
Dothan	2	3			
Jackson_TN	2	3			
Lafayette_LA	2	3			
Ward, k=3, Differences	in Assignments for Cluster 3	and 2			
Metropolitan Statistical Area	Standardized Crime 2018	y1, y2, y3			
Battle_Creek	3	2			
Billings	3	2			
Chattanooga	3	2			
Cleveland	3	2			
Colorado_Springs	3	2			
Columbia_SC	3	2			
Farmington	3	2			
Fresno	3	2			
Gainesville_FL	3	2			
Gulfport	3	2			
Jackson_MI	3	2			
Medford	3	2			
Modesto	3	2			
Muskegon	3	2			
Panama_City	3	2			
Rapid_City	3	2			
Salt_Lake	3	2			
San_Francisco	3	2			
Seattle	3	2			
Stockton	3	2			
Tuscaloosa	3	2			
Warner_Robins	3	2			

Table 6.7.5: Ward, k = 3, Differences in Cluster Assignments for Clusters 1,2,3

Ward, k=3, Standardized Crime 2018, Sample Cluster Mean Vectors									
cluster	MURDER	RAPE	ROBBERY	ASSAULT	BURGLARY	LARCENY	VEHICLE		
1	22.7091	111.782	312.0364	865.3273	811.3909	3678.564	681.509		
2	8.33125	64.8854	103.6271	439.1646	783.5458	2451.292	348.371		
3	3.81493	46.0082	54.68246	215.4007	358.2201	1543.24	172.183		
	Ward, k=3, y1, y2, y3, Sample Cluster Mean Vectors								
cluster	MURDER	RAPE	ROBBERY	ASSAULT	BURGLARY	LARCENY	VEHICLE		
1	29.55	153.2	316.9833	1010	925.7	4402.25	819.6		
2	7.72639	65.3292	113.8458	434.8806	703.2458	2419.354	359.553		
3	3.76908	44.3831	52.05863	204.6365	346.7896	1490.392	158.868		
Original Sample Mean Vector for Crime 2018									
	MURDER	RAPE	ROBBERY	ASSAULT	BURGLARY	LARCENY	VEHICLE		
	5.11	50.99	70.52	270.11	435.9	1748.36	215.18		

Table 6.7.6 shows that the cluster sample means for Wards method are not very different than those of the *k*-Means (Table 6.6.4). Thus, for both inputs we can again label cluster 3 the *below average crime cluster*; cluster 2 the *above average crime cluster*; and cluster 1 the *extreme crime cluster*. Despite that similarity to the *k*-Means, there are some key differences. First, the cluster 3 sample means for input Standardized Crime 2018 and input $\hat{y}_1, \hat{y}_2, \hat{y}_3$ are larger than the cluster 3 samples means for input Standardized Crime 2018 and input $\hat{y}_1, \hat{y}_2, \hat{y}_3$ are larger than the cluster 2 sample means for input Standardized Crime 2018 and input $\hat{y}_1, \hat{y}_2, \hat{y}_3$ are larger than the cluster 2 sample means for input Standardized Crime 2018 and input $\hat{y}_1, \hat{y}_2, \hat{y}_3$ are larger than the cluster 2 sample means for input Standardized Crime 2018 and input $\hat{y}_1, \hat{y}_2, \hat{y}_3$ are larger than the cluster 2 sample means for input Standardized Crime 2018 and input $\hat{y}_1, \hat{y}_2, \hat{y}_3$ are larger than the sample means from the *k*-Means analysis (Table 6.6.4). This can be visualized later using the cluster assignments plotted on $\hat{y}_1, \hat{y}_2, \hat{y}_3$. Third and finally, the sample mean components for input $\hat{y}_1, \hat{y}_2, \hat{y}_3$, in Wards, are not systematically larger than the sample mean components for input Standardized Crime 2018; as they were with *k*-Means (Table 6.6.4).



6.7.3.2.4 Ward, k = 3, Rectangular Dendrograms

Figure 6.7.2: Ward, k = 3, Input Standardized Crime 2018, Rectangular Dendrogram In the dendrogram displayed above, Figure 6.7.2, each leaf corresponds to a metropolitan statistical area. As we move up the tree, areas that are similar to each other are combined into branches, which are themselves fused at a higher height. The height of the fusion, provided on the vertical axis, indicates the similarity/distance between the two objects/clusters. The higher the height of the fusion, the less similar the objects/clusters are [11].



Figure 6.7.3: Ward, k = 3, Input $\hat{y}_1, \hat{y}_2, \hat{y}_3$, Rectangular Dendrogram Comparing Figure 6.7.3 to Figure 6.7.4, one can visually see that for input $\hat{y}_1, \hat{y}_2, \hat{y}_3$, Wards method produces a larger cluster 2 and a smaller cluster 3.

6.7.3.2.5 Ward, k = 3, Scatterplots on $\hat{y}_1, \hat{y}_2, \hat{y}_3$



Ward, k=3, Input Standardized Crime 2018, Plotted on y1, y2, y3





Ward, k=3, Input y1, y2, y3, Plotted on y1, y2, y3

Figure 6.7.5: Ward, k = 3, Input $\hat{y}_1, \hat{y}_2, \hat{y}_3$, Plotted on $\hat{y}_1, \hat{y}_2, \hat{y}_3$

Referring to Figure 6.7.4 and Figure 6.7.5, one can see that cluster 3 for input Standardized Crime 2018 and input \hat{y}_1 , \hat{y}_2 , \hat{y}_3 have become larger compared to their *k*-Means counterparts in Figure 6.6.1 and Figure 6.6.2. Therefore, using Wards algorithm, cluster 3's centroids, on the \hat{y}_1 axis, have shifted to the left. Since \hat{y}_1 is the *general crime component*, shifting the cluster 3's to the left, causes the sample mean components in Table 6.7.6 to increase. This is because the \hat{y}_1 eigenvector components are negative; consequently, areas with larger crime rates will have more negative scores on \hat{y}_1 .

Cluster 2, for input Standardized Crime 2018 and input \hat{y}_1 , \hat{y}_2 , \hat{y}_3 have become smaller compared to their *k*-Means counterparts in Figure 6.6.1 and Figure 6.6.2. Since, cluster 2 lost metropolitan statistical areas further to the right with respect to the \hat{y}_1 dimension, the *general crime component*, the sample mean components in Table 6.7.6 have also increasing. That is, cluster two lost areas with lower crime rates to cluster 3. Hence, the sample mean components increase in the original Crime 2018 dimensions.



6.7.3.2.6 Ward, k = 3, Scatterplots on Original Crime 2018 Dimensions

Figure 6.7.6: Ward, k = 3, Input Standardized Crime 2018, Original Crime 2018 For Figure 6.7.6, once can verify that cluster 1, has the largest sample means, cluster 2, has the second largest sample means, and cluster 3, has the smallest cluster means.



Figure 6.7.7: Ward, k = 3, Input \hat{y}_1 , \hat{y}_2 , \hat{y}_3 , Original Crime 2018

For Figure 6.7.7, once can verify that cluster 1, has the largest sample means, cluster 2, has the second largest sample means, and cluster 3, has the smallest cluster means.



6.7.4.1 Choosing *k*

Figure 6.7.8: NbClust, Black-Box Method, Average

In Figure 6.7.8, the optimal number of clusters is found to be k = 3 for input Standardized US Crime 2018 and k = 2, 3 for input $\hat{y}_1, \hat{y}_2, \hat{y}_3$. We will continue our analysis with k = 3.

6.7.4.2.1 Average, k = 3, Cluster Sizes

k=3, Average, Standardized Crime 2018, Cluster Size							
cluster 1 2 3							
size	3	3	321				
k=3, Average, y1, y2, y3, Cluster Size							
cluster 1 2 3							
size	3	3	321				

One can see from Table 6.7.7 that the cluster sizes are the same for clusters 1, 2, and

3. We shall see that each cluster also contains the same metropolitan statistical

areas. Therefore, there are no differences in cluster assignments for Average, k = 3.

6.7.4.2.2 Average, k = 3, Sample Mean Vectors

Average, k=3, Standardized Crime 2018, Sample Cluster Mean Vectors									
cluster	MURDER	RAPE	ROBBERY	ASSAULT	BURGLARY	LARCENY	VEHICLE		
1	45.63333	139.9	374.9	1096.767	863.5	3190.1	870.9667		
2	13.46667	166.5	259.0667	923.2333	987.9	5614.4	768.2333		
3	4.656698	49.081	65.91745	256.2801	426.7424	1698.757	203.881		
	Ave	rage, k=3	3, y1, y2, y3	, Sample Clu	uster Mean V	ectors			
cluster	MURDER	RAPE	ROBBERY	ASSAULT	BURGLARY	LARCENY	VEHICLE		
1	45.63333	139.9	374.9	1096.767	863.5	3190.1	870.9667		
2	13.46667	166.5	259.0667	923.2333	987.9	5614.4	768.2333		
3	4.656698	49.081	65.91745	256.2801	426.7424	1698.757	203.881		
Original Sample Mean Vector for Crime 2018									
	MURDER	RAPE	ROBBERY	ASSAULT	BURGLARY	LARCENY	VEHICLE		
	5.11	50.99	70.52	270.11	435.9	1748.36	215.18		

Table 6.7.8: Average, k = 3, Sample Cluster Mean Vectors

Looking at Table 6.7.8 one can see that cluster 3's sample mean vector components, using Average method, have very similar values to the original Crime 2018 sample

mean vector components. After all, cluster 3 has 321/327 of the metropolitan statistical areas in its cluster. One could label cluster 3 as the *average crime cluster* even though it is likely composed of places with low, medium, and high crime rates. Cluster 2 and 3 are a bit harder to precisely name. It is clear that, cluster 2 and cluster 3 have larger sample mean components then cluster 1. Although, one can say, cluster 1 has the largest sample mean components on murder, robbery, assault, and vehicle theft. Whereas, cluster 2 has the largest sample mean components on rape, burglary, and larceny. It would be convenient if the clusters were split by crime type, but this is not the case.

6.7.4.2.3 Average, k = 3, Rectangular Dendrograms



Average, k=3, Input Standardized Crime 2018, Rectangular Dendrogram

Figure 6.7.9: Average, k = 3, Input Standardized Crime 2018, Rectangular

Dendrogram





Figure 6.7.10: Average, k = 3, Input \hat{y}_1 , \hat{y}_2 , \hat{y}_3 , Rectangular Dendrogram After reviewing Figure 6.7.9 and Figure 6.7.10, one can see that even though the dendrograms have the same cluster assignments for k = 3, they do *not* have identical tree structure. Undoubtedly, if one would increase k (increase the number of clusters), the cluster assignments would change for input Standardized Crime 2018 compared to input \hat{y}_1 , \hat{y}_2 , \hat{y}_3 . In Section 6.7.5, we will analytically compare all combinations of dendrograms with respect to inputs and algorithms.

6.7.4.2.4 Average, k = 3, Scatterplots on $\hat{y}_1, \hat{y}_2, \hat{y}_3$



Average, k=3, Input Standardized Crime 2018, Plotted on y1, y2, y3

Figure 6.7.11: Average, k = 3, Input Standardized Crime 2018, Plotted on $\hat{y}_1, \hat{y}_2, \hat{y}_3$

Average, k=3, Input y1, y2, y3, Plotted on y1, y2, y3



Figure 6.7.12: Average, k = 3, Input $\hat{y}_1, \hat{y}_2, \hat{y}_3$, Plotted on $\hat{y}_1, \hat{y}_2, \hat{y}_3$

As has been noted, cluster assignments for both inputs are same when k = 3. Hence, Figure 6.7.11 and Figure 6.7.12 are indistinguishable. Cluster 1 has metropolitan statistical areas St. Louis, Detroit, and New Orleans. Cluster 2 has metropolitan statistical areas Myrtle Beach, Anchorage, and Little Rock.

Looking at the left plot $\hat{y}_2 \sim \hat{y}_1$, one can see cluster 1 sits in the upper left region. In terms of \hat{y}_1 (*general crime component*), we know these areas have been classified as having extremely high crime rates. In terms of \hat{y}_2 (*heinous crime component*), we know that these areas will have higher murder rates relative to rape rapes because the component is dominated by a negative eigenvector coefficient for rape and a positive eigenvector coefficient for murder. From Figure 6.2.2 (Murder Outliers), one can see that St. Louis, Detroit, and New Orleans have the largest murder rates of the sample in descending order. One should note that in Figure 6.2.4 (Rape Outliers), New Orleans ranks third. Therefore, New Orleans is being pulled back down in the \hat{y}_2 direction. Nevertheless, we could cautiously call cluster 1, the *murder cluster*.

Continuing to look at the left plot $\hat{y}_2 \sim \hat{y}_1$, one can see cluster 2 sits in the lower left region. In terms of \hat{y}_1 , we also know these areas have been classified as having extremely high crime rates. In terms of \hat{y}_2 , we know that these areas will have higher rape rates relative to murder rapes. This is certainly true for Anchorage and Myrtle Beach because they have the highest rape rates, in descending order, according to Figure 6.2.4. Little Rock, however, has large crime

rates on murder and rape; thus, it's getting pulled up in the \hat{y}_2 direction. Regardless, one could label cluster 2, the *rape cluster*.



6.7.4.2.5 Average, k = 3, Scatterplots on Original Crime 2018 Dimensions

Figure 6.7.13: Average, k = 3, Input Standardized Crime 2018, Original Crime 2018



Figure 6.7.14: Average, k = 3, Input \hat{y}_1 , \hat{y}_2 , \hat{y}_3 , Original Crime 2018 From Figure 6.7.13 and Figure 6.7.14, once can see cluster 1 has the largest sample mean components on murder, robbery, assault, and vehicle theft. While, cluster 2 has the largest sample mean components on rape, burglary, and larceny (as seen in Table 6.7.8).

6.7.5 Comparing Ward and Average Dendrograms Using

Tanglegrams

To visually compare two dendrograms, we'll use the *tanglegram* function (in the R dendextend package), which plots two dendrograms, side by side, with their labels connected by lines. Colored lines represent common subtrees between the two dendrograms, and dashed lines represent unique branches (not common to both trees).

6.7.5.1 Ward, Input S. Crime 2018 vs. Ward, Input $\hat{y}_1, \hat{y}_2, \hat{y}_3$



Ward, Input y1, y2, y3



Figure 6.7.15: Ward, Input S. Crime 2018 vs. Ward, Input $\hat{y}_1, \hat{y}_2, \hat{y}_3$

The tanglegram for Ward input S. Crime 2018 vs. Ward, input $\hat{y}_1, \hat{y}_2, \hat{y}_3$, in Figure

6.7.15 shows a few common lower initial subtrees where all outer branches are unique. Thus, different input on same algorithm gives very unique dendrograms in this analysis.

6.7.5.2 Average, Input S. Crime 2018 vs. Average, Input $\hat{y}_1, \hat{y}_2, \hat{y}_3$

Average, Input S. Crime 2018

Average, Input y1, y2, y3



Figure 6.7.16: Average, Input S. Crime 2018 vs. Average, Input $\hat{y}_1, \hat{y}_2, \hat{y}_3$ The tanglegram for Average input S. Crime 2018 vs. Average, input $\hat{y}_1, \hat{y}_2, \hat{y}_3$, in Figure 6.7.16 shows a few more common lower initial subtrees where all outer branches are unique. Nonetheless, different input on same algorithm gives very unique dendrograms in this analysis.



6.7.5.3 Ward, Input S. Crime 2018 vs. Average, Input S. Crime 2018

Figure 6.7.17: Ward, Input S. Crime 2018 vs. Average, Input S. Crime 2018 The tanglegram for Ward input S. Crime 2018 vs. Average input S. Crime 2018 in Figure 6.7.17 shows many common lower subtrees where all outer branches are unique. In contrast from the last two tanglegrams, the same inputs on a different algorithm gives very similar dendrograms.

6.7.5.4 Ward, Input $\hat{y}_1, \hat{y}_2, \hat{y}_3$ vs. Average, Input $\hat{y}_1, \hat{y}_2, \hat{y}_3$



Figure 6.7.18: Ward, Input $\hat{y}_1, \hat{y}_2, \hat{y}_3$ vs. Average, Input $\hat{y}_1, \hat{y}_2, \hat{y}_3$

The tanglegram for Ward input \hat{y}_1 , \hat{y}_2 , \hat{y}_3 vs. Average, input \hat{y}_1 , \hat{y}_2 , \hat{y}_3 in Figure 6.7.18 shows many common lower subtrees where all outer branches are unique. As one has noted in the previous tanglegram, the same inputs on a different algorithm gives very similar dendrograms.

6.8 Comparison of k-Means, Ward, and Average

We will take a last look at the cluster assignments for *k*-Means, Ward, and Average methods on \hat{y}_1 , \hat{y}_2 and compare their respective cluster sizes.

6.8.1 *k*-Means, Ward, and Average, k = 3, Scatterplots on

$\widehat{y}_1, \widehat{y}_2$ and Cluster Sizes



Figure 6.8.1: *k*-Means, Ward, and Average, k = 3, Scatterplots on \hat{y}_1, \hat{y}_2

k=3, k-Means, Standardized Crime 2018, Cluster Size				k=3, k-Means, y1, y2, y3, Cluster Size				
cluster	1	2	3	cluster	3			
size	11	116	200	size	6	109	212	
k=3, Ward, S	k=3, Ward, Standardized Crime 2018, Cluster Size				k=3, Ward, y1, y2, y3, Cluster Size			
cluster	1	2	3	cluster	1	2	3	
size	11	48	268	size	6	72	249	
k=3, Average, Standardized Crime 2018, Cluster Size				k=3, A	verage, y1, y	2, y3, Cluster	r Size	
cluster	1	2	3	cluster	1	2	3	
size	3	3	321	size	3	3	321	

Table 6.8.1: k-Means, Ward, and Average, k = 3, Cluster Sizes

From Figure 6.8.1 and Table 6.8.1, one can see a few general patterns. First, the 6-11 highest crime metropolitan statistical areas are generally in the same cluster, far to the left in the \hat{y}_1 direction. With exception of the Average algorithm where the top 6 areas are split by dimension \hat{y}_2 (and \hat{y}_3 for that matter). That is, cluster 1, is in the upper-left region of $\hat{y}_2 \sim \hat{y}_1$ and cluster 2 is in the lower-left region of $\hat{y}_2 \sim \hat{y}_1$. Next, comparing *k*-Means and Ward, *k*-Means cluster sizes for cluster 2 are larger than Ward cluster sizes for cluster 2. Conversely, *k*-Means cluster sizes for cluster 3 are smaller than Ward cluster sizes for cluster 3. Last, *k*-Means and Ward are similar insofar as, for input Standardized Crime 2018, they include Albuquerque, Chicago, Houston, Memphis, and Nashville into cluster 1. Further, *k*-Means and Ward, include Chicago, Houston, Memphis, and Nashville into cluster 2 for input $\hat{y}_1, \hat{y}_2, \hat{y}_3$.

Chapter 7

Conclusion and Future Study

There were several interesting findings when conducting our research on the US Crime 2018 data.

Firstly, many of the extreme univariate outliers also stood out in the scatterplots of the sample principal components $\hat{y}_2 \sim \hat{y}_1$ and $\hat{y}_3 \sim \hat{y}_1$. Next, \hat{y}_1 , the *general crime component* was a good point estimator for the overall crime in an area because it accounted for 63% of the total variability in the Standardized Crime 2018 data and the eigenvector coefficients had approximately equal magnitude with all negative coefficients. Thus, metropolitan statistical areas with larger crime rates generally were farther to the left in the \hat{y}_1 dimension.

Then, we observed *k*-Means and Ward algorithms clustered areas with extreme crime together, above average crime together, and below average crime together. When viewing these assignments on the sample principal components and the original Crime 2018 dimensions, we also noticed that the 2-d scatters where most dense for the below average crime cluster, less dense for the above average crime cluster, and sparse for the extreme clime cluster. This intuitively makes sense because the univariate crime variables are right skewed, so in 2-d, clusters become less dense as crime increases.

Following this, it was clear when comparing dendrograms for Average and Wards methods, using the same inputs gave remarkably similar tree structures. Meanwhile, when using different inputs on the same algorithm, either Average or Ward, the tree structures were vastly different. This was not expected. Although, one should remember that the input Standardized Crime 2018 was 7 dimensions and the input $\hat{y}_1, \hat{y}_2, \hat{y}_3$ was only 3. As a result, we expect that the general tree structures for agglomerative methods, are more sensitive to dimensionality differences in the distance calculations then in differences in the link function criterions.

Largely, this research uncovered metropolitan statistical areas with extreme crime rates on one or more variables using a combination of univariate and bivariate analysis, principal components, and clustering. However, what this paper did not do, was attempt to try to explain the underlying reasons behind these crime intensities. This is a more nuanced question which necessitates qualitative research along with quantitative research. One would need to conduct interviews with local officials, experts in the area, and people in the community. Also, one would need to research newspaper archives, laws, and get a feel for the culture. Therefore, my future research may be to choose a single metropolitan statistical area and focus on one aspect of crime such as looking at why St. Louis has the highest murder rate in the country or why Myrtle Beach or Anchorage have the highest number of rapes per 100, 000 residents. Finally, Table 7.1 provides all metropolitan statistical areas that could be of interest for future study that have very high or extremely high crime rates on multiples variables. The 1st, 2nd, and 3rd highest crime rates are highlighted below.

Metropolitan Statistical Areas of Interest for Future Study										
1st Highest Crime Rate, 2nd Highest Crime Rate, 3rd Highest Crime Rate										
METRO	MURDER	RAPE	ROBBERY	ASSAULT	BURGLARY	LARCENY	VEHICLE			
Albuquerque	9.5	70	238.2	766.9	869.9	2838.9	817.8			
Anchorage	8.4	200.1	235.2	819.9	703.4	3342.5	970.9			
Baltimore	13.3	38.3	258.4	410.8	399.8	1804.4	266.3			
Chicago	20.7	66.1	356.1	563.1	429.8	2379.2	372.6			
Detroit	38.9	147.2	344	1477.8	1108.3	2235	961.5			
Houston	11.8	53.8	373.6	587	696	2804.6	509.6			
Lake_Charles	5.8	63.7	85.8	392.5	1576.1	2852.1	348.4			
Little_Rock	20.1	109.4	159.1	1130.5	1043.2	4942.6	562			
Memphis	17.2	50.6	254.4	820.3	847.1	2994	430			
Myrtle_Beach	11.9	190	382.9	819.3	1217.1	8558.1	771.8			
Nashville	13.3	69.9	308.2	721.3	528.3	3034	449			
New_Orleans	37.1	171.8	307.5	646.9	511.4	3290.3	755.3			
St_Louis	60.9	100.7	473.2	1165.6	970.8	4045	896.1			

Table 7.1: Metropolitan Statistical Areas of Interest for Future Study

References

- B. Everitt, An R and S-Plus Companion to Multivariate Analysis, London: Springer Science+Business Media, 2005.
- U. S. D. o. J. F. B. o. I. Justice, "Crime in the United States, 2018," [Online].
 Available: https://ucr.fbi.gov/crime-in-the-u.s/2018/crime-in-the-u.s. 2018/topic-pages/about-cius. [Accessed 18 03 2020].
- [3] R. A. Johnson and D. W. Wichern, Applied Multivariate Statistical Analysis, Sixth ed., Upper Saddle River, New Jersey: Pearson, 2019.
- [4] O. Bretscher, Linear Algebra with Applications, 5th ed., Upper Saddle River: Pearson, 2013.
- [5] D. A. Harville, Matrix Algebra From A Statistician's Perspective, New York: Springer Science+Business Media, LLC, 2008.
- [6] M. C. Hogg, Introduction to Mathematical Statistics, 8th ed., Boston: Pearson, 2019.
- [7] Wikipedia, "Definiteness_of_a_matrix," 17 3 2020. [Online]. Available: https://en.wikipedia.org/wiki/Definiteness_of_a_matrix. [Accessed 25 3 2020].
- [8] Wikipedia, "Log-normal distribution," [Online]. Available: https://en.wikipedia.org/wiki/Log-normal_distribution. [Accessed 3 8 2020].

- K. Nosakhere, "Reasons for the high rate of rape in Alaska," Anchorage Press, 2018 5 2018. [Online]. Available: https://www.anchoragepress.com/columnists/reasons-for-the-high-rate-of-rape-in-alaska/article_564f37fa-4e4f-11e8-b893-47cba59bd374.html.
 [Accessed 9 3 2020].
- [10] W. F. C. A. C. Rencher, Methods of Multivariate Analysis, 3rd ed., Hoboken, NJ: John Wiley & Sons, Inc., 2012.
- [11] A. Kassamdara, Practical Guide To Cluster Analysis in R: Unsupervised Machine Learning, 2017.
- [12] S. L. M. L. D. S. B. S. Everitt, Cluster Analysis, 5th ed., London: John Wiley & Sons, 2011.

Appendix

```
3
4 # US CRIME 2018 Metropolitan Statistical Area
5
6 # Thesis
7
10
11 # US Crime 2018 features
12
13 # univariate distribution analysis
14
15 # descriptives
16
18
19 #install.packages("psych", dependencies = TRUE)
20 library("psych")
21
22 describe(CRIME_2018_FEAT)
23
24 summary(CRIME_2018_FEAT)
25
```

```
29 # US Crime 2018 features
30
31 # univariate distribution analysis
32
33 # Murder
34
35 # density, histogram, boxplot, outliers, lower 2.5%
36 # (percentile) crime, qqplots, and shapiro-wilk tests
37
39
40 #install.packages("lattice", dependencies = TRUE)
41 library("lattice")
42
43 #install.packages("gridExtra", dependencies = TRUE)
44 library("gridExtra")
45
46 #install.packages("goft", dependencies = TRUE)
47 library("goft")
48
49 #install.packages("ggplot2", dependencies = TRUE)
50 library("ggplot2")
51
52 #install.packages("magrittr", dependencies = TRUE)
53 library("magrittr")
54
55 #install.packages("ggpubr", dependencies = TRUE)
56 library("ggpubr")
57
60 # Murder
61
62 #install.packages("lattice", dependencies = TRUE)
63 #library("lattice")
64
65 # Density
66
67 MURDER_DENSITY <- densityplot(~MURDER, data = US_CRIME_2018,
                                main="Murder Density Plot",
col = "#00c9f7")
68
69
70
71 # Histogram
72
73 MURDER_HISTOGRAM <- histogram(x=~MURDER,data=US_CRIME_2018,
74
                                type="density",
75
                                main="Murder Histogram",
76
                                col = "#00c9f7",
                                nint = 50)
77
78
79 # test distribution is normal
80
81 shapiro.test(x=US_CRIME_2018$MURDER)
```

```
83 # Normal QQ-Plot
 84
 85 MURDER_QQ_QNORM <- qqmath(x=~MURDER, data = US_CRIME_2018,</pre>
                              distribution = qnorm,
 86
 87
                              prepanel = prepanel.qqmathline,
                              panel = function(x, ...) {
 88 -
                                panel.qqmathline(x, ...)
 89
 90
                                panel.qqmath(x, ...)
 91
                              }.
                              main = "Murder Normal QQ-Plot \n SW-Test p-value = 0",
col = "#00c9f7")
 92
 93
 94
 95 # test distribution is log normal
 96
 97 # 0 in distribution (cannot compute statistic)
 98
 99 #install.packages("goft", dependencies = TRUE)
100 #library("goft")
101
102 Inorm_test(x=US_CRIME_2018$MURDER)
103
104 # Log-Normal QQ-Plot
105
106 MURDER_QQ_QLNORM <- qqmath(x=~MURDER, data = US_CRIME_2018,
                               distribution = qlnorm,
107
108
                               prepanel = prepanel.ggmathline,
109 -
                               panel = function(x, ...) {
110
                                 panel.qqmathline(x, ...)
111
                                 panel.qqmath(x, ...)
112
                               },
113
                               main = "Murder Lognormal QQ-Plot \ N  SW-Test p-value = NA",
                               col = "#00c9f7")
114
128
129 # MURDER
130
131 # find outliers based on boxplot
132
133 OutVals_murder <- boxplot(US_CRIME_2018$MURDER)$out</pre>
134
135 which(US_CRIME_2018$MURDER %in% OutVals_murder)
136
137 US_CRIME_2018[c(20, 23, 58, 76, 81, 99, 169, 185, 206, 209, 224, 227, 273, 286),]
138
139 sub_murder_outlier <- as.data.frame(US_CRIME_2018[c(20, 23, 58, 76, 81, 99, 169,
140
                                                    185, 206, 209, 224, 227, 273, 286),])
141
142 sub_murder_outlier_order <- order(sub_murder_outlier$MURDER,decreasing = TRUE)
143
144 sub_murder_outlier[sub_murder_outlier_order,]
145
146 OUTLIER_MURDER <- as.data.frame(sub_murder_outlier[sub_murder_outlier_order,])
```

```
148 # barplot of outliers for Murder
149
150 #install.packages("ggplot2", dependencies = TRUE)
151 # library("ggplot2")
152 #install.packages("ggpubr", dependencies = TRUE)
153 # library("ggpubr")
154
155 OUTLIER_MURDER_BAR <- ggplot(OUTLIER_MURDER, aes(x = reorder(CITY,-MURDER),
                                                   y = MURDER)) +
156
       geom_bar(fill = "#00c9f7", stat = "identity") +
157
158
      geom_text(aes(label = MURDER), vjust = -0.3) +
159
      theme_pubclean() +
      ggtitle("Murder Outliers of Sample") +
160
      xlab("METRO") + ylab("MURDER per 100,000") +
161
      ggpubr::rotate_x_text() +
162
163
      theme(plot.title = element_text(hjust = 0.5))
164
166
167 # Murder 2.5th percentile
168
169 guantile(US_CRIME_2018$MURDER, .025)
170
172
173 # ordered at or below 2.5th percentile metro statistical
174 # areas in terms of murder per 100,000
175
176 US_CRIME_2018[US_CRIME_2018$MURDER <= .615,]
177
178 sub_by_murder_2_5 <- as.data.frame(US_CRIME_2018[US_CRIME_2018$MURDER <= .615,])
179
180 sub_by_murder_2_5_order <- order(sub_by_murder_2_5$MURDER,decreasing = FALSE)</pre>
181
182 sub_by_murder_2_5[sub_by_murder_2_5_order,]
183
184 BOTTOM_MURDER <- as.data.frame(sub_by_murder_2_5[sub_by_murder_2_5_order,])
```

```
186 #install.packages("ggplot2", dependencies = TRUE)
187 #library("ggplot2")
188 #install.packages("ggpubr", dependencies = TRUE)
189 #library("ggpubr")
190
191 # barplot Cities with Murder At or Below 2.5th Percentile in ascending order
192
\texttt{BOTTOM\_MURDER\_BAR} <- \texttt{ggplot(BOTTOM\_MURDER}, \texttt{aes}(x = \texttt{reorder(CITY,MURDER}, y = \texttt{MURDER}))
      geom_bar(fill = "#00c9f7", stat = "identity") +
194
      geom_text(aes(label = MURDER), vjust = -0.3) +
195
196
      theme_pubclean() +
     ggtitle("Murder Lower 2.5% of Sample") +
xlab("METRO") + ylab("MURDER per 100,000") +
197
198
199
      ggpubr::rotate_x_text() +
200
      theme(plot.title = element_text(hjust = 0.5))
201
203
204 # Murder barplot with lower 2.5% and outliers
205
206 #install.packages("magrittr", dependencies = TRUE)
207 #library("magrittr")
208 #install.packages("ggpubr", dependencies = TRUE)
209 #library("ggpubr")
210
211 MURDER_BAR <- ggarrange(BOTTOM_MURDER_BAR,
212
                          OUTLIER_MURDER_BAR,
213
                          ncol = 2,
214
                          nrow = 1)
215 MURDER_BAR
219
220 # US Crime 2018 features
221
222 # univariate distribution analysis
223
224 # Rape
225
226 # density, histogram, boxplot, outliers, lower 2.5%
227 # (percentile) crime, qqplots, and shapiro-wilk tests
228
230
231 #install.packages("lattice", dependencies = TRUE)
232 #library("lattice")
233
234 #install.packages("gridExtra", dependencies = TRUE)
235 #library("gridExtra")
236
237 #install.packages("goft", dependencies = TRUE)
238 #library("goft")
239
240 #install.packages("ggplot2", dependencies = TRUE)
241 #library("ggplot2")
242
243 #install.packages("magrittr", dependencies = TRUE)
244 #library("magrittr")
245
246 #install.packages("ggpubr", dependencies = TRUE)
247 #library("ggpubr")
248
```

```
251 # Rape
252
253 #install.packages("lattice", dependencies = TRUE)
254 #library("lattice")
255
256 # Density
257
258 RAPE_DENSITY <- densityplot(~RAPE, data = US_CRIME_2018,</pre>
259
                                  main="Rape Density Plot",
                                  col = "#b88cd1")
260
261
262 # Histogram
263
264 RAPE_HISTOGRAM <- histogram(x=~RAPE, data=US_CRIME_2018,
265
                                  type="density",
                                  main="Rape Histogram",
col = "#b88cd1",
266
267
268
                                  nint = 50)
269
270 # test distribution is normal
271
272 shapiro.test(x=US_CRIME_2018$RAPE)
274 # Normal QQ-Plot
275
276 RAPE_QQ_QNORM <- ggmath(x=~RAPE, data = US_CRIME_2018,
277
                             distribution = qnorm,
278
                             prepanel = prepanel.ggmathline,
                             panel = function(x, ...) {
279 -
                               panel.qqmathline(x, ...)
280
281
                               panel.qqmath(x, ...)
282
                             },
                             main = "Rape Normal QQ-Plot \n SW-Test p-value = 0",
283
                             col = "#b88cd1")
284
285
286 # test distribution is log normal
287
288 #install.packages("goft", dependencies = TRUE)
289 #library("goft")
290
291 Inorm_test(x=US_CRIME_2018$RAPE)
292
293 # Log-Normal QQ-Plot
294
295 RAPE_QQ_QLNORM <- qqmath(x=~RAPE, data = US_CRIME_2018,
296
                              distribution = qlnorm,
297
                              prepanel = prepanel.qqmathline,
                              panel = function(x, ...) {
298 -
299
                                panel.qqmathline(x, ...)
300
                                panel.qqmath(x, ...)
301
                              },
302
                              main = "Rape Lognormal QQ-Plot \n SW-Test p-value = 0.164",
                              col = "#b88cd1")
303
```

```
305 # combine 4 plots rape
306
307 #install.packages("gridExtra", dependencies = TRUE)
308 # library("gridExtra")
309
310 grid.arrange(RAPE_DENSITY,
311
                   RAPE_HISTOGRAM,
                   RAPE_QQ_QNORM,
312
313
                   RAPE_QQ_QLNORM,
314
                   nco1=2)
315
317
318 # RAPE
319
320 # find outliers based on boxplot
321
322 OutVals_rape <- boxplot(US_CRIME_2018$RAPE)$out</pre>
323
324 which(US_CRIME_2018$RAPE %in% OutVals_rape)
325
326 US_CRIME_2018[c(10, 24, 76, 81, 95, 134, 169, 203, 209, 238, 241, 285),]
327
328 sub_rape_outlier <- as.data.frame(US_CRIME_2018[c(10, 24, 76, 81, 95,
329 134, 169, 203, 209,
134, 169, 203, 209,
330
                                                           238, 241, 285),])
331
332 sub_rape_outlier_order <- order(sub_rape_outlier$RAPE,decreasing = TRUE)</pre>
333
334 sub_rape_outlier[sub_rape_outlier_order,]
335
336 OUTLIER_RAPE <- as.data.frame(sub_rape_outlier[sub_rape_outlier_order,])
338 # barplot of outliers for Rape
339
340 #install.packages("ggplot2", dependencies = TRUE)
341 # library("ggplot2")
342 #install.packages("ggpubr", dependencies = TRUE)
343 # library("ggpubr")
344
345 OUTLIER_RAPE_BAR <- ggplot(OUTLIER_RAPE, aes(x = reorder(CITY,-RAPE), y = RAPE)) +
346 geom_bar(fill = "#b88cd1", stat = "identity") +
        geom_text(aes(label = RAPE), vjust = -0.3) +
347
348
       theme_pubclean() +
349
        ggtitle("Rape Outliers of Sample") +
350
       xlab("METRO") + ylab("RAPE per 100,000") +
        ggpubr::rotate_x_text() +
351
352
       theme(plot.title = element_text(hjust = 0.5))
353
354 - ################
355
356 # Rape 2.5th percentile
357
358 quantile(US_CRIME_2018$RAPE, .025)
359
360 - #################
```

```
362 # at or below 2.5th percentile metro statistical areas in terms of rape per 100,000
363
364 US_CRIME_2018[US_CRIME_2018$RAPE <= 19.375,]
365
366 sub_by_rape_2_5 <- as.data.frame(US_CRIME_2018[US_CRIME_2018$RAPE <= 19.375,])
367
368 sub_by_rape_2_5_order <- order(sub_by_rape_2_5$RAPE,decreasing = FALSE)
369
370 sub_by_rape_2_5[sub_by_rape_2_5_order,]
371
372 BOTTOM_RAPE <- as.data.frame(sub_by_rape_2_5[sub_by_rape_2_5_order,])</pre>
373
374 #install.packages("ggplot2", dependencies = TRUE)
375 #library("ggplot2")
376 #install.packages("ggpubr", dependencies = TRUE)
377 #library("ggpubr")
378
379 # barplot Cities with RAPE At or Below 2.5th Percentile in ascending order
380
381 BOTTOM_RAPE_BAR <- ggplot(BOTTOM_RAPE , aes(x = reorder(CITY,RAPE), y = RAPE)) +</pre>
     geom_bar(fill = "#b88cd1", stat = "identity") +
382
383
      geom_text(aes(label = RAPE), vjust = -0.3) +
384
      theme_pubclean() +
385
      ggtitle("Rape Lower 2.5% of Sample") +
386
     xlab("METRO") + ylab("RAPE per 100,000") +
387
      ggpubr::rotate_x_text() +
388
      theme(plot.title = element_text(hjust = 0.5))
389
392 # Rape barplot with lower 2.5% and outliers
393
394 #install.packages("magrittr", dependencies = TRUE)
395 #library("magrittr")
396 #install.packages("ggpubr", dependencies = TRUE)
397 #library("ggpubr")
398
399 RAPE_BAR <- ggarrange(BOTTOM_RAPE_BAR,
400
                         OUTLIER_RAPE_BAR,
401
                         ncol = 2,
402
                         nrow = 1)
403 RAPE_BAR
404
```
```
408 # US Crime 2018 features
409
410 # univariate distribution analysis
411
412 # Robbery
413
414 # density, histogram, boxplot, outliers, lower 2.5%
415 # (percentile) crime, qqplots, and shapiro-wilk tests
416
418
419 #install.packages("lattice", dependencies = TRUE)
420 #library("lattice")
421
422 #install.packages("gridExtra", dependencies = TRUE)
423 #library("gridExtra")
424
425 #install.packages("goft", dependencies = TRUE)
426 #library("goft")
427
428 #install.packages("ggplot2", dependencies = TRUE)
429 #library("ggplot2")
430
431 #install.packages("magrittr", dependencies = TRUE)
432 #library("magrittr")
433
434 #install.packages("ggpubr", dependencies = TRUE)
435 #library("ggpubr")
436
439 # ROBBERY
440
441 #install.packages("lattice", dependencies = TRUE)
442 #library("lattice")
443
444 # Density
445
446 ROBBERY_DENSITY <- densityplot(~ROBBERY, data = US_CRIME_2018,
447
                                      main="Robbery Density Plot",
448
                                      col = "#7449f7")
449
450 # Histogram
451
452 ROBBERY_HISTOGRAM <- histogram(x=~ROBBERY,data=US_CRIME_2018,
453
                                      type="density"
454
                                      main="Robbery Histogram",
455
                                      col = "#7449f7",
456
                                      nint = 50)
457
458 # test distribution is normal
459
460 shapiro.test(x=US_CRIME_2018$ROBBERY)
```

```
462 # Normal QQ-Plot
463
464
    ROBBERY_QQ_QNORM <- qqmath(x=~ROBBERY, data = US_CRIME_2018,
465
                              distribution = qnorm,
466
                              prepanel = prepanel.qqmathline,
467 -
                              panel = function(x, ...) {
                               panel.qqmathline(x, ...)
468
469
                               panel.qqmath(x, ...)
470
                              },
                              main = "Robbery Normal QQ-Plot \n SW-Test p-value = 0",
471
                              col = "#7449f7")
472
473
474 # test distribution is log normal
475
476 #install.packages("goft", dependencies = TRUE)
477 #library("goft")
478
479 lnorm_test(x=US_CRIME_2018$ROBBERY)
480
481 # Log-Normal QQ-Plot
482
483 ROBBERY_QQ_QLNORM <- qqmath(x=~ROBBERY, data = US_CRIME_2018,
484
                               distribution = qlnorm,
485
                               prepanel = prepanel.qqmathline,
486 -
                               panel = function(x, ...) {
487
                                panel.qqmathline(x, ...)
488
                                panel.qqmath(x, ...)
489
                               },
490
                               main = "Robbery Lognormal QQ-Plot \n SW-Test p-value = 0",
                               col = "#7449f7")
491
493 # combine 4 plots robbery
494
495 #install.packages("gridExtra", dependencies = TRUE)
496 # library("gridExtra")
497
498 grid.arrange(ROBBERY_DENSITY,
499
                ROBBERY_HISTOGRAM,
                ROBBERY_QQ_QNORM,
500
501
                ROBBERY_QQ_QLNORM,
502
                nco1=2)
503
505
506 # ROBBERY
507
508 # find outliers based on boxplot
509
510 OutVals_robbery <- boxplot(US_CRIME_2018$ROBBERY)$out
511
512 which(US_CRIME_2018$ROBBERY %in% OutVals_robbery)
513
514 US_CRIME_2018[c(5, 10, 20, 58, 81, 130, 185, 203, 206, 209, 224, 259, 286, 287),]
515
209, 224, 259, 286, 287),])
518
519
520 sub_robbery_outlier_order <- order(sub_robbery_outlier$ROBBERY,decreasing = TRUE)
521
522 sub_robbery_outlier[sub_robbery_outlier_order,]
```

```
524 OUTLIER_ROBBERY <- as.data.frame(sub_robbery_outlier[sub_robbery_outlier_order,])
525
526 # barplot of outliers for Robbery
527
528 #install.packages("ggplot2", dependencies = TRUE)
529 # library("ggplot2")
530 #install.packages("ggpubr", dependencies = TRUE)
531 # library("ggpubr")
532
533 OUTLIER_ROBBERY_BAR <- qgplot(OUTLIER_ROBBERY, aes(x = reorder(CITY, -ROBBERY)),
534
                                                         y = ROBBERY)) +
       geom_bar(fill = "#7449f7", stat = "identity") +
535
       geom_text(aes(label = ROBBERY), vjust = -0.3) +
536
537
       theme_pubclean() +
      ggtitle("Robbery Outliers of Sample") +
538
      xlab("METRO") + ylab("ROBBERY per 100,000") +
539
540
       ggpubr::rotate_x_text() +
541
      theme(plot.title = element_text(hjust = 0.5))
542
543 - ################
544
545 # Robbery 2.5th percentile
546
547 quantile(US_CRIME_2018$ROBBERY, .025)
548
549 - #################
551 # at or below 2.5th percentile metro statistical areas in terms of robbery per 100,000
552
553 US_CRIME_2018[US_CRIME_2018$ROBBERY <= 9.73,]
554
555 sub_by_robbery_2_5 <- as.data.frame(US_CRIME_2018[US_CRIME_2018$ROBBERY <= 9.73,])</pre>
556
557 sub_by_robbery_2_5_order <- order(sub_by_robbery_2_5$ROBBERY, decreasing = FALSE)
558
559 sub_by_robbery_2_5[sub_by_robbery_2_5_order,]
560
561 BOTTOM_ROBBERY <- as.data.frame(sub_by_robbery_2_5[sub_by_robbery_2_5_order,])
562
563 #install.packages("ggplot2", dependencies = TRUE)
564 #library("ggplot2")
565 #install.packages("ggpubr", dependencies = TRUE)
566 #library("ggpubr")
567
568 # barplot Cities with ROBBERY At or Below 2.5th Percentile in ascending order
569
570 BOTTOM_ROBBERY_BAR <- ggplot(BOTTOM_ROBBERY , aes(x = reorder(CITY,ROBBERY),
                                                       y = ROBBERY)) +
571
      geom_bar(fill = "#7449f7", stat = "identity") +
572
573
       geom_text(aes(label = ROBBERY), vjust = -0.3) +
574
      theme_pubclean() +
      ggtitle("Robbery Lower 2.5% of Sample") +
575
      xlab("METRO") + ylab("ROBBERY per 100,000") +
576
577
      ggpubr::rotate_x_text() +
578
       theme(plot.title = element_text(hjust = 0.5))
579
```

```
582 # Robbery barplot with lower 2.5% and outliers
583
584 #install.packages("magrittr", dependencies = TRUE)
585 #library("magrittr")
#install.packages("ggpubr", dependencies = TRUE)
#library("ggpubr")
588
589 ROBBERY_BAR <- ggarrange(BOTTOM_ROBBERY_BAR,
590
                             OUTLIER_ROBBERY_BAR,
591
                             ncol = 2,
592
                             nrow = 1)
593 ROBBERY_BAR
597
598 # US Crime 2018 features
599
600 # univariate distribution analysis
601
602 # Assault
603
604 # density, histogram, boxplot, outliers, lower 2.5%
605 # (percentile) crime, qqplots, and shapiro-wilk tests
606
608
609 #install.packages("lattice", dependencies = TRUE)
610 #library("lattice")
611
612 #install.packages("gridExtra", dependencies = TRUE)
613 #library("gridExtra")
614
615 #install.packages("goft", dependencies = TRUE)
616 #library("goft")
617
618 #install.packages("ggplot2", dependencies = TRUE)
619 #library("ggplot2")
620
621 #install.packages("magrittr", dependencies = TRUE)
622 #library("magrittr")
623
624 #install.packages("ggpubr", dependencies = TRUE)
625 #library("ggpubr")
626
```

```
628
629 # Assault
630
631 #install.packages("lattice", dependencies = TRUE)
632 #library("lattice")
633
634 # Density
635
636 ASSAULT_DENSITY <- densityplot(~ASSAULT, data = US_CRIME_2018,
637
                                     main="Assault Density Plot",
                                     col = "#ca3886")
638
639
640 # Histogram
641
642 ASSAULT_HISTOGRAM <- histogram(x=~ASSAULT,data=US_CRIME_2018,
643
                                     type="density",
                                     main="Assault Histogram",
col = "#ca3886",
644
645
646
                                     nint = 50)
647
648 # test distribution is normal
649
650 shapiro.test(x=US_CRIME_2018$ASSAULT)
652 # Normal QQ-Plot
653
654 ASSAULT_QQ_QNORM <- qqmath(x=~ASSAULT, data = US_CRIME_2018,
655
                                distribution = qnorm,
656
                                prepanel = prepanel.qqmathline,
657 -
                                panel = function(x, ...) {
658
                                  panel.qqmathline(x, ...)
659
                                  panel.qqmath(x, ...)
660
                                3.
                                main = "Assault Normal QQ-Plot \n SW-Test p-value = 0",
col = "#ca3886")
661
662
663
664 # test distribution is log normal
665
666 #install.packages("goft", dependencies = TRUE)
667 #library("goft")
668
669 lnorm_test(x=US_CRIME_2018$ASSAULT)
670
671 # Log-Normal QQ-Plot
672
ASSAULT_QQ_QLNORM <- qqmath(x=~ASSAULT, data = US_CRIME_2018,
674
                                 distribution = qlnorm,
675
                                 prepanel = prepanel.qqmathline,
                                 panel = function(x, ...) {
676 -
                                   panel.qqmathline(x, ...)
677
678
                                   panel.qqmath(x, ...)
679
                                 3.
                                 main = "Assault Log-Normal QQ-Plot \n SW-Test p-value = 0.3352",
col = "#ca3886")
680
681
```

```
683 # combine 4 plots robbery
684
685
   #install.packages("gridExtra", dependencies = TRUE)
686 # library("gridExtra")
687
688 grid.arrange(ASSAULT_DENSITY,
689
                 ASSAULT_HISTOGRAM,
690
                 ASSAULT_QQ_QNORM,
691
                 ASSAULT_QQ_QLNORM,
692
                 ncol=2
693
695
696 # ASSAULT
697
698 # find outliers based on boxplot
699
700 OutVals_assault <- boxplot(US_CRIME_2018$ASSAULT)$out
701
702 which(US_CRIME_2018$ASSAULT %in% OutVals_assault)
703
704 US_CRIME_2018[c(3, 5, 6, 10, 12, 76, 81, 95, 120, 130, 169, 174, 185, 203, 206, 209, 214, 227, 286, 318),]
706
707 sub_assault_outlier <- as.data.frame(US_CRIME_2018[c(3, 5, 6, 10, 12, 76, 81,
708
                                                        95, 120, 130, 169, 174, 185,
                                                        203, 206, 209, 214, 227, 286, 318),])
709
710
711
712 sub_assault_outlier_order <- order(sub_assault_outlier$ASSAULT,decreasing = TRUE)
714 sub_assault_outlier[sub_assault_outlier_order,]
715
716 OUTLIER_ASSAULT <- as.data.frame(sub_assault_outlier[sub_assault_outlier_order,])
717
718 # barplot of outliers for Assault
719
720 #install.packages("ggplot2", dependencies = TRUE)
721 # library("ggplot2")
722 #install.packages("ggpubr", dependencies = TRUE)
723 # library("ggpubr")
724
725 OUTLIER_ASSAULT_BAR <- ggplot(OUTLIER_ASSAULT, aes(x = reorder(CITY,-ASSAULT),
726
                                                       y = ASSAULT) +
727
       geom_bar(fill = "#ca3886", stat = "identity") +
728
      geom_text(aes(label = ASSAULT), vjust = -0.3) +
729
       theme_pubclean() +
730
      ggtitle("Assault Outliers of Sample") +
731
      xlab("METRO") + ylab("ASSAULT per 100,000") +
732
       ggpubr::rotate_x_text() +
733
       theme(plot.title = element_text(hjust = 0.5))
734
735
737
738 # Assault 2.5th percentile
739
740 quantile(US_CRIME_2018$ASSAULT, .025)
741
```

```
743 # at or below 2.5th percentile metro statistical areas in terms of assault per 100,000
744
745 US_CRIME_2018[US_CRIME_2018$ASSAULT <= 61.59,]
746
747 sub_by_assault_2_5 <- as.data.frame(US_CRIME_2018[US_CRIME_2018$ASSAULT <= 61.59,])
748
749 sub_by_assault_2_5_order <- order(sub_by_assault_2_5$ASSAULT,decreasing = FALSE)
750
751 sub_by_assault_2_5[sub_by_assault_2_5_order,]
752
753 BOTTOM_ASSAULT <- as.data.frame(sub_by_assault_2_5[sub_by_assault_2_5_order,])
754
755 #install.packages("ggplot2", dependencies = TRUE)
756 #library("ggplot2")
757 #install.packages("ggpubr", dependencies = TRUE)
758 #library("ggpubr")
759
760 # barplot Cities with ASSAULT At or Below 2.5th Percentile in ascending order
761
762 BOTTOM_ASSAULT_BAR <- ggplot(BOTTOM_ASSAULT , aes(x = reorder(CITY,ASSAULT),
                                                 y = ASSAULT)) +
763
      geom_bar(fill = "#ca3886", stat = "identity") +
764
765
      geom_text(aes(label = ASSAULT), vjust = -0.3) +
766
      theme_pubclean() +
767
      ggtitle("Assault Lower 2.5% of Sample") +
768
      xlab("METRO") + ylab("ASSAULT per 100,000") +
769
      ggpubr::rotate_x_text() +
770
      theme(plot.title = element_text(hjust = 0.5))
771
774 # Assault barplot with lower 2.5% and outliers
775
776 #install.packages("magrittr", dependencies = TRUE)
777 #library("magrittr")
778 #install.packages("ggpubr", dependencies = TRUE)
779 #library("ggpubr")
780
781 ASSAULT_BAR <- ggarrange(BOTTOM_ASSAULT_BAR,
782
                          OUTLIER_ASSAULT_BAR,
783
                          ncol = 2,
784
                          nrow = 1)
785 ASSAULT_BAR
786
```

```
790 # US Crime 2018 features
791
792 # univariate distribution analysis
793
794 # Burglary
795
796 # density, histogram, boxplot, outliers, lower 2.5%
797 # (percentile) crime, qqplots, and shapiro-wilk tests
798
800
801 #install.packages("lattice", dependencies = TRUE)
802 #library("lattice")
803
804 #install.packages("gridExtra", dependencies = TRUE)
805 #library("gridExtra")
806
807 #install.packages("goft", dependencies = TRUE)
808 #library("goft")
809
810 #install.packages("ggplot2", dependencies = TRUE)
811 #library("ggplot2")
812
813 #install.packages("magrittr", dependencies = TRUE)
814 #library("magrittr")
815
816 #install.packages("ggpubr", dependencies = TRUE)
817 #library("ggpubr")
818
821 # Burglary
822
823 #install.packages("lattice", dependencies = TRUE)
824 #library("lattice")
825
826 # Density
827
828 BURGLARY_DENSITY <- densityplot(~BURGLARY, data = US_CRIME_2018,
829
                                     main="Burglary Density Plot",
                                     col = "#2b507c")
830
831
832 # Histogram
833
834 BURGLARY_HISTOGRAM <- histogram(x=~BURGLARY,data=US_CRIME_2018,
                                     type="density",
main="Burglary Histogram",
835
836
                                     col = "#2b507c",
837
838
                                     nint = 50)
839
840 # test distribution is normal
841
842 shapiro.test(x=US_CRIME_2018$BURGLARY)
```

```
844 # Normal QQ-Plot
845
846
    BURGLARY_QQ_QNORM <- qqmath(x=~BURGLARY, data = US_CRIME_2018,
847
                                 distribution = gnorm,
848
                                 prepanel = prepanel.qqmathline,
849
                                 panel = function(x, ...)
850
                                   panel.qqmathline(x, ...)
851
                                   panel.qqmath(x, ...)
852
                                 3.
853
                                 main = "Burglary Normal QQ-Plot \n SW-Test p-value = 0",
                                 col = "#2b507c")
854
855
856 # test distribution is log normal
857
858 #install.packages("goft", dependencies = TRUE)
859 #library("goft")
860
861 lnorm_test(x=US_CRIME_2018$BURGLARY)
862
863 # Log-Normal QQ-Plot
864
865 BURGLARY_QQ_QLNORM <- qqmath(x=~BURGLARY, data = US_CRIME_2018,
                                  distribution = qlnorm,
866
867
                                  prepanel = prepanel.qqmathline,
868 -
                                  panel = function(x, ...) {
869
                                   panel.qqmathline(x, ...)
870
                                    panel.qqmath(x, ...)
871
                                  3.
                                  main = "Burglary Log-Normal QQ-Plot \n SW-Test p-value = 0.4335",
col = "#2b507c")
872
873
875 # combine 4 plots burglary
876
877 #install.packages("gridExtra", dependencies = TRUE)
878 # library("gridExtra")
879
880 grid.arrange(BURGLARY_DENSITY,
881
                 BURGLARY_HISTOGRAM,
882
                  BURGLARY_QQ_QNORM,
883
                 BURGLARY_QQ_QLNORM,
884
                 nco1=2)
885
887
888 # BURGLARY
889
890 # find outliers based on boxplot
891
892 OutVals_burglary <- boxplot(US_CRIME_2018$BURGLARY)$out
893
894 which(US_CRIME_2018$BURGLARY %in% OutVals_burglary)
895
896 US_CRIME_2018[c(6, 81, 120, 128, 142, 155, 169, 196, 203, 227, 263),]
897
898 sub_burglary_outlier <- as.data.frame(US_CRIME_2018[c(6, 81, 120, 128,</pre>
                                                          142, 155, 169, 196,
203, 227, 263),])
899
900
901
902 sub_burglary_outlier_order <- order(sub_burglary_outlier$BURGLARY,decreasing = TRUE)
903
904 sub_burglary_outlier[sub_burglary_outlier_order,]
905
906 OUTLIER_BURGLARY <- as.data.frame(sub_burglary_outlier[sub_burglary_outlier_order,])
```

```
908 # barplot of outliers for Burglary
909
910 #install.packages("ggplot2", dependencies = TRUE)
911 # library("ggplot2")
912 #install.packages("ggpubr", dependencies = TRUE)
913 # library("ggpubr")
914
915 OUTLIER_BURGLARY_BAR <- ggplot(OUTLIER_BURGLARY, aes(x = reorder(CITY,-BURGLARY),
916
                                                         y = BURGLARY)) +
       geom_bar(fill = "#2b507c", stat = "identity") +
917
       geom_text(aes(label = BURGLARY), vjust = -0.3) +
918
919
       theme_pubclean() +
920
       ggtitle("Burglary Outliers of Sample") +
921
       xlab("METRO") + ylab("BURGLARY per 100,000") +
922
       ggpubr::rotate_x_text() +
923
       theme(plot.title = element_text(hjust = 0.5))
924
926
927 # Burglary 2.5th percentile
928
929 quantile(US_CRIME_2018$BURGLARY, .025)
930
933 # at or below 2.5th percentile metro statistical areas in terms of burglary per 100,000
934
935 US_CRIME_2018[US_CRIME_2018$BURGLARY <= 137.430,]
936
937 sub_by_burglary_2_5 <- as.data.frame(US_CRIME_2018[US_CRIME_2018$BURGLARY <= 137.430,])
938
939 sub_by_burglary_2_5_order <- order(sub_by_burglary_2_5$BURGLARY,decreasing = FALSE)
940
941 sub_by_burglary_2_5[sub_by_burglary_2_5_order,]
942
943 BOTTOM_BURGLARY <- as.data.frame(sub_by_burglary_2_5[sub_by_burglary_2_5_order,])
944
945 #install.packages("ggplot2", dependencies = TRUE)
946 #library("ggplot2")
947 #install.packages("ggpubr", dependencies = TRUE)
948 #library("ggpubr")
949
950 # barplot Cities with BURGLARY At or Below 2.5th Percentile in ascending order
951
952 BOTTOM_BURGLARY_BAR <- ggplot(BOTTOM_BURGLARY , aes(x = reorder(CITY, BURGLARY), 
953
                                                       y = BURGLARY)) +
954
      geom_bar(fill = "#2b507c", stat = "identity") +
955
      geom_text(aes(label = BURGLARY), vjust = -0.3) +
956
      theme_pubclean() +
      ggtitle("Burglary Lower 2.5% of Sample") +
957
958
      xlab("METRO") + ylab("BURGLARY per 100,000") +
959
       ggpubr::rotate_x_text() +
960
      theme(plot.title = element_text(hjust = 0.5))
961
```

```
964 # Burglary barplot with lower 2.5% and outliers
965
966 #install.packages("magrittr", dependencies = TRUE)
967 #library("magrittr")
968 #install.packages("ggpubr", dependencies = TRUE)
969 #library("ggpubr")
970
971 BURGLARY_BAR <- ggarrange(BOTTOM_BURGLARY_BAR,
972
                           OUTLIER_BURGLARY_BAR,
973
                           ncol = 2,
974
                           nrow = 1)
975 BURGLARY_BAR
976
980 # US Crime 2018 features
981
 982 # univariate distribution analysis
 983
 984 # Larceny
 985
 986 # density, histogram, boxplot, outliers, lower 2.5%
 987 # (percentile) crime, qqplots, and shapiro-wilk tests
 988
 990
 991 #install.packages("lattice", dependencies = TRUE)
 992 #library("lattice")
 993
 994 #install.packages("gridExtra", dependencies = TRUE)
 995 #library("gridExtra")
 996
 997 #install.packages("goft", dependencies = TRUE)
998 #library("goft")
999
1000 #install.packages("ggplot2", dependencies = TRUE)
1001 #library("ggplot2")
1002
1003 #install.packages("magrittr", dependencies = TRUE)
1004 #library("magrittr")
1005
1006 #install.packages("ggpubr", dependencies = TRUE)
1007 #library("ggpubr")
1008
```

```
1011 # Larceny
1012
1013 #install.packages("lattice", dependencies = TRUE)
1014 #library("lattice")
1015
1016 # Density
1017
1018 LARCENY_DENSITY <- densityplot(~LARCENY, data = US_CRIME_2018,
1019
                                     main="Larceny Density Plot",
1020
                                     col = "#ff743b")
1021
1022 # Histogram
1023
1024 LARCENY_HISTOGRAM <- histogram(x=~LARCENY,data=US_CRIME_2018,
1025
                                     type="density",
1026
                                     main="Larceny Histogram",
1027
                                     col = "#ff743b",
1028
                                     nint = 50)
1029
1030 # test distribution is normal
1031
1032 shapiro.test(x=US_CRIME_2018$LARCENY)
1034 # Normal QQ-Plot
1035
1036 LARCENY_QQ_QNORM <- qqmath(x=~LARCENY, data = US_CRIME_2018,
1037
                                 distribution = qnorm,
1038
                                 prepanel = prepanel.qqmathline,
1039 -
                                 panel = function(x, ...) {
1040
                                  panel.qqmathline(x, ...)
1041
                                  panel.qqmath(x, ...)
1042
                                 3.
1043
                                 main = "Larceny Normal QQ-Plot \n SW-Test p-value = 0",
                                 col = "#ff743b")
1044
1045
1046 # test distribution is log normal
1047
1048 #install.packages("goft", dependencies = TRUE)
1049 #library("goft")
1050
1051 lnorm_test(x=US_CRIME_2018$LARCENY)
1052
1053 # Log-Normal QQ-Plot
1054
1055 LARCENY_QQ_QLNORM <- qqmath(x=~LARCENY, data = US_CRIME_2018,
1056
                                  distribution = qlnorm,
1057
                                  prepanel = prepanel.qqmathline,
                                  panel = function(x, ...) {
1058 -
                                   panel.qqmathline(x, ...)
1059
1060
                                   panel.qqmath(x, ...)
1061
                                  }.
1062
                                  main = "Larceny Log-Normal QQ-Plot \n SW-Test p-value = 0.04679",
1063
                                 col = "#ff743b")
```

```
1065 # combine 4 plots larceny
1066
1067 #install.packages("gridExtra", dependencies = TRUE)
1068 # library("gridExtra")
1069
1070 grid.arrange(LARCENY_DENSITY,
1071
                   LARCENY_HISTOGRAM,
1072
                   LARCENY_QQ_QNORM,
1073
                   LARCENY_QQ_QLNORM,
1074
                   ncol=2)
1075
1078 # LARCENY
1079
1080 # find outliers based on boxplot
1081
1082 OutVals_larceny <- boxplot(US_CRIME_2018$LARCENY)$out</pre>
1083
1084 which(US_CRIME_2018$LARCENY %in% OutVals_larceny)
1085
1086 US_CRIME_2018[c(10, 93, 96, 169, 203, 209, 276, 286, 292),]
1087
1088 sub_larceny_outlier <- as.data.frame(US_CRIME_2018[c(10, 93,
96, 169, 203, 209,
96, 169, 203, 209,
1090
                                                         276, 286, 292),])
1091
1092 sub_larceny_outlier_order <- order(sub_larceny_outlier$LARCENY,decreasing = TRUE)
1093
1094 sub_larceny_outlier[sub_larceny_outlier_order,]
1095
1096 OUTLIER_LARCENY <- as.data.frame(sub_larceny_outlier[sub_larceny_outlier_order,])
1098 # barplot of outliers for Larceny
1099
1100 #install.packages("ggplot2", dependencies = TRUE)
1101 # library("ggplot2")
1102 #install.packages("ggpubr", dependencies = TRUE)
1103 # library("ggpubr")
1104
1105 OUTLIER_LARCENY_BAR <- ggplot(OUTLIER_LARCENY, aes(x = reorder(CITY,-LARCENY),
1106
                                                          y = LARCENY)) +
        geom_bar(fill = "#ff743b", stat = "identity") +
1107
1108
        geom_text(aes(label = LARCENY), vjust = -0.3) +
1109
        theme_pubclean() +
1110
        ggtitle("Larceny Outliers of Sample") +
1111
        xlab("METRO") + ylab("LARCENY per 100,000") +
        ggpubr::rotate_x_text() +
theme(plot.title = element_text(hjust = 0.5))
1112
1113
1114
1116
1117 # Larceny 2.5th percentile
1118
1119 quantile(US_CRIME_2018$LARCENY, .025)
1120
```

```
1123 # at or below 2.5th percentile metro statistical areas in terms of larceny per 100,000
1124
1125 US_CRIME_2018[US_CRIME_2018$LARCENY <= 806.320,]
1126
1127 sub_by_larceny_2_5 <- as.data.frame(US_CRIME_2018[US_CRIME_2018$LARCENY <= 806.320,])
1128
1129 sub_by_larceny_2_5_order <- order(sub_by_larceny_2_5$LARCENY,decreasing = FALSE)
1130
1131 sub_by_larceny_2_5[sub_by_larceny_2_5_order,]
1132
1133 BOTTOM_LARCENY <- as.data.frame(sub_by_larceny_2_5[sub_by_larceny_2_5_order,])</pre>
1134
1135 #install.packages("ggplot2", dependencies = TRUE)
1136 #library("ggplot2")
1137 #install.packages("ggpubr", dependencies = TRUE)
1138 #library("ggpubr")
1139
1140 # barplot Cities with LARCENY At or Below 2.5th Percentile in ascending order
1141
1142 BOTTOM_LARCENY_BAR <- qgplot(BOTTOM_LARCENY), aes(x = reorder(CITY,LARCENY)),
1143
                                                    y = LARCENY)) +
       geom_bar(fill = "#ff743b", stat = "identity") +
1144
1145
       geom_text(aes(label = LARCENY), vjust = -0.3) +
1146
       theme_pubclean() +
1147
       ggtitle("Larceny Lower 2.5% of Sample") +
1148
       xlab("METRO") + ylab("LARCENY per 100,000") +
1149
       ggpubr::rotate_x_text() +
1150
       theme(plot.title = element_text(hjust = 0.5))
1151
1154 # Larceny barplot with lower 2.5% and outliers
1155
1156 #install.packages("magrittr", dependencies = TRUE)
1150 # Histarry("magrittr")
1158 #install.packages("ggpubr", dependencies = TRUE)
1159 #library("ggpubr")
1160
1161 LARCENY_BAR <- ggarrange(BOTTOM_LARCENY_BAR,
1162
                            OUTLIER_LARCENY_BAR,
1163
                            ncol = 2,
1164
                            nrow = 1)
1165 LARCENY_BAR
1166
```

```
1170 # US Crime 2018 features
1171
1172 # univariate distribution analysis
1173
1174 # Vehicle
1175
1176 # density, histogram, boxplot, outliers, lower 2.5%
1177 # (percentile) crime, qqplots, and shapiro-wilk tests
1178
1180
1181 #install.packages("lattice", dependencies = TRUE)
1182 #library("lattice")
1183
1184 #install.packages("gridExtra", dependencies = TRUE)
1185 #library("gridExtra")
1186
1187 #install.packages("goft", dependencies = TRUE)
1188 #library("goft")
1189
1190 #install.packages("ggplot2", dependencies = TRUE)
1191 #library("ggplot2")
1192
1193 #install.packages("magrittr", dependencies = TRUE)
1194 #library("magrittr")
1195
1196 #install.packages("ggpubr", dependencies = TRUE)
1197 #library("ggpubr")
1198
1201 # Vehicle
1202
1203 #install.packages("lattice", dependencies = TRUE)
1204 #library("lattice")
1205
1206 # Density
1207
1208 VEHICLE_DENSITY <- densityplot(~VEHICLE, data = US_CRIME_2018,
1209
                                     main="Vehicle Density Plot",
                                     col = "#c82300")
1210
1211
1212 # Histogram
1213
1214 VEHICLE_HISTOGRAM <- histogram(x=~VEHICLE,data=US_CRIME_2018,
1215
                                     type="density"
                                     main="Vehicle Histogram",
1216
1217
                                     col = "#c82300",
                                     nint = 50)
1218
1219
1220 # test distribution is normal
1221
1222 shapiro.test(x=US_CRIME_2018$VEHICLE)
```

```
1224 # Normal QQ-Plot
1225
1226 VEHICLE_QQ_QNORM <- qqmath(x=~VEHICLE, data = US_CRIME_2018,
1227
                               distribution = qnorm,
1228
                               prepanel = prepanel.qqmathline,
1229 -
                               panel = function(x, ...) {
1230
                                 panel.qqmathline(x, ...)
1231
                                 panel.qqmath(x, ...)
1232
1233
                               main = "Vehicle Normal QQ-Plot \n SW-Test p-value = 0",
                               col = "#c82300")
1234
1235
1236 # test distribution is log normal
1237
1238 #install.packages("goft", dependencies = TRUE)
1239 #library("goft")
1240
1241 lnorm_test(x=US_CRIME_2018$VEHICLE)
1242
1243 # Log-Normal QQ-Plot
1244
1245 VEHICLE_QQ_QLNORM <- qqmath(x=~VEHICLE, data = US_CRIME_2018,
1246
                                distribution = qlnorm,
1247
                                prepanel = prepanel.qqmathline,
                                panel = function(x, ...) {
1248 -
                                 panel.qqmathline(x, ...)
1249
1250
                                  panel.qqmath(x, ...)
1251
                                3.
                                main = "Vehicle Log-Normal QQ-Plot \n SW-Test p-value = 0.03367",
col = "#c82300")
1252
1253
1255 # combine 4 plots vehicle
1256
1257 #install.packages("gridExtra", dependencies = TRUE)
1258 # library("gridExtra")
1259
1260 grid.arrange(VEHICLE_DENSITY,
1261
                  VEHICLE HISTOGRAM.
1262
                  VEHICLE_QQ_QNORM,
1263
                  VEHICLE_QQ_QLNORM,
1264
                  nco1=2)
1265
1268 # VEHICLE
1269
1270 # find outliers based on boxplot
1271
1272 OutVals_vehicle <- boxplot(US_CRIME_2018$VEHICLE)$out
1273
1274 which(US_CRIME_2018$VEHICLE %in% OutVals_vehicle)
1275
1276 US_CRIME_2018[c(5, 10, 19, 81, 169, 203, 209, 238, 286),]
1277
1278 sub_vehicle_outlier <- as.data.frame(US_CRIME_2018[c(5, 10, 19, 81, 169,
1279
                                                           203, 209, 238, 286),])
1280
1281 sub_vehicle_outlier_order <- order(sub_vehicle_outlier$VEHICLE,decreasing = TRUE)</pre>
1282
1283 sub_vehicle_outlier[sub_vehicle_outlier_order,]
1284
1285 OUTLIER_VEHICLE <- as.data.frame(sub_vehicle_outlier[sub_vehicle_outlier_order,])
```

```
1287 # barplot of outliers for Vehicle
1288
1289 #install.packages("ggplot2", dependencies = TRUE)
1290 # library("ggplot2")
1291 #install.packages("ggpubr", dependencies = TRUE)
1292 # library("ggpubr")
1293
1294 OUTLIER_VEHICLE_BAR <- ggplot(OUTLIER_VEHICLE, aes(x = reorder(CITY,-VEHICLE),
1295
                                                     y = VEHICLE)) +
       geom_bar(fill = "#c82300", stat = "identity") +
1296
1297
       geom_text(aes(label = VEHICLE), vjust = -0.3) +
1298
       theme_pubclean() +
       ggtitle("Vehicle Outliers of Sample") +
1299
1300
       xlab("METRO") + ylab("VEHICLE per 100,000") +
1301
       ggpubr::rotate_x_text() +
1302
       theme(plot.title = element_text(hjust = 0.5))
1303
1305
1306 # Vehicle 2.5th percentile
1307
1308 quantile(US_CRIME_2018$VEHICLE, .025)
1309
```

```
1312 # at or below 2.5th percentile metro statistical areas in terms of vehicle per 100,000
1313
1314 US_CRIME_2018 US_CRIME_2018 VEHICLE <= 36.430, ]
1315
1316 sub_by_vehicle_2_5 <- as.data.frame(US_CRIME_2018[US_CRIME_2018$VEHICLE <= 36.430,])
1317
1318 sub_by_vehicle_2_5_order <- order(sub_by_larceny_2_5$VEHICLE,decreasing = FALSE)
1319
1320 sub_by_vehicle_2_5[sub_by_vehicle_2_5_order,]
1321
1322 BOTTOM_VEHICLE <- as.data.frame(sub_by_vehicle_2_5[sub_by_larceny_2_5_order,])</pre>
1323
1324 #install.packages("ggplot2", dependencies = TRUE)
1325 #library("ggplot2")
1326 #install.packages("ggpubr", dependencies = TRUE)
1327 #library("ggpubr")
1328
1329 # barplot Cities with VEHICLE At or Below 2.5th Percentile in ascending order
1330
1331 BOTTOM_VEHICLE_BAR <- ggplot(BOTTOM_VEHICLE , aes(x = reorder(CITY,VEHICLE),
1332
                                                      y = VEHICLE)) +
       geom_bar(fill = "#c82300", stat = "identity") +
1333
1334
       geom_text(aes(label = VEHICLE), vjust = -0.3) +
1335
       theme_pubclean() +
1336
       ggtitle("Vehicle Lower 2.5% of Sample") +
1337
       xlab("METRO") + ylab("VEHICLE per 100,000") +
1338
       ggpubr::rotate_x_text() +
1339
       theme(plot.title = element_text(hjust = 0.5))
1340
```

```
1343 # Vehicle barplot with lower 2.5% and outliers
1344
1345 #install.packages("magrittr", dependencies = TRUE)
1346 #library("magrittr")
1347 #install.packages("ggpubr", dependencies = TRUE)
1348 #library("ggpubr")
1349
1350 VEHICLE_BAR <- ggarrange(BOTTOM_VEHICLE_BAR,
1351
                       OUTLIER_VEHICLE_BAR,
1352
                       ncol = 2,
1353
                       nrow = 1
1354 VEHICLE_BAR
1355
1358
1359 # US Crime 2018 features
1360
1361 # correlation matrix
1362
1364
1365 #install.packages("ggplot2", dependencies = TRUE)
1366 #library("ggplot2")
1367
1368 #install.packages("GGally", dependencies = TRUE)
1369 library("GGally")
1370
1372
1373 # correlation matrix
1374
1375 # Custom options.
1376
1377 ggcorr(
     CRIME_2018_FEAT,
1378
1379
     label = TRUE,
     name = "Sample Correlation",
geom = "circle",
1380
1381
1382
     max_size = 20,
1383
     min_size = 4,
1384
     size = 4,
     hjust = 0.75,
1385
1386
     nbreaks = 6,
     angle = 0,
palette = "PuOr")
1387
1388
```

260

```
1392
1393 # US Crime 2018 features
1394
1395 # contour plots and ggpairs 2d combination plots
1396
1398
1399 #install.packages("ggplot2", dependencies = TRUE)
1400 #library("ggplot2")
1401
1402 #install.packages("GGally", dependencies = TRUE)
1403 #library("GGally")
1404
1405 # install.packages("scales")
1406 library("scales")
1407
1410 # ggpairs matrix
1411
1412 # upper = scatterplot
1413
1414 # diagnol = density
1415
1416 # lower = contour plot
1417
1418 - upperfun <- function(data, mapping){
1419
      ggplot(data = data, mapping = mapping)+
        geom_point(alpha = .2, col="#0D0887FF")
1420
1421 }
1422
1423 - diagfun <- function(data, mapping) {
      ggplot(data = data, mapping = mapping)+
geom_density(fill = "#B12A90FF", colour = "#0D0887FF")
1424
1425
1426 }
1427
1428 - lowerfun <- function(data,mapping){
1429
      ggplot(data = data, mapping = mapping)+
1430
        stat_density2d(aes(fill = stat(level)), geom="polygon") +
        scale_fill_viridis_c(option = "plasma") +
1431
1432
        theme(legend.position = "magma")
1433 }
1434 ggpairs(CRIME_2018_FEAT,upper = list(continuous = wrap(upperfun)),
1435
           lower = list(continuous = wrap(lowerfun)),
1436
           diag = list(continuous = wrap(diagfun))
1437 )
1438
```

1442 # US Crime 2018 original variables 1443 1444 # multivariate crime analysis 1445 1446 # Test multivariate normality and chi-square plot 1447 1449 1450 # install.packages("goft", dependencies = TRUE) 1451 # library("goft") 1452 1453 # install.packages("car", dependencies = TRUE) 1454 library("car") 1455 1457 1458 # https://ggplot2.tidyverse.org/reference/geom_qq.html 1459 1461 1462 # Shapiro-Wilk test for multivariate normality 1463 1464 # A generalization of Shapiro-Wilk test for multivariate normality 1465 # (Villasenor-Alva and GonzalezEstrada, 2009). 1466 1467 #install.packages("goft", dependencies = TRUE) 1468 # library("goft") 1469 1470 mvshapiro_test(X=data.matrix(CRIME_2018_FEAT)) 1476 1477 # US Crime 2018 standardized features 1478 1479 # univariate distribution analysis 1480 1481 # descriptives 1482 1484 1485 #install.packages("psych", dependencies = TRUE) 1486 #library("psych") 1487 1488 describe(SCALED_CRIME_2018_FEAT) 1489 1490 summary(SCALED_CRIME_2018_FEAT) 1491

```
1494
1495 # US Crime 2018 standardized features
1496
1497 # bivariate distribution analysis
1498
1499 # scatter matrix
1500
1502
1503 #install.packages("ggplot2", dependencies = TRUE)
1504 #library("ggplot2")
1505
1506 #install.packages("GGally", dependencies = TRUE)
1507 #library("GGally")
1508
1510
1511 # https://ggobi.github.io/ggally/rd.html#ggmatrix
1512
1514
1515 CRIME_scatM_S_FEAT <- ggpairs(SCALED_CRIME_2018_FEAT,
1516
                      lower = list(continuous = wrap("points", alpha = 0.25)))
1517 CRIME_scatM_S_FEAT
1518
1524 # Crime 2018 standardized features
1525
1526 # PCA
1527
1529
1530 # eigenvalues of correlation matrix
1531
1532 eigen(x=cor(SCALED_CRIME_2018_FEAT))
1533
1534 # PCA of standardized features
1535
1536 SCALED_CRIME_2018_FEAT_PCA <- prcomp(CRIME_2018_FEAT, scale = TRUE)
1537 SCALED_CRIME_2018_FEAT_PCA</pre>
1538
1539 summary(SCALED_CRIME_2018_FEAT_PCA)
1540
1541 SCALED_CRIME_2018_FEAT_PCA$sdev
1542
1543 SCALED_CRIME_2018_FEAT_PCA$rotation
1544
1545 SCALED_CRIME_2018_FEAT_PCA$center
1546
1547 SCALED_CRIME_2018_FEAT_PCA$scale
1548
1549 SCALED_CRIME_2018_FEAT_PCA$x
1550
```

1554 # Crime 2018 standardized features 1555 1556 # create dataframe with PCA scores 1557 1559 1560 # 7-d pca scores 1561 1562 PCA_CRIME_2018 <- as.data.frame(SCALED_CRIME_2018_FEAT_PCA\$x) 1563 PCA_CRIME_2018 1564 1565 # change PCAi to Yi notation 1566 1567 y1 <- PCA_CRIME_2018\$PC1 1568 y2 <- PCA_CRIME_2018\$PC2 1569 y3 <- PCA_CRIME_2018\$PC3 1570 y4 <- PCA_CRIME_2018\$PC4 1571 y5 <- PCA_CRIME_2018\$PC5 1572 y6 <- PCA_CRIME_2018\$PC6 1573 y7 <- PCA_CRIME_2018\$PC7 1574 1575 PCA_CRIME_2018 <- data.frame(y1, y2, y3, y4, y5, y6, y7) 1576 PCA_CRIME_2018 1577 1578 # rename CITY for final dataframe 1579 1580 CITY <- CRIME_2018\$CITY 1581 1582 # dataframe of pca scores with CITY variable 1583 1584 PCA_CRIME_2018 <- as.data.frame(cbind(CITY, PCA_CRIME_2018))</pre> 1585 PCA_CRIME_2018 1587 # set rownames (again) 1588 1589 row.names(PCA_CRIME_2018) <- PCA_CRIME_2018\$CITY 1590 1591 # check rownames 1592 1593 PCA_CRIME_2018 1594 1595 # standardized pca scores -- PC1, PC2, PC3 1596 1597 PC1_PC2_PC3_CRIME <- as.data.frame(PCA_CRIME_2018[,2:4])</pre> 1598 PC1_PC2_PC3_CRIME 1599 1600 # set rownames (again) 1601 1602 row.names(PC1_PC2_PC3_CRIME) <- PCA_CRIME_2018\$CITY 1603 1604 # check rownames 1605 1606 PC1_PC2_PC3_CRIME 1607 1608 # see ranges of pc1, pc2, pc3 1609 1610 summary(PC1_PC2_PC3_CRIME) 1611

```
1614
1615 # Crime 2018 standardized features
1616
1617 # percent of explained variance pca of scaled 2018 crime data
1618
1619 - ################
1620
1621 #install.packages("ggplot2", dependencies = TRUE)
1622 #library("ggplot2")
1623 #install.packages("factoextra", dependencies = TRUE)
1624 library("factoextra")
1625
1626 - ################
1627
1628 fviz_screeplot(SCALED_CRIME_2018_FEAT_PCA, addlabels = TRUE, ylim = c(0, 90)) +
      labs(title = "Explained Standardized Sample Variance by Principal Component",
    subtitle = " for US Crime 2018",
1629
1630
1631
          x = "Standardized Sample Principal Components",
          y = "% of Explained Standardized Sample Variance") +
1632
1633
      theme(plot.title = element_text(hjust = 0.5),
           plot.subtitle = element_text(hjust = (0.5))
1634
1635
1639 # Crime 2018 standardized features
1640
1641 # Variable contributions to the principal axes scaled 2018 crime data
1642
1644
1645 #install.packages("ggplot2", dependencies = TRUE)
1646 #library("ggplot2")
1647
1648 #install.packages("factoextra", dependencies = TRUE)
1649 #library("factoextra")
1650
1651 #install.packages("magrittr", dependencies = TRUE)
1652 #library("magrittr")
1653
1654 #install.packages("ggpubr", dependencies = TRUE)
1655 #library("ggpubr")
1656
```

1659 # Percent Contributions of standardized variables to PC1 1660 1661 CONT_PC1 <- fviz_contrib(SCALED_CRIME_2018_FEAT_PCA, choice = "var", 1662 1663 axes = 1, top = 7, title = "Contributions to y1") + 1664 1665 1666 theme(plot.title = element_text(hjust = 0.5)) 1667 1668 # Percent Contributions of standardized variables to PC2 1669 1670 CONT_PC2 <- fviz_contrib(SCALED_CRIME_2018_FEAT_PCA, 1671 choice = "var", 1672 axes = 2, top = 7, title = "Contributions to y2") + 1673 1674 1675 theme(plot.title = element_text(hjust = 0.5)) 1676 1677 # Percent Contributions of standardized variables to PC3 1678 1679 CONT_PC3 <- fviz_contrib(SCALED_CRIME_2018_FEAT_PCA, choice = "var", 1680 1681 axes = 3, top = 7, title = "Contributions to y3") + 1682 1683 1684 theme(plot.title = element_text(hjust = 0.5)) 1685 1686 1689 # grid of plots contributions to principal components of 1690 # standardized variables Crime 2018 1691 1692 #install.packages("magrittr", dependencies = TRUE) 1693 #library("magrittr") 1694 #install.packages("ggpubr", dependencies = TRUE) 1695 #library("ggpubr") 1696 1697 CONT_PCS_PLOT_1_3 <- ggarrange(CONT_PC1, 1698 CONT_PC2, 1699 CONT_PC3, 1700 ncol = 2, 1701 nrow = 2) 1702 CONT_PCS_PLOT_1_3 1703 1707 # Crime 2018 PCA from standardized features 1708 1709 # correlation pc's with original components 1710 1712 1713 #install.packages("ggplot2", dependencies = TRUE)
1714 #library("ggplot2") 1715 1716 #install.packages("GGally", dependencies = TRUE) 1717 #library("GGally") 1718 1720 1721 # combined 2018 standardized features and pca scores 1722 1723 SCALED_PCA_AND_SCALED_CRIME_2018 <- as.data.frame(cbind(PCA_CRIME_2018, 1724 SCALED_CRIME_2018_FEAT)) 1725 1728 # Custom options angle = 0 1729 ggcorr(1730 SCALED_PCA_AND_SCALED_CRIME_2018[,-1], 1731 label = TRUE, name = "Sample Correlation", 1732 geom = "circle", 1733 $max_size = 20$, 1734 1735 min_size = 4, 1736 size = 4, hjust = 0.75, 1737 1738 nbreaks = 6, angle = 0, palette = "PuOr") 1739 1740 1741 1745 # Crime 2018 PCA from standardized features 1746 1747 # scatterplot matrix of PC1, PC2, PC3 components 1748 1749 # Scatterplot of pairs PC1, PC2, PC3 1750 1752 1753 #install.packages("ggplot2", dependencies = TRUE) 1754 #library("ggplot2") 1755

```
1758 # Scatterplot of pairs PC1, PC2, PC3
1759
1760 # Scatterplot PC1, PC2
1761
1762 PC1_PC2_SCATTER <- ggplot(PCA_CRIME_2018, aes(x=y1, y=y2)) +</pre>
1763
     geom_text(label=rownames(PCA_CRIME_2018)) +
     # ggtitle("Scatterplot of y2 ~ y1") +
labs(x = "y1 (63.05%)", y = "y2 (10.99%)") +
1764
1765
     theme(plot.title = element_text(hjust = 0.5))
1766
1767
1769
1770 # Scatterplot PC1, PC3
1771
1772 PC1_PC3_SCATTER <- ggplot(PCA_CRIME_2018, aes(x=y1, y=y3)) +</pre>
1773
     geom_text(label=rownames(PCA_CRIME_2018)) +
     # ggtitle("Scatterplot of y3 ~ y1") +
labs(x = "y1 (63.05%)", y = "y3 (9.383%)") +
1774
1775
     theme(plot.title = element_text(hjust = 0.5))
1776
1777
1779
1780 # Scatterplot PC2, PC3
1781
1782 PC2_PC3_SCATTER <- ggplot(PCA_CRIME_2018, aes(x=y2, y=y3)) +</pre>
1783
     geom_text(label=rownames(PCA_CRIME_2018)) +
     ggtitle("Scatterplot of y3 ~ y2") +
labs(x = "y2 (10.99%)", y = "y3 (9.383%)") +
1784
1785
1786
     theme(plot.title = element_text(hjust = 0.5))
1792
1793 # Clustering Methods
1794
1795 # Partitioning
1796
1797 # K-means
1798
1802 # K-means
1803
1804 # Scaled Crime 2018 Features
1805
1806 # PC1, PC2, PC3 from Scaled Crime 2018 Features
1807
1808 # Estimating the optimal number of clusters
1809
1810 # NbClust Method
1811
```

```
1814 # install.packages("factoextra")
1815 # library("factoextra")
1816
1817 # install.packages("NbClust", dependencies = TRUE)
1818 library("NbClust")
1819
1820 # install.packages("magrittr", dependencies = TRUE)
1821 # library("magrittr")
1822
1823 # install.packages("ggpubr", dependencies = TRUE)
1824 # library("ggpubr")
1825
1828 # Scaled Crime 2018 Features
1829
1830 # install.packages("NbClust", dependencies = TRUE)
1831 # library("NbClust")
1832
1833 NBCLUST_SCALE_CRIME_2018_FEAT_KMEANS <- NbClust(SCALED_CRIME_2018_FEAT,
1834
                                                     distance = "euclidean".
1835
                                                     min.nc = 2,
                                                     max.nc = 6,
1836
                                                     method = "kmeans")
1837
1838
1839 # install.packages("factoextra", dependencies = TRUE)
1840 # library("factoextra")
1841
1842 NBCLUST_SCALE_CRIME_2018_FEAT_KMEANS_BOX <- fviz_nbclust(NBCLUST_SCALE_CRIME_2018_FEAT_KMEANS,
                                                             barfill = "steelblue",
barcolor = "steelblue") +
1843
1844
       labs(title= "NbClust, Black-Box Method, k-Means, Input Standardized Crime 2018") +
1845
1846
       xlab("# of clusters k") +
       ylab("Freq. Among Indices") +
1847
1848
       theme(plot.title = element_text(hjust = 0.5))
1849
1852 # PC1, PC2, PC3 from Scaled Crime 2018 Features
1853
1854 # install.packages("NbClust", dependencies = TRUE)
1855 # library("NbClust")
1856
1857 NBCLUST_PC1_PC2_PC3_CRIME_2018_KMEANS <- NbClust(PC1_PC2_PC3_CRIME,
1858
                                                   distance = "euclidean",
1859
                                                   min.nc = 2.
1860
                                                   max.nc = 6.
1861
                                                   method = "kmeans")
1862
1863 # install.packages("factoextra", dependencies = TRUE)
1864 # library("factoextra")
1865
1866 NBCLUST_PC1_PC2_PC3_CRIME_2018_KMEANS_BOX <- fviz_nbclust(NBCLUST_PC1_PC2_PC3_CRIME_2018_KMEANS
1867
                                                           barfill = "steelblue",
barcolor = "steelblue") +
1868
1869
       labs(title= "NbClust, Black-Box Method, k-Means, Input y1, y2, y3") +
       xlab("# of clusters k") +
ylab("Freq. Among Indices")
1870
1871
1872
       theme(plot.title = element_text(hjust = 0.5))
1873
```

```
1876 # grid of plots for OPTIMAL K
1877
1878 # install.packages("magrittr", dependencies = TRUE)
1879 # library("magrittr")
1880 # install.packages("ggpubr", dependencies = TRUE)
1881 # library("ggpubr")
1882
1883 OPTIMAL_K_KMEAN_CRIME_2018 <- ggarrange(NBCLUST_SCALE_CRIME_2018_FEAT_KMEANS_BOX,
1884
                                     NBCLUST_PC1_PC2_PC3_CRIME_2018_KMEANS_BOX,
1885
                                      ncol = 1,
1886
                                      nrow = 2
1887
1891
1892 # K-means using eclust() in "factoextra" package
1893
1894 \# k = 3
1895
1896 # Scaled Crime 2018 Features
1897
1898 # PC1, PC2, PC3 from Scaled Crime 2018 Features
1899
1900 # create dataframe with CITY, 7 Crime 2018 variables,
1901 # PC1 PC2 PC3, k=3 kmean cluster assignment(s), and rownames
1902
1905 # install.packages("factoextra")
1906 # library("factoextra")
1907
1908 #install.packages("plyr", dependencies = TRUE)
1909 library("plyr")
1910
1911 # Compute k-means with k = 3 with scaled crime 2018 features
1912
1913 # install.packages("factoextra", dependencies = TRUE)
1914 # library("factoextra")
1915
1916 set.seed(125)
1917 KM3_SCALED_CRIME <- eclust(SCALED_CRIME_2018_FEAT, "kmeans"
                            k = 3, nstart = 25, graph = FALSE)
1918
1919 KM3_SCALED_CRIME$size
1920
1921 # Compute k-means with k = 3 PC 1, 2, 3
1922
1923 # install.packages("factoextra", dependencies = TRUE)
1924 # library("factoextra")
1925
1926 set.seed(126)
1927 KM3_PC1_PC2_PC3_CRIME <- eclust(PC1_PC2_PC3_CRIME, "kmeans",
1928
                                 k = 3, nstart = 25, graph = FALSE)
1929 KM3_PC1_PC2_PC3_CRIME$size
1930
```

```
1933 # create factors from cluster assignments
1934
1935 # CITY <- CRIME_2018$CITY
1936 KM3_SCALED_ASSIGN <- as.character(KM3_SCALED_CRIME$cluster)
1937 KM3_PC1_PC2_PC3_ASSIGN <- as.character(KM3_PC1_PC2_PC3_CRIME$cluster)
1938
1939 #install.packages("plyr", dependencies = TRUE)
1940 #library("plyr")
1941
1942 KM3_SCALED_ASSIGN <- revalue(KM3_SCALED_ASSIGN, c("1"="2", "2"="3", "3"="1"))
1943 KM3_PC1_PC2_PC3_ASSIGN <- revalue(KM3_PC1_PC2_PC3_ASSIGN, c("1"="2", "2"="1", "3"="3"))</pre>
1944
1945 # create dataframe to combine all the data and results
1946
1947 CRIME_2018_ASSIGN <- as.data.frame(cbind(
1948
     CITY,
1949
      CRIME_2018_FEAT,
1950
      PC1_PC2_PC3_CRIME,
1951
      KM3_SCALED_ASSIGN,
1952
      KM3_PC1_PC2_PC3_ASSIGN)
1953)
1954
1955 CRIME_2018_ASSIGN
1956
1957 # set rownames CITY_CRIME_2018
1958
1959 row.names(CRIME_2018_ASSIGN) <- CRIME_2018_ASSIGN$CITY
1960
1961 # summary
1962
1963 summary(CRIME_2018_ASSIGN)
1968
1969 # K-means using eclust() in "factoextra" package
1970
1971 \# k = 3
1972
1973 # Scaled Crime 2018 Features
1974
1975 # PC1, PC2, PC3 derived from Scaled Crime 2018 Features
1976
1977 # Cluster City Names, Cluster Mean Vectors, Cluster sd Vectors,
1978
1979 # check for differences in cluster assignments
1980
```

1982 1983 # cluster city names 1984 1985 # k-mean 3 cluster solution on standardized features 1986 1987 # cluster 1 1988 1989 sort(CRIME_2018_ASSIGN[CRIME_2018_ASSIGN\$KM3_SCALED_ASSIGN == "1",1]) 1990 1991 # cluster 2 1992 1993 sort(CRIME_2018_ASSIGN[CRIME_2018_ASSIGN\$KM3_SCALED_ASSIGN == "2",1]) 1994 1995 # cluster 3 1996 1997 sort(CRIME_2018_ASSIGN[CRIME_2018_ASSIGN\$KM3_SCALED_ASSIGN == "3",1]) 1998 2001 # cluster city names 2002 2003 # k-mean 3 cluster solution on PC1, PC2, PC3 derived from Scaled Crime 2018 Features 2004 2005 # cluster 1 2006 2007 sort(CRIME_2018_ASSIGN[CRIME_2018_ASSIGN\$KM3_PC1_PC2_PC3_ASSIGN == "1",1]) 2008 2009 # cluster 2 2010 2011 sort(CRIME_2018_ASSIGN[CRIME_2018_ASSIGN\$KM3_PC1_PC2_PC3_ASSIGN == "2",1]) 2012 2013 # cluster 3 2014 2015 sort(CRIME_2018_ASSIGN[CRIME_2018_ASSIGN%KM3_PC1_PC2_PC3_ASSIGN == "3",1]) 2016 2018 2019 # 1-way table of cluster assignments 2020 2021 table(KM3_SCALED_ASSIGN) 2022 2023 table(KM3_PC1_PC2_PC3_ASSIGN) 2024 2026 2027 # 2-way table of cluster assignments (confusion matrix) 2028 2029 table(KM3_SCALED_ASSIGN, KM3_PC1_PC2_PC3_ASSIGN)

272

```
2032
2033 # city names for misclassifications
2034
2035 # (in cluster "1" for KM3_SCALED_ASSIGN) and (in cluster "2" for KM3_PC1_PC2_PC3_ASSIGN )
2036
2037 CRIME_2018_ASSIGN [CRIME_2018_ASSIGN $KM3_SCALED_ASSIGN == "1"
2038
                  &
2039
                  CRIME_2018_ASSIGN$KM3_PC1_PC2_PC3_ASSIGN == "2",
2040
                  c(12,13)]
2041
2042 # (in cluster "2" for KM3_SCALED_ASSIGN) and (in cluster "3" for KM3_PC1_PC2_PC3_ASSIGN )
2043
2044 CRIME_2018_ASSIGN [CRIME_2018_ASSIGN $KM3_SCALED_ASSIGN == "2"
2045
                  &
                  CRIME_2018_ASSIGN$KM3_PC1_PC2_PC3_ASSIGN == "3",
2046
2047
                  c(12,13)]
2048
2050
2051 # cluster means, k-means, k=3, standardized Crime 2018 Features
2052
2053 aggregate(CRIME_2018_ASSIGN[, 2:8], list(CRIME_2018_ASSIGN$KM3_SCALED_ASSIGN), mean)
2054
2056
2057 # cluster means, k-means, k=3, PC1, PC2, PC3 derived from Scaled Crime 2018 Features
2058
2059 aggregate(CRIME_2018_ASSIGN[, 2:8], list(CRIME_2018_ASSIGN%KM3_PC1_PC2_PC3_ASSIGN), mean)
2060
2063 #install.packages("psych", dependencies = TRUE)
2064 # library("psych")
2065
2066 # original sample means
2067
2068 describe(CRIME_2018_FEAT)
2069
2074 # K-means using eclust() in "factoextra" package
2075
2076 \# k = 3
2077
2078 # Scaled Crime 2018 Features
2079
2080 # Scatterplots on PC1, PC2, PC3 of cluster assignments
2081
2083
2084 #install.packages("ggplot2", dependencies = TRUE)
2085 #library("ggplot2")
2086
2087 #install.packages("GGally", dependencies = TRUE)
2088 #library("GGally")
2089
2090 #install.packages("gridExtra", dependencies = TRUE)
2091 # library("gridExtra")
2092
```

```
2095 # 3 group k-mean scatterplot with CITY labels input standardized crime 2018 data.
2096 # Plotted on PC1, PC2
2097
2098 #install.packages("ggplot2", dependencies = TRUE)
2099 #library("ggplot2")
2100
2101 KM3_SCALE_CRIME_DIM1_DIM2 <- ggplot(CRIME_2018_ASSIGN,
2102
                                             aes(x=y1, y=y2, color=KM3_SCALED_ASSIGN)) +
2103
        geom_text(label=rownames(CRIME_2018_ASSIGN)) +
        # ggtitle("k-Means, k=3, Input Standardized Crime 2018") +
labs(x = "y1 (63.05%)", y = "y2 (10.99%)", col = "cluster") +
2104
2105
        scale_color_manual(values=c("#FE3A07", "#0080FF", "#009788")) +
2106
2107
        # theme(plot.title = element_text(hjust = 0.5)) +
2108
        # theme(legend.position = "none")
        theme(legend.position = "bottom")
2109
2110
2113 # 3 group k-mean scatterplot with CITY labels input standardized crime 2018 data.
2114 # Plotted on PC1, PC3
2115
2116 #install.packages("ggplot2", dependencies = TRUE)
2117 #library("ggplot2")
2118
2119 KM3_SCALE_CRIME_DIM1_DIM3 <- ggplot(CRIME_2018_ASSIGN,
2120
                                            aes(x=y1, y=y3, color=KM3_SCALED_ASSIGN)) +
        geom_text(label=rownames(CRIME_2018_ASSIGN)) +
2121
2122
        # ggtitle("k-Means, k=3, Input Standardized Crime 2018") +
        labs(x = "y1 (63.05%)", y = "y3 (9.383%)", col = "cluster") +
scale_color_manual(values=c("#FE3A07", "#0080FF", "#009788")) +
2123
2124
        # theme(plot.title = element_text(hjust = 0.5)) +
2125
        theme(legend.position = "none")
2126
2127
2130 # change legend position for final plot
2131
2132 #install.packages("gridExtra", dependencies = TRUE)
2133 # library("gridExtra")
2134
2135 - get_legend<-function(myggplot){</pre>
2136
       tmp <- ggplot_gtable(ggplot_build(myggplot))</pre>
2137
        leg <- which(sapply(tmp$grobs, function(x) x$name) == "guide-box")</pre>
2138
       legend <- tmp$grobs[[leg]]</pre>
2139
       return(legend)
2140 }
2141
2142 legend <- get_legend(KM3_SCALE_CRIME_DIM1_DIM2)</pre>
2143
2144 KM3_SCALE_CRIME_DIM1_DIM2 <- KM3_SCALE_CRIME_DIM1_DIM2 +
2145
                                   theme(legend.position = "none")
2146
```

```
2149 # Combine 2 scatterplots, K-Means, K=3
2150
2151 # Input Scaled Crime 2018 Features
2152
2153 #install.packages("gridExtra", dependencies = TRUE)
2154 # library("gridExtra")
2155
2156 grid.arrange(KM3_SCALE_CRIME_DIM1_DIM2,
2157
                KM3_SCALE_CRIME_DIM1_DIM3,
2158
                legend,
2159
                ncol=2,
2160
                nrow = 2.
2161
                layout_matrix = rbind(c(1,2), c(3,3)),
2162
                widths = c(2.7, 2.7), heights = c(2.5, 0.2),
                top = text_grob("k-Means, k=3, Input Standardized Crime 2018,
2163
                              Plotted on y1, y2, y3",
color = "black", face = "bold", size = 14))
2164
2165
2166
2170 # K-means using eclust() in "factoextra" package
2171
2172 # k = 3
2173
2174 # PC1, PC2, PC3 derived from Scaled Crime 2018 Features
2175
2176 # Scatterplots on PC1, PC2, PC3 of cluster assignments
2177
2179
2180 #install.packages("ggplot2", dependencies = TRUE)
2181 #library("ggplot2")
2182
2183 #install.packages("GGally", dependencies = TRUE)
2184 #library("GGally")
2185
2186 #install.packages("gridExtra", dependencies = TRUE)
2187 # library("gridExtra")
2188
2191 # 3 group k-mean scatterplot with CITY labels, input PC1, PC2, PC3
2192 # Plotted on PC1, PC2
2193
2194 #install.packages("ggplot2", dependencies = TRUE)
2195 #library("ggplot2")
2196
2197 KM3_PC1_PC2_PC3_DIM1_DIM2 <- ggplot(CRIME_2018_ASSIGN,
                                      aes(x=y1, y=y2, color=KM3_PC1_PC2_PC3_ASSIGN)) +
2198
2199
       geom_text(label=rownames(CRIME_2018_ASSIGN)) +
       # ggtitle("k-Means, k=3, Input y1, y2, y3") +
labs(x = "y1 (63.05%)", y = "y2 (10.99%)", col = "cluster") +
2200
2201
       scale_color_manual(values=c("#FE3A07", "#0080FF", "#009788")) +
2202
2203
       theme(plot.title = element_text(hjust = 0.5)) +
2204
       # theme(legend.position = "none")
       theme(legend.position = "bottom")
2205
2206
```

```
2209 # 3 group k-mean scatterplot with CITY labels, input PC1, PC2, PC3
2210 # Plotted on PC1, PC3
2211
2212 #install.packages("ggplot2", dependencies = TRUE)
2213 #library("ggplot2")
2214
2215 KM3_PC1_PC2_PC3_DIM1_DIM3 <- ggplot(CRIME_2018_ASSIGN,
                                    aes(x=y1, y=y3, color=KM3_PC1_PC2_PC3_ASSIGN)) +
2216
2217
      geom_text(label=rownames(CRIME_2018_ASSIGN)) +
2218
       # ggtitle("k-Means, k=3, Input y1, y2, y3") +
      labs(x = "y1 (63.05%)", y = "y3 (9.383%)", col = "cluster") + scale_color_manual(values=c("#FE3A07", "#0080FF", "#009788")) +
2219
2220
2221
      # theme(plot.title = element_text(hjust = 0.5)) +
2222
       theme(legend.position = "none")
2223
2226 # change legend position for final plot
2227
2228 #install.packages("gridExtra", dependencies = TRUE)
2229 # library("gridExtra")
2230
2231 - get_legend<-function(myggplot){</pre>
      tmp <- ggplot_gtable(ggplot_build(myggplot))</pre>
2232
2233
      leg <- which(sapply(tmp$grobs, function(x) x$name) == "guide-box")</pre>
2234
      legend <- tmp$grobs[[leg]]</pre>
      return(legend)
2235
2236
     3
2237
2238 legend <- get_legend(KM3_PC1_PC2_PC3_DIM1_DIM2)</pre>
2239
2240 KM3_PC1_PC2_PC3_DIM1_DIM2 <- KM3_PC1_PC2_PC3_DIM1_DIM2 +
2241
                              theme(legend.position = "none")
2242
2245
2246 # Combine 2 scatterplots, K-Means K=3
2247
2248 # Input PC1, PC2, PC3
2249
2250 #install.packages("gridExtra", dependencies = TRUE)
2251 # library("gridExtra")
2252
2253 grid.arrange(KM3_PC1_PC2_PC3_DIM1_DIM2,
2254
                KM3_PC1_PC2_PC3_DIM1_DIM3,
                legend.
2255
2256
                ncol=2,
2257
                nrow=2.
2258
                layout_matrix = rbind(c(1,2), c(3,3)),
                2259
2260
2261
2262
```

```
2267 # K-means using eclust() in "factoextra" package
2268
2269 \# k = 3
2270
2271 # Standardized Crime 2018 Features
2272
2273 # Scattermatrix on original dimensions
2274
2276
2277 #install.packages("ggplot2", dependencies = TRUE)
2278 #library("ggplot2")
2279
2280 #install.packages("GGally", dependencies = TRUE)
2281 #library("GGally")
2282
2286
2287 # ggpairs
2288
2289 # k = 3
2290
2291 # K-Means
2292
2293 p <- ggpairs(CRIME_2018_ASSIGN, c(2,3,4,5,6,7,8,12),
2294 mapping = ggplot2::aes_string(color = "KM3_SCALED_ASSIGN"))
2295
2296 - for(i in 1:p$nrow) {
2297 - for(j in 1:p$ncol){
2298
        p[i,j] <- p[i,j] +
          scale_fill_manual(values=c("#FE3A07", "#0080FF", "#009788")) +
scale_color_manual(values=c("#FE3A07", "#0080FF", "#009788"))
2299
2300
2301
      }
2302 }
2303
2304 p
2308
2309 # K-means using eclust() in "factoextra" package
2310
2311 \# k = 3
2312
2313 # PC1, PC2, PC3 derived from Scaled Crime 2018 Features
2314
2315 # Scattermatrix on original dimensions
2316
2318
2319 #install.packages("ggplot2", dependencies = TRUE)
2320 #library("ggplot2")
2321
2322 #install.packages("GGally", dependencies = TRUE)
2323 #library("GGally")
2324
```

2327 # PC1, PC2, PC3 derived from Scaled Crime 2018 Features 2328 2329 # ggpairs 2330 2331 # k = 32332 2333 # K-Means 2334 2335 p <- ggpairs(CRIME_2018_ASSIGN, c(2,3,4,5,6,7,8,13), mapping = ggplot2::aes_string(color = "KM3_PC1_PC2_PC3_ASSIGN")) 2336 2337 2338 - for(i in 1:p\$nrow) { 2339 - for(j in 1:p\$ncol){ p[i,j] <- p[i,j] +
 scale_fill_manual(values=c("#FE3A07", "#0080FF", "#009788")) +</pre> 2340 2341 scale_color_manual(values=c("#FE3A07", "#0080FF", "#009788")) 2342 2343 } 2344 } 2345 2346 p 2347 2353 # Clustering Methods 2354 2355 # Hierarchical 2356 2357 # Agglomerative 2358 2361 2362 # Distance Matrices 2363 2364 # Scaled Crime 2018 Features 2365 2366 # PC1, PC2, PC3 from Scaled Crime 2018 Features 2367 2370 # Euclidean Distance Matrix on Scaled Crime 2018 Features 2371 2372 DIST_SCALED_CRIME <- dist(SCALED_CRIME_2018_FEAT, method = "euclidean") 2373 DIST_SCALED_CRIME 2374 2375 # Subset the first 5 columns and rows on Scaled Crime 2018 Features 2376 2377 round(as.matrix(DIST_SCALED_CRIME)[1:5, 1:5], 1) 2378 2380 2381 # Euclidean Distance Matrix on PC1, PC2, PC3 2382 2383 DIST_PC1_PC2_PC3_CRIME <- dist(PC1_PC2_PC3_CRIME, method = "euclidean") 2384 DIST_PC1_PC2_PC3_CRIME 2385 2386 # Subset the first 5 columns and rows on PC1, PC2, PC3 2387 2388 round(as.matrix(DIST_PC1_PC2_PC3_CRIME)[1:5, 1:5], 1)
```
2393
2394 # Agglomerative Clustering
2395
2396 # WARD.D2
2397
2398 # Scaled Crime 2018 Features
2399
2400 # PC1, PC2, PC3 from Scaled Crime 2018 Features
2401
2402 # Estimating the optimal number of clusters
2403
2404 # NbClust Method
2405
2407
2408 # install.packages("factoextra")
2409 # library("factoextra")
2410
2411 # install.packages("NbClust", dependencies = TRUE)
2412 # library("NbClust")
2413
2414 # install.packages("magrittr", dependencies = TRUE)
2415 # library("magrittr")
2416
2417 # install.packages("ggpubr", dependencies = TRUE)
2418 # library("ggpubr")
2419
2422 # Scaled Crime 2018 Features
2423
2424 # install.packages("NbClust", dependencies = TRUE)
2425 # library("NbClust")
2426
2427 NBCLUST_SCALE_CRIME_2018_FEAT_WARD.D2 <- NbClust(SCALED_CRIME_2018_FEAT,
2428
                                          distance = "euclidean",
2429
                                          min.nc = 2,
                                          max.nc = 6,
2430
2431
                                          method = "ward.D2")
2432
2433 # install.packages("factoextra", dependencies = TRUE)
2434 # library("factoextra")
2435
2436 NBCLUST_SCALE_CRIME_2018_FEAT_WARD.D2_BOX <- fviz_nbclust(NBCLUST_SCALE_CRIME_2018_FEAT_WARD.D2
                                                 barfill = "steelblue",
barcolor = "steelblue")
2437
2438
2439
      labs(title= "NbClust, Black-Box Method, Ward, Input Standardized US Crime 2018") +
     xlab("# of clusters k") +
2440
     ylab("Freq. Among Indices") +
2441
2442
      theme(plot.title = element_text(hjust = 0.5))
2443
```

```
2446 # PC1, PC2, PC3 from Scaled Crime 2018 Features
2447
2448 # install.packages("NbClust", dependencies = TRUE)
2449 # library("NbClust")
2450
2451 NBCLUST_PC1_PC2_PC3_CRIME_2018_WARD.D2 <- NbClust(PC1_PC2_PC3_CRIME,
2452
                                              distance = "euclidean",
2453
                                              min.nc = 2,
2454
                                             max.nc = 6,
2455
                                             method = "ward.D2")
2456
2457 # install.packages("factoextra", dependencies = TRUE)
2458 # library("factoextra")
2459
2460 NBCLUST_PC1_PC2_PC3_CRIME_2018_WARD.D2_BOX <- fviz_nbclust(NBCLUST_PC1_PC2_PC3_CRIME_2018_WARD.D2
                                                     barfill = "steelblue"
2461
                                                     barcolor = "steelblue") +
2462
      labs(title= "NbClust, Black-Box Method, Ward, Input y1, y2, y3") +
2463
     xlab("# of clusters k") +
ylab("Freq. Among Indices") +
2464
2465
      theme(plot.title = element_text(hjust = 0.5))
2466
2467
2469
2470 # grid of plots for OPTIMAL K
2471
2472 # install.packages("magrittr", dependencies = TRUE)
2473 # library("magritt")
2474 # install.packages("ggpubr", dependencies = TRUE)
2475 # library("ggpubr")
2476
2477 OPTIMAL_WARD.D2_CRIME_2018 <- ggarrange(NBCLUST_SCALE_CRIME_2018_FEAT_WARD.D2_BOX,
2478
                                   NBCLUST_PC1_PC2_PC3_CRIME_2018_WARD.D2_BOX,
2479
                                   ncol = 1.
2480
                                   nrow = 2
2481
2486 # Agglomerative Clustering
2487
2488 # WARD.D2
2489
2490 # Compute algorithm
2491
2492 # Scaled Crime 2018 Features
2493
2494 # PC1, PC2, PC3 from Scaled Crime 2018 Features
2495
2497
2498 # Compute "ward.d2" using Scaled Crime 2018 Features
2499
2500 set.seed(150)
2501 SCALED_WARD.D2 <- hclust(d = DIST_SCALED_CRIME, method = "ward.D2")
2502
2503 # Compute "ward.d2" using PC1, PC2, PC3
2504
2505 set.seed(151)
2506 PC1_PC2_PC3_WARD.D2 <- hclust(d = DIST_PC1_PC2_PC3_CRIME, method = "ward.D2")
2507
```

2512 # Agglomerative Clustering 2513 2514 # WARD.D2 2515 2516 # K = 3 2517 2518 # add cluster assignments 2519 2520 # Scaled Crime 2018 Features 2521 2522 # PC1, PC2, PC3 from Scaled Crime 2018 Features 2523 2525 2526 #install.packages("plyr", dependencies = TRUE) 2527 #library("plyr") 2528 2530 2531 # Cut tree into 3 clusters "ward.d2" from Scaled Crime 2018 Features 2532 2533 K3_SCALED_WARD.D2 <- as.character(cutree(SCALED_WARD.D2, k = 3))</pre> 2534 2535 table(K3_SCALED_WARD.D2) 2536 2537 # Cut tree into 3 clusters "ward.d2" from PC1, PC2, PC3 2538 2539 K3_PC1_PC2_PC3_WARD.D2 <- as.character(cutree(PC1_PC2_PC3_WARD.D2, k = 3)) 2540 2541 table(K3 PC1 PC2 PC3 WARD.D2) 2543 #install.packages("plyr", dependencies = TRUE)
2544 #library("plyr") 2545 2546 K3_SCALED_WARD.D2 <- revalue(K3_SCALED_WARD.D2, c("1"="3", "2"="2", "3"="1")) 2547 K3_PC1_PC2_PC3_WARD.D2 <- revalue(K3_PC1_PC2_PC3_WARD.D2, c("1"="3", "2"="2", "3"="1")) 2548 2549 # create dataframe to combine all the data and results 2550 2551 CRIME_2018_ASSIGN <- as.data.frame(cbind(2552 CRIME 2018 ASSIGN. K3_SCALED_WARD.D2, K3_PC1_PC2_PC3_WARD.D2) 2553 2554 2555) 2556 2557 CRIME 2018 ASSIGN 2558 2559 # set rownames CITY_CRIME_2018 2560 2561 row.names(CRIME_2018_ASSIGN) <- CRIME_2018_ASSIGN\$CITY 2562 2563 # summary 2564 2565 summary(CRIME_2018_ASSIGN) 2566

2571 # WARD.D2 2572 2573 # k = 3 2574 2575 # Scaled Crime 2018 Features 2576 2577 # PC1, PC2, PC3 derived from Scaled Crime 2018 Features 2578 2579 # Cluster City Names, Cluster Mean Vectors, Cluster sd Vectors, 2580 # check for differences in cluster assignments 2581 2583 2584 # cluster city names 2585 2586 # WARD.D2 3 cluster solution on standardized features 2587 2588 # cluster 1 2589 2590 sort(CRIME_2018_ASSIGN[CRIME_2018_ASSIGN\$K3_SCALED_WARD.D2 == "1",1]) 2591 2592 # cluster 2 2593 2594 sort(CRIME_2018_ASSIGN[CRIME_2018_ASSIGN\$K3_SCALED_WARD.D2 == "2",1]) 2595 2596 # cluster 3 2597 2598 sort(CRIME_2018_ASSIGN[CRIME_2018_ASSIGN\$K3_SCALED_WARD.D2 == "3",1]) 2599 2602 # cluster city names 2603 2604 # WARD.D2 3 cluster solution on PC1, PC2, PC3 derived from Scaled Crime 2018 Features 2605 2606 # cluster 1 2607 2608 sort(CRIME_2018_ASSIGN[CRIME_2018_ASSIGN%K3_PC1_PC2_PC3_WARD.D2== "1",1]) 2609 2610 # cluster 2 2611 2612 sort(CRIME_2018_ASSIGN[CRIME_2018_ASSIGN\$K3_PC1_PC2_PC3_WARD.D2 == "2",1]) 2613 2614 # cluster 3 2615 2616 sort(CRIME_2018_ASSIGN[CRIME_2018_ASSIGN\$K3_PC1_PC2_PC3_WARD.D2 == "3",1]) 2617 2619 2620 # 1-way table of cluster assignments 2621 2622 table(K3_SCALED_WARD.D2) 2623 2624 table(K3_PC1_PC2_PC3_WARD.D2) 2625 2627 2628 # 2-way table of cluster assignments (confusion matrix) 2629 2630 table(K3_SCALED_WARD.D2, K3_PC1_PC2_PC3_WARD.D2) 2631

```
2634 # city names for misclassifications
2635
2636 # (in cluster "1" for K3_SCALED_WARD.D2) and (in cluster "2" for K3_PC1_PC2_PC3_WARD.D2)
2637
2638 CRIME_2018_ASSIGN [CRIME_2018_ASSIGN $K3_SCALED_WARD.D2 == "1"
2639
2640
                  CRIME_2018_ASSIGN$K3_PC1_PC2_PC3_WARD.D2 == "2", c(14,15)]
2641
2642 # (in cluster "2" for K3_SCALED_WARD.D2) and (in cluster "3" for KM3_PC1_PC2_PC3_ASSIGN)
2643
2644 CRIME_2018_ASSIGN [CRIME_2018_ASSIGN %K3_SCALED_WARD.D2 == "2"
2645
                   &
2646
                  CRIME_2018_ASSIGN$K3_PC1_PC2_PC3_WARD.D2 == "3", c(14,15)]
2647
2648 # (in cluster "3" for K3_SCALED_WARD.D2) and (in cluster "2" for K3_PC1_PC2_PC3_WARD.D2)
2649
2650 CRIME_2018_ASSIGN[CRIME_2018_ASSIGN$K3_SCALED_WARD.D2 == "3"
2651
                   &
2652
                  CRIME_2018_ASSIGN$K3_PC1_PC2_PC3_WARD.D2 == "2", c(14,15)]
2653
2655
2656 # cluster means, WARD.D2, k=3, standardized Crime 2018 Features
2657
2658 aggregate(CRIME_2018_ASSIGN[, 2:8], list(CRIME_2018_ASSIGN$K3_SCALED_WARD.D2), mean)
2659
2668 #install.packages("psych", dependencies = TRUE)
2669 # library("psych")
2670
2671 # original sample means
2672
2673 describe(CRIME_2018_FEAT)
2674
2679 # Agglomerative Clustering
2680
2681 # WARD.D2
2682
2683 # K = 3
2684
2685 # Dendrograms
2686
2687 # Scaled Crime 2018 Features
2688
2690
2691 # install.packages("ggplot2", dependencies = TRUE)
2692 # library("ggplot2")
2693
2694 # install.packages("factoextra")
2695 # library("factoextra")
2696
```

```
2699 # WARD.D2, Input Scaled Crime 2018, Rectangle Dendrogram, K=3
2700
2701 DEND_RECTANGLE_K3_SCALED_WARD.D2 <- fviz_dend(SCALED_WARD.D2, k = 3, # Cut in three groups
2702
                                          cex = 0.5, # label size
                                          palette = c("#009788","#0080FF", "#FE3A07"),
color_labels_by_k = TRUE, # color labels by groups
2703
2704
2705
                                          rect = TRUE, # Add rectangle around groups
2706
                                           # main = NULL,
                                           main = "Ward, k=3, Input Standardized Crime 2018,
2707
2708
                                           Rectangular Dendrogram"
                                           rect_border = c("#009788", "#0080FF", "#FE3A07"),
2709
2710
                                           rect_fill = TRUE) +
2711
                                           theme(plot.title = element_text(hjust = 0.5))
2712
2716
2717 # Agglomerative Clustering
2718
2719 # WARD.D2
2720
2721 # K = 3
2722
2723 # Dendrograms
2724
2725 # PC1, PC2, PC3 from Scaled Crime 2018 Features
2726
2729 # install.packages("ggplot2", dependencies = TRUE)
2730 # library("ggplot2")
2731
2732 # install.packages("factoextra")
2733 # library("factoextra")
2734
2736
2737 # WARD.D2, Input PC1, PC2, PC3, Rectangle Dendrogram, K=3
2738
2739 DEND_RECTANGLE_K3_PC1_PC2_PC3_WARD.D2 <- fviz_dend(PC1_PC2_PC3_WARD.D2, k = 3, # Cut in two groups
                                             cex = 0.5, # label size
palette = c("#009788", "#FE3A07", "#0080FF"),
color_labels_by_k = TRUE, # color labels by groups
2741
2742
2743
                                             rect = TRUE, # Add rectangle around groups
                                             main = "Ward, k=3, Input y1, y2, y3,
2744
2745
                                             Rectangular Dendrogram"
2746
                                             rect_border = c("#009788", "#FE3A07", "#0080FF"),
2747
                                             rect_fill = TRUE) +
2748
                                             theme(plot.title = element_text(hjust = 0.5))
2749
2750
2751
2752
```

```
2754 # Agglomerative Clustering
2755
2756 # WARD.D2
2757
2758 # K = 3
2759
2760 # Scaled Crime 2018 Features
2761
2762 # Scatterplots on PC1, PC2, PC3 of cluster assignments and ggpairs matrix
2763
2765
2766 #install.packages("ggplot2", dependencies = TRUE)
2767 #library("ggplot2")
2768
2769 #install.packages("GGally", dependencies = TRUE)
2770 #library("GGally")
2771
2772 #install.packages("gridExtra", dependencies = TRUE)
2773 # library("gridExtra")
2774
2777 # 3 group WARD.D2 scatterplot with CITY labels input scaled crime 2018 data.
2778 # Plotted on PC1, PC2
2779
2780 #install.packages("ggplot2", dependencies = TRUE)
2781 #library("ggplot2")
2782
2783 K3_SCALED_WARD.D2_DIM1_DIM2 <- ggplot(CRIME_2018_ASSIGN, aes(x=y1, y=y2,
2784
                                                                    color=K3_SCALED_WARD.D2)) +
2785
        geom_text(label=rownames(CRIME_2018_ASSIGN)) +
       # ggtitle("Wards, k=3, Input Standardized Crime 2018") +
labs(x = "y1 (63.05%)", y = "y2 (10.99%)", col = "cluster") +
scale_color_manual(values=c( "#FE3A07", "#0080FF", "#009788")) +
2786
2787
2788
2789
        # theme(plot.title = element_text(hjust = 0.5)) +
2790
        # theme(legend.position = "none")
        theme(legend.position = "bottom")
2791
2792
2795 # 3 group WARD.D2 scatterplot with CITY labels input scaled crime 2018 data.
2796 # Plotted on PC1, PC3
2797
2798 #install.packages("ggplot2", dependencies = TRUE)
2799 #library("ggplot2")
2800
2801 K3_SCALED_WARD.D2_DIM1_DIM3 <- ggplot(CRIME_2018_ASSIGN, aes(x=y1, y=y3,
2802
                                                                     color=K3_SCALED_WARD.D2)) +
2803
        geom_text(label=rownames(CRIME_2018_ASSIGN)) +
        # ggtitle("Wards, k=3, Input Scaled Crime 2018") +
2804
       labs(x = "y1 (63.05%)", y = "y3 (9.383%)", col = "cluster") + scale_color_manual(values=c("#FE3A07", "#0080FF", "#009788")) +
2805
2806
        # theme(plot.title = element_text(hjust = 0.5)) +
2807
2808
        theme(legend.position = "none")
2809
```

```
2812 # change legend position for final plot
2813
2814 #install.packages("gridExtra", dependencies = TRUE)
2815 # library("gridExtra")
2816
2817 - get_legend<-function(myggplot){</pre>
2818
       tmp <- ggplot_gtable(ggplot_build(myggplot))</pre>
2819
       leg <- which(sapply(tmp$grobs, function(x) x$name) == "guide-box")</pre>
2820
       legend <- tmp$grobs[[leg]]</pre>
2821
       return(legend)
2822 }
2823
2824 legend <- get_legend(K3_SCALED_WARD.D2_DIM1_DIM2)</pre>
2825
2826 K3_SCALED_WARD.D2_DIM1_DIM2 <- K3_SCALED_WARD.D2_DIM1_DIM2 +
                                  theme(legend.position = "none")
2827
2828
2831 # Combine 2 scatterplots, WARD.D2, K=3
2832
2833 # Input Scaled Crime 2018 Features
2834
2835 #install.packages("gridExtra", dependencies = TRUE)
2836 # library("gridExtra")
2837
2838 grid.arrange(K3_SCALED_WARD.D2_DIM1_DIM2,
2839
                K3_SCALED_WARD.D2_DIM1_DIM3,
2840
                legend,
2841
                ncol=2.
2842
                nrow= 2,
2843
                layout_matrix = rbind(c(1,2), c(3,3)),
2844
                widths = c(2.7, 2.7), heights = c(2.5, 0.2),
                top = text_grob("Ward, k=3, Input Standardized Crime 2018, Plotted on y1, y2, y3",
2845
                              color = "black", face = "bold", size = 14))
2846
2847
2851 # Agglomerative Clustering
2852
2853 # WARD.D2
2854
2855 # K = 3
2856
2857 # PC1, PC2, PC3 from Scaled Crime 2018 Features
2858
2859 # Scatterplots on PC1, PC2, PC3 of cluster assignments
2860
2862
2863 #install.packages("ggplot2", dependencies = TRUE)
2864 #library("ggplot2")
2865
2866 #install.packages("GGally", dependencies = TRUE)
2867 #library("GGally")
2868
2869 #install.packages("gridExtra", dependencies = TRUE)
2870 # library("gridExtra")
2871
```

```
2874 # 3 group WARD.D2 scatterplot with CITY labels, input PC1, PC2, PC3
2875 # Plotted on PC1, PC2
2876
2877 #install.packages("ggplot2", dependencies = TRUE)
2878 #library("ggplot2")
2879
2880 K3_PC1_PC2_PC3_WARD.D2_DIM1_DIM2 <- ggplot(CRIME_2018_ASSIGN, aes(x=y1, y=y2,
                                                     color=K3_PC1_PC2_PC3_WARD.D2)) +
2881
2882
        geom_text(label=rownames(CRIME_2018_ASSIGN)) +
2883
        # ggtitle("Ward, k=3, Input y1, y2, y3") +
        labs(x = "y1 (63.05%)", y = "y2 (10.99%)", col = "cluster") +
2884
        scale_color_manual(values=c("#FE3A07", "#0080FF", "#009788")) +
2885
2886
        theme(plot.title = element_text(hjust = 0.5)) +
2887
        # theme(legend.position = "none")
        theme(legend.position = "bottom")
2888
2889
2892 # 3 group WARD.D2 scatterplot with CITY labels, input PC1, PC2, PC3
2893 # Plotted on PC1, PC3
2894
2895 #install.packages("ggplot2", dependencies = TRUE)
2896 #library("ggplot2")
2897
2898 K3_PC1_PC2_PC3_WARD.D2_DIM1_DIM3 <- ggplot(CRIME_2018_ASSIGN, aes(x=y1, y=y3,
2899
                                                    color=K3_PC1_PC2_PC3_WARD.D2)) +
2900
       geom_text(label=rownames(CRIME_2018_ASSIGN)) +
       # ggtitle("Ward, k=3, Input y1, y2, y3") +
labs(x = "y1 (63.05%)", y = "y3 (9.383%)", col = "cluster") +
scale_color_manual(values=c("#FE3A07", "#0080FF", "#009788")) +
2901
2902
2903
2904
       theme(plot.title = element_text(hjust = 0.5)) +
2905
        # theme(plot.title = element_text(hjust = 0.5)) +
2906
       theme(legend.position = "none")
2907
2908
2910 # change legend position for final plot
2911
2912 #install.packages("gridExtra", dependencies = TRUE)
2913 # library("gridExtra")
2914
2915 - get_legend<-function(myggplot){</pre>
2916
        tmp <- ggplot_gtable(ggplot_build(myggplot))</pre>
2917
        leg <- which(sapply(tmp$grobs, function(x) x$name) == "guide-box")</pre>
2918
        legend <- tmp$grobs[[leg]]</pre>
2919
       return(legend)
2920 }
2921
2922 legend <- get_legend(K3_PC1_PC2_PC3_WARD.D2_DIM1_DIM3)</pre>
2923
2924 K3_PC1_PC2_PC3_WARD.D2_DIM1_DIM3 <- K3_PC1_PC2_PC3_WARD.D2_DIM1_DIM3 +
2925
                                           theme(legend.position = "none")
2926
```

```
2929 # Combine 2 scatterplots, WARD.D2, K=3
2930
2931 # Input PC1, PC2, PC3 from Scaled Crime 2018 Features
2932
2933 #install.packages("gridExtra", dependencies = TRUE)
2934 # library("gridExtra")
2935
2936 grid.arrange(KM3_PC1_PC2_PC3_DIM1_DIM2,
2937
              KM3_PC1_PC2_PC3_DIM1_DIM3,
2938
              legend,
2939
              ncol=2,
2940
              nrow=2,
2941
              layout_matrix = rbind(c(1,2), c(3,3)),
2942
              widths = c(2.7, 2.7), heights = c(2.5, 0.2),
              2943
2944
2945
2950 # Agglomerative Clustering
2951
2952 # WARD.D2
2953
2954 \# k = 3
2955
2956 # Standardized Crime 2018 Features
2957
2958 # Scattermatrix on original dimensions
2959
2961
2962 #install.packages("ggplot2", dependencies = TRUE)
2963 #library("ggplot2")
2964
2965 #install.packages("GGally", dependencies = TRUE)
2966 #library("GGally")
2967
2970 # Scaled Crime 2018 Features
2971
2972 # ggpairs
2973
2974 \# k = 3
2975
2976 # WARD.D2
2977
2978 p <- ggpairs(CRIME_2018_ASSIGN, c(2,3,4,5,6,7,8,14),
             mapping = ggplot2::aes_string(color = "K3_SCALED_WARD.D2"))
2979
2980
2981 - for(i in 1:p$nrow) {
2982 - for(j in 1:p$ncol){
2983
      p[i,j] <- p[i,j] +
        scale_fill_manual(values=c("#FE3A07", "#0080FF", "#009788")) +
scale_color_manual(values=c("#FE3A07", "#0080FF", "#009788"))
2984
2985
2986
     }
2987 }
2988
2989 p
2990
```

```
2994 # Agglomerative Clustering
2995
2996 # WARD.D2
2997
2998 # k = 3
2999
3000 # PC1, PC2, PC3 derived from Scaled Crime 2018 Features
3001
3002 # Scattermatrix on original dimensions
3003
3005
3006 #install.packages("ggplot2", dependencies = TRUE)
3007 #library("ggplot2")
3008
3009 #install.packages("GGally", dependencies = TRUE)
3010 #library("GGally")
3011
3014 # PC1, PC2, PC3 derived from Scaled Crime 2018 Features
3015
3016 # ggpairs
3017
3018 \# k = 3
3019
3020 # WARD.D2
3021
3022 p <- ggpairs(CRIME_2018_ASSIGN, c(2,3,4,5,6,7,8,15),</pre>
             mapping = ggplot2::aes_string(color = "K3_PC1_PC2_PC3_WARD.D2"))
3023
3024
3025 - for(i in 1:p$nrow) {
3026 - for(j in 1:p$ncol){
3027
       p[i,j] <- p[i,j] +
        scale_fill_manual(values=c("#FE3A07", "#0080FF", "#009788")) +
scale_color_manual(values=c("#FE3A07", "#0080FF", "#009788"))
3028
3029
3030
     }
3031 }
3032
3033
    р
3034
```

3040 # Agglomerative Clustering 3041 3042 # AVERAGE 3043 3044 # Scaled Crime 2018 Features 3045 3046 # Estimating the optimal number of clusters 3047 3048 # NbClust Method 3049 3051 3052 # install.packages("factoextra") 3053 # library("factoextra") 3054 3055 # install.packages("NbClust", dependencies = TRUE) 3056 # library("NbClust") 3057 3058 # install.packages("magrittr", dependencies = TRUE) 3059 # library("magrittr") 3060 3061 # install.packages("ggpubr", dependencies = TRUE) 3062 # library("ggpubr") 3063 3066 # Scaled Crime 2018 Features 3067 3068 # install.packages("NbClust", dependencies = TRUE) 3069 # library("NbClust") 3070 3071 NBCLUST_SCALE_CRIME_2018_FEAT_AVERAGE <- NbClust(SCALED_CRIME_2018_FEAT, 3072 distance = "euclidean", 3073 min.nc = 2, 3074 max.nc = 6, 3075 method = "average") 3076 3077 # install.packages("factoextra", dependencies = TRUE) 3078 # library("factoextra") 3079 3080 NBCLUST_SCALE_CRIME_2018_FEAT_AVERAGE_BOX <- fviz_nbclust(NBCLUST_SCALE_CRIME_2018_FEAT_AVERAGE barfill = "steelblue", barcolor = "steelblue"; 3081 3082 3083 labs(title= "NbClust, Black-Box Method, Average, Input Standardized US Crime 2018") + xlab("# of clusters k") +
ylab("Freq. Among Indices") + 3084 3085 3086 theme(plot.title = element_text(hjust = 0.5)) 3087

290

```
3091 # Agglomerative Clustering
3092
3093 # AVERAGE
3094
3095 # Estimating the optimal number of clusters
3096
3097 # PC1, PC2, PC3 from Scaled Crime 2018 Features
3098
3100
3101 # install.packages("ggplot2", dependencies = TRUE)
3102 # library("ggplot2")
3103
3104 # install.packages("factoextra")
3105 # library("factoextra")
3106
3107 # install.packages("NbClust", dependencies = TRUE)
3108 # library("NbClust")
3109
3112 # PC1, PC2, PC3 from Scaled Crime 2018 Features
3113
3114 # install.packages("NbClust", dependencies = TRUE)
3115 # library("NbClust")
3116
3117 NBCLUST_PC1_PC2_PC3_CRIME_2018_AVERAGE <- NbClust(PC1_PC2_PC3_CRIME,
3118
                                               distance =
                                                         'euclidean".
3119
                                               min.nc = 2,
3120
                                               max.nc = 6,
3121
                                               method = "average")
3122
3123 # install.packages("factoextra", dependencies = TRUE)
3124 # library("factoextra")
3125
3126 NBCLUST_PC1_PC2_PC3_CRIME_2018_AVERAGE_BOX <- fviz_nbclust(NBCLUST_PC1_PC2_PC3_CRIME_2018_AVERAGE
                                                      barfill = "steelblue",
barcolor = "steelblue") +
3127
3128
3129
      labs(title= "NbClust, Black-Box Method, Average, Input y1, y2, y3") +
3130
      xlab("# of clusters k") +
ylab("Freq. Among Indices") +
3131
3132
      theme(plot.title = element_text(hjust = 0.5))
3133
3136 # grid of plots for OPTIMAL K
3137
3138 # install.packages("magrittr", dependencies = TRUE)
3139 # library("magrittr")
3140 # install.packages("ggpubr", dependencies = TRUE)
3141 # library("ggpubr")
3142
3143 OPTIMAL_AVERAGE_CRIME_2018 <- ggarrange(NBCLUST_SCALE_CRIME_2018_FEAT_AVERAGE_BOX,
                                       NBCLUST_PC1_PC2_PC3_CRIME_2018_AVERAGE_BOX,
3144
3145
                                       ncol = 1,
3146
                                       nrow = 2
3147
```

3152 # Agglomerative Clustering 3153 3154 # AVERAGE 3155 3156 # Compute algorithm 3157 3158 # Scaled Crime 2018 Features 3159 3160 # PC1, PC2, PC3 from Scaled Crime 2018 Features 3161 3163 3164 # Compute "average" using Scaled Crime 2018 Features 3165 3166 set.seed(153)
3167 SCALED_AVERAGE <- hclust(d = DIST_SCALED_CRIME, method = "average")</pre> 3168 3169 # Compute "average" using PC1, PC2, PC3 3170 3171 set.seed(154) 3172 PC1_PC2_PC3_AVERAGE <- hclust(d = DIST_PC1_PC2_PC3_CRIME, method = "average") 3173 3178 # Agglomerative Clustering 3179 3180 # AVERAGE 3181 3182 # K = 3 3183 3184 # add cluster assignments 3185 3186 # Scaled Crime 2018 Features 3187 3188 # PC1, PC2, PC3 from Scaled Crime 2018 Features 3189 3191 3192 #install.packages("plyr", dependencies = TRUE) 3193 #library("plyr") 3194

```
3197 # Cut tree into 3 clusters "AVERAGE" from Scaled Crime 2018 Features
3198
3199 K3_SCALED_AVERAGE <- as.character(cutree(SCALED_AVERAGE, k = 3))
3200
3201 table(K3_SCALED_AVERAGE)
3202
3203 # Cut tree into 3 clusters "AVERAGE" from PC1, PC2, PC3
3204
3205 K3_PC1_PC2_PC3_AVERAGE <- as.character(cutree(PC1_PC2_PC3_AVERAGE, k = 3))
3206
3207 table(K3 PC1 PC2 PC3 AVERAGE)
3208
3209 #install.packages("plyr", dependencies = TRUE)
3210 #library("plyr")
3211

      3212
      K3_SCALED_AVERAGE <- revalue(K3_SCALED_AVERAGE, c("1"="3", "2"="2", "3"="1"))</td>

      3213
      K3_PC1_PC2_PC3_AVERAGE <- revalue(K3_PC1_PC2_PC3_AVERAGE, c("1"="3", "2"="2", "3"="1"))</td>

3214
3215 # create dataframe to combine all the data and results
3216
3217 CRIME_2018_ASSIGN <- as.data.frame(cbind(</pre>
3218
      CRIME_2018_ASSIGN,
      K3_SCALED_AVERAGE,
3219
3220
      K3_PC1_PC2_PC3_AVERAGE)
3221 )
3222
3223 CRIME_2018_ASSIGN
3224
3225 # set rownames CITY_CRIME_2018
3226
3227 row.names(CRIME_2018_ASSIGN) <- CRIME_2018_ASSIGN$CITY
3229 # summary
3230
3231 summary(CRIME_2018_ASSIGN)
3232
3236
3237 # AVERAGE
3238
3239 # k = 3
3240
3241 # Scaled Crime 2018 Features
3242
3243 # PC1, PC2, PC3 derived from Scaled Crime 2018 Features
3244
3245 # Cluster City Names, Cluster Mean Vectors, Cluster sd Vectors,
3246 # check for differences in cluster assignments
3247
```

3250 # cluster city names 3251 3252 # AVERAGE 3 cluster solution on Scaled Features 2018 3253 3254 # cluster 1 3255 3256 sort(CRIME_2018_ASSIGN[CRIME_2018_ASSIGN\$K3_SCALED_AVERAGE == "1",1]) 3257 3258 # cluster 2 3259 3260 sort(CRIME_2018_ASSIGN[CRIME_2018_ASSIGN%K3_SCALED_AVERAGE == "2",1]) 3261 3262 # cluster 3 3263 3264 sort(CRIME_2018_ASSIGN[CRIME_2018_ASSIGN\$K3_SCALED_AVERAGE == "3",1]) 3265 3268 # cluster city names 3269 3270 # AVERAGE 3 cluster solution on PC1, PC2, PC3 derived from Scaled Crime 2018 Features 3271 3272 # cluster 1 3273 3274 sort(CRIME_2018_ASSIGN[CRIME_2018_ASSIGN\$K3_PC1_PC2_PC3_AVERAGE == "1",1]) 3275 3276 # cluster 2 3277 3278 sort(CRIME_2018_ASSIGN[CRIME_2018_ASSIGN\$K3_PC1_PC2_PC3_AVERAGE == "2",1]) 3279 3280 # cluster 3 3281 3282 sort(CRIME_2018_ASSIGN[CRIME_2018_ASSIGN\$K3_PC1_PC2_PC3_AVERAGE == "3",1]) 3283 3285 3286 # 1-way table of cluster assignments 3287 3288 table(K3_SCALED_AVERAGE) 3289 3290 table(K3_PC1_PC2_PC3_AVERAGE) 3291 3293 3294 # 2-way table of cluster assignments (confusion matrix) 3295 3296 table(K3_SCALED_AVERAGE, K3_PC1_PC2_PC3_AVERAGE) 3297

3306 # cluster means, AVERAGE, k=3, standardized Crime 2018 Features 3307 3308 aggregate(CRIME_2018_ASSIGN[, 2:8], list(CRIME_2018_ASSIGN\$K3_SCALED_AVERAGE), mean) 3309 3311 3312 # cluster means, AVERAGE, k=3, PC1, PC2, PC3 derived from Scaled Crime 2018 Features 3313 3314 aggregate(CRIME_2018_ASSIGN[, 2:8], list(CRIME_2018_ASSIGN\$K3_PC1_PC2_PC3_AVERAGE), mean) 3315 3317 3318 #install.packages("psych", dependencies = TRUE) 3319 # library("psych") 3320 3321 # original sample means 3322 3323 describe(CRIME_2018_FEAT) 3324 3329 # Agglomerative Clustering 3330 3331 # AVERAGE 3332 3333 # K = 33334 3335 # Dendrograms 3336 3337 # Scaled Crime 2018 Features 3338 3340 3341 # install.packages("ggplot2", dependencies = TRUE) 3342 # library("ggplot2") 3343 3344 # install.packages("factoextra") 3345 # library("factoextra") 3346 3349 # AVERAGE, Input Scaled Crime 2018, Rectangle Dendrogram, K=3 3350 3351 DEND RECTANGLE K3_SCALED AVERAGE <- fviz dend(SCALED AVERAGE, k = 3, # Cut in three groups cex = 0.5, # label size
palette = c("#009788", "#FE3A07", "#0080FF"),
color_labels_by_k = TRUE, # color labels by groups 3352 3353 3354 rect = TRUE, # Add rectangle around groups 3355 3356 # main = NULL, 3357 main = "Average, k=3, Input Standardized 3358 Crime 2018, Rectangular Dendrogram' rect_border = c("#009788", "#FE3A07", "#0080FF"), 3359 rect_fill = TRUE) + 3360 3361 theme(plot.title = element_text(hjust = 0.5)) 3362

3367 # Agglomerative Clustering 3368 3369 # AVERAGE 3370 3371 # K = 3 3372 3373 # Dendrograms 3374 3375 # PC1, PC2, PC3 from Scaled Crime 2018 Features 3376 3378 3379 # install.packages("ggplot2", dependencies = TRUE) 3380 # library("ggplot2") 3381 3382 # install.packages("factoextra") 3383 # library("factoextra") 3384 3387 # AVERAGE, Input PC1, PC2, PC3, Rectangle Dendrogram, K=3 3388 3389 DEND_RECTANGLE_K3_PC1_PC2_PC3_AVERAGE <- fviz_dend(PC1_PC2_PC3_AVERAGE, k = 3, # Cut in three groups cex = 0.5, # label size
palette = c("#009788", "#FE3A07", "#0080FF"), 3390 3391 color_labels_by_k = TRUE, # color labels by groups 3392 rect = TRUE, # Add rectangle around groups
main = "Average, k=3, Input y1, y2, y3, 3393 3394 Rectangular Dendrogram' 3395 rect_border = c("#009788", "#FE3A07", "#0080FF"), 3396 rect_fill = TRUE) + 3397 theme(plot.title = element_text(hjust = 0.5)) 3398 3399 3406 # AVERAGE 3407 3408 # K = 3 3409 3410 # Scaled Crime 2018 Features 3411 3412 # Scatterplots on PC1, PC2, PC3 of cluster assignments and ggpairs matrix 3413 3415 3416 #install.packages("ggplot2", dependencies = TRUE) 3417 #library("ggplot2") 3418 3419 #install.packages("GGally", dependencies = TRUE) 3420 #library("GGally") 3421 3422 #install.packages("gridExtra", dependencies = TRUE) 3423 # library("gridExtra") 3424 3426 3427 # 3 group AVERAGE scatterplot with CITY labels input scaled crime 2018 data. 3428 # Plotted on PC1, PC2 3429 3430 #install.packages("ggplot2", dependencies = TRUE) 3431 #library("ggplot2") 3432

```
3435 # 3 group AVERAGE scatterplot with CITY labels input scaled crime 2018 data.
3436 # Plotted on PC1, PC2
3437
3438 #install.packages("ggplot2", dependencies = TRUE)
3439 #library("ggplot2")
3440
3441 K3_SCALED_AVERAGE_DIM1_DIM2 <- ggplot(CRIME_2018_ASSIGN,
3442
                                               aes(x=y1, y=y2, color=K3_SCALED_AVERAGE)) +
        geom_text(label=rownames(CRIME_2018_ASSIGN)) +
3443
        # ggtitle("Average, k=3, Input Scaled Crime 2018") +
labs(x = "y1 (63.05%)", y = "y2 (10.99%)", col = "cluster") +
scale_color_manual(values=c( "#FE3A07", "#0080FF", "#009788")) +
3444
3445
3446
3447
        # theme(plot.title = element_text(hjust = 0.5)) +
3448
        # theme(legend.position = "none")
3449
        theme(legend.position = "bottom")
3450
3453 # 3 group AVERAGE scatterplot with CITY labels input scaled crime 2018 data.
3454 # Plotted on PC1, PC3
3455
3456 #install.packages("ggplot2", dependencies = TRUE)
3457 #library("ggplot2")
3458
3459 K3_SCALED_AVERAGE_DIM1_DIM3 <- ggplot(CRIME_2018_ASSIGN,
                                                  aes(x=y1, y=y3, color=K3_SCALED_AVERAGE)) +
3460
3461
        geom_text(label=rownames(CRIME_2018_ASSIGN))
3462
         # ggtitle("Average, k=3, Input Scaled Crime 2018") +
        labs(x = "y1 (63.05%)", y = "y3 (9.383%)", col = "cluster") +
scale_color_manual(values=c("#FE3A07", "#0080FF", "#009788")) +
3463
3464
3465
        # theme(plot.title = element_text(hjust = 0.5)) +
3466
        theme(legend.position = "none")
3467
3470 # change legend position for final plot
3471
3472 #install.packages("gridExtra", dependencies = TRUE)
3473 # library("gridExtra")
3474
3475 - get_legend<-function(myggplot){</pre>
        tmp <- ggplot_gtable(ggplot_build(myggplot))
leg <- which(sapply(tmp$grobs, function(x) x$name) == "guide-box")
legend <- tmp$grobs[[leg]]</pre>
3476
3477
3478
3479
        return(legend)
3480 }
3481
3482 legend <- get_legend(K3_SCALED_AVERAGE_DIM1_DIM2)</pre>
3483
3484 K3_SCALED_AVERAGE_DIM1_DIM2 <- K3_SCALED_AVERAGE_DIM1_DIM2 +
3485
                                        theme(legend.position = "none")
3486
```

297

```
3489 # Combine 2 scatterplots, AVERAGE, K=3
3490
3491 # Input Scaled Crime 2018 Features
3492
3493 #install.packages("gridExtra", dependencies = TRUE)
3494 # library("gridExtra")
3495
3496 grid.arrange(K3_SCALED_AVERAGE_DIM1_DIM2,
3497
                 K3_SCALED_AVERAGE_DIM1_DIM3,
3498
                 legend,
3499
                 ncol=2,
3500
                 nrow = 2.
                 layout_matrix = rbind(c(1,2), c(3,3)),
3501
                 widths = c(2.7, 2.7), heights = c(2.5, 0.2),
3502
3503
                 top = text_grob("Average, k=3, Input Standardized Crime 2018,
                                Plotted on y1, y2, y3",
color = "black", face = "bold", size = 14))
3504
3505
3506
3510 # Agglomerative Clustering
3511
3512 # AVERAGE
3513
3514 \# K = 3
3515
3516 # PC1, PC2, PC3 from Scaled Crime 2018 Features
3517
3518 # Scatterplots on PC1, PC2, PC3 of cluster assignments
3519
3521
3522 #install.packages("ggplot2", dependencies = TRUE)
3523 #library("ggplot2")
3524
3525 #install.packages("GGally", dependencies = TRUE)
3526 #library("GGally")
3527
3528 #install.packages("gridExtra", dependencies = TRUE)
3529 # library("gridExtra")
3530
3533 # 3 group AVERAGE scatterplot with CITY labels, input PC1, PC2, PC3
3534 # Plotted on PC1, PC2
3535
3536 #install.packages("ggplot2", dependencies = TRUE)
3537 #library("ggplot2")
3538
3539 K3_PC1_PC2_PC3_AVERAGE_DIM1_DIM2 <- ggplot(CRIME_2018_ASSIGN,
                                              aes(x=y1, y=y2, color=K3_PC1_PC2_PC3_AVERAGE)) +
3540
3541
       geom_text(label=rownames(CRIME_2018_ASSIGN)) +
       # ggtitle(Average, k=3, Input y1, y2, y3") +
labs(x = "y1 (63.05%)", y = "y2 (10.99%)", col = "cluster") +
scale_color_manual(values=c("#FE3A07", "#0080FF", "#009788")) +
3542
3543
3544
       theme(plot.title = element_text(hjust = 0.5)) +
# theme(legend.position = "none")
3545
3546
       theme(legend.position = "bottom")
3547
3548
```

```
3551 # 3 group AVERAGE scatterplot with CITY labels, input PC1, PC2, PC3
3552 # Plotted on PC1, PC3
3553
3554 #install.packages("ggplot2", dependencies = TRUE)
3555 #library("ggplot2")
3556
3557 K3_PC1_PC2_PC3_AVERAGE_DIM1_DIM3 <- ggplot(CRIME_2018_ASSIGN,
3558
                                          aes(x=y1, y=y3, color=K3_PC1_PC2_PC3_AVERAGE))
3559
      geom_text(label=rownames(CRIME_2018_ASSIGN)) +
      # ggtitle("Average, k=3, Input y1, y2, y3") +
labs(x = "y1 (63.05%)", y = "y3 (9.383%)", col = "cluster") +
scale_color_manual(values=c("#FE3A07", "#0080FF", "#009788")) +
3560
3561
3562
3563
      theme(plot.title = element_text(hjust = 0.5)) +
3564
      theme(legend.position = "none")
3565
3568 # change legend position for final plot
3569
3570 #install.packages("gridExtra", dependencies = TRUE)
3571 # library("gridExtra")
3572
3573 - get_legend<-function(myggplot){</pre>
3574
       tmp <- ggplot_gtable(ggplot_build(myggplot))</pre>
3575
       leg <- which(sapply(tmp$grobs, function(x) x$name) == "guide-box")</pre>
3576
      legend <- tmp$grobs[[leg]]</pre>
3577
      return(legend)
3578 }
3579
3580 legend <- get_legend(K3_PC1_PC2_PC3_AVERAGE_DIM1_DIM2)</pre>
3581
3582 K3_PC1_PC2_PC3_AVERAGE_DIM1_DIM2 <- K3_PC1_PC2_PC3_AVERAGE_DIM1_DIM2 +
3583
                                     theme(legend.position = "none")
3584
3587 # Combine 2 scatterplots, AVERAGE, K=3
3588
3589 # Input PC1, PC2, PC3 from Scaled Crime 2018 Features
3590
3591 #install.packages("gridExtra", dependencies = TRUE)
3592 # library("gridExtra")
3593
3594 grid.arrange(K3_PC1_PC2_PC3_AVERAGE_DIM1_DIM2,
3595
                 K3_PC1_PC2_PC3_AVERAGE_DIM1_DIM3,
                 legend,
3596
3597
                 ncol=2,
3598
                 nrow=2.
3599
                 layout_matrix = rbind(c(1,2), c(3,3)),
3600
                 widths = c(2.7, 2.7), heights = c(2.5, 0.2),
                 3601
3602
3603
3604
```

3609 # Agglomerative Clustering 3610 3611 # AVERAGE 3612 3613 # k = 3 3614 3615 # Standardized Crime 2018 Features 3616 3617 # Scattermatrix on original dimensions 3618 3620 3621 #install.packages("ggplot2", dependencies = TRUE) 3622 #library("ggplot2") 3623 3624 #install.packages("GGally", dependencies = TRUE) 3625 #library("GGally") 3626 3629 # Scaled Crime 2018 Features 3630 3631 # ggpairs 3632 3633 # k = 3 3634 3635 # AVERAGE 3636 3637 p <- ggpairs(CRIME_2018_ASSIGN, c(2,3,4,5,6,7,8,16), 3638 mapping = ggplot2::aes_string(color = "K3_SCALED_AVERAGE")) 3639 3640 - for(i in 1:p\$nrow) { 3641 - for(j in 1:p\$ncol){ 3642 p[i,j] <- p[i,j] + scale_fill_manual(values=c("#FE3A07", "#0080FF", "#009788")) +
scale_color_manual(values=c("#FE3A07", "#0080FF", "#009788")) 3643 3644 3645 } 3646 } 3647 3648 p 3649 3653 # Agglomerative Clustering 3654 3655 # AVERAGE 3656 3657 # k = 33658 3659 # PC1,PC2, PC3 from Scaled Crime 2018 Features 3660 3661 # Scattermatrix on original dimensions 3662 3664 3665 #install.packages("ggplot2", dependencies = TRUE)
3666 #library("ggplot2") 3667 3668 #install.packages("GGally", dependencies = TRUE) 3669 #library("GGally") 3670

3673 # PC1,PC2, PC3 from Scaled Crime 2018 Features 3674 3675 # ggpairs 3676 3677 # k = 3 3678 3679 # AVERAGE 3680 3681 p <- ggpairs(CRIME_2018_ASSIGN, c(2,3,4,5,6,7,8,17), 3682 mapping = ggplot2::aes_string(color = "K3_PC1_PC2_PC3_AVERAGE")) 3683 3684 - for(i in 1:p\$nrow) { 3685 - for(j in 1:p\$ncol){ p[i,j] <- p[i,j] + 3686 scale_fill_manual(values=c("#FE3A07", "#0080FF", "#009788")) +
scale_color_manual(values=c("#FE3A07", "#0080FF", "#009788")) 3687 3688 3689 } 3690 } 3691 3692 p 3693 3699 # AVERAGE, WARD.D2 3700 3701 # Scaled Crime 2018 Features 3702 3703 # PC1,PC2, PC3 from Scaled Crime 2018 Features 3704 3705 # Compare Common Dendrograms Branches using Tanglegrams 3706 3708 3709 # install.packages("dendextend", dependencies = TRUE) 3710 library("dendextend") 3711 3714 # create dendrograms that work with "dendextend" 3715 3716 # ward.d2 3717 3718 # scaled crime 2018 3719 3720 DEND_SCALE_WARD.D2 <- as.dendrogram(SCALED_WARD.D2) 3721 3722 # pc1, pc2, pc3 3723 3724 DEND_PC1_PC2_PC3_WARD.D2 <- as.dendrogram(PC1_PC2_PC3_WARD.D2) 3725 3726 # average 3727 3728 # scaled crime 2018 3729 3730 DEND_SCALE_AVERAGE <- as.dendrogram(SCALED_AVERAGE) 3731 3732 # pc1, pc2, pc3 3733 3734 DEND_PC1_PC2_PC3_AVERAGE <- as.dendrogram(PC1_PC2_PC3_AVERAGE) 3735

3738 # tanglegram 3739 3740 TANGLE_WARD_SCALE_PC1_PC2_PC3 <- tanglegram(dend1 = DEND_SCALE_WARD.D2, 3741 dend2 = DEND_PC1_PC2_PC3_WARD.D2, 3742 common_subtrees_color_branches = TRUE, 3743 main_left = "Ward, Input S. Crime 2018", 3744 main_right = "Ward, Input y1, y2, y3" 3745) 3746 3747 TANGLE_AVERAGE_SCALE_PC1_PC2_PC3 <- tanglegram(dend1 = DEND_SCALE_AVERAGE, 3748 dend2 = DEND_PC1_PC2_PC3_AVERAGE, 3749 common_subtrees_color_branches = TRUE, 3750 main_left = "Average, Input S. Crime 2018", main_right = "Average, Input y1, y2, y3" 3751 3752) 3753 3754 TANGLE_WARD_AVERAGE_SCALE <- tanglegram(dend1 = DEND_SCALE_WARD.D2, 3755 dend2 = DEND SCALE AVERAGE. 3756 common_subtrees_color_branches = TRUE, 3757 main_left = "Ward, Input S. Crime 2018" 3758 main_right = "Average, Input S. Crime 2018" 3759 3760 3761 TANGLE_WARD_AVERAGE_PC1_PC2_PC3 <- tanglegram(dend1 = DEND_PC1_PC2_PC3_WARD.D2, 3762 dend2 = DEND_PC1_PC2_PC3_AVERAGE, 3763 common_subtrees_color_branches = TRUE, 3764 main_left = "Ward, Input y1, y2, y3", 3765 main_right = "Average, Input y1, y2, y3", 3766)

3785 #install.packages("ggplot2", dependencies = TRUE)

3788 #install.packages("GGally", dependencies = TRUE)

3791 # install.packages("gridExtra", dependencies = TRUE)

3786 #library("ggplot2")

3789 #library("GGally")

3792 # library("gridExtra")

3784

3787

3790

3793

302

```
3796 # 3 group k-mean scatterplot with CITY labels input standardized crime 2018 data.
3797 # Plotted on PC1, PC2
3798
3799 #install.packages("ggplot2", dependencies = TRUE)
3800 #library("ggplot2")
3801
3802 KM3_SCALE_CRIME_DIM1_DIM2 <- ggplot(CRIME_2018_ASSIGN,
3803
                                            aes(x=y1, y=y2, color=KM3_SCALED_ASSIGN)) +
3804
        geom_text(label=rownames(CRIME_2018_ASSIGN)) +
        ggtitle("k-Means, k=3, Input Standardized Crime 2018") +
labs(x = "y1 (63.05%)", y = "y2 (10.99%)", col = "cluster") +
scale_color_manual(values=c("#FE3A07", "#0080FF", "#009788")) +
3805
3806
3807
3808
        theme(plot.title = element_text(hjust = 0.5))
3809 # theme(legend.position = "none")
3810 # theme(legend.position = "bottom")
3811
3814 # 3 group k-mean scatterplot with CITY labels, input PC1, PC2, PC3
3815 # Plotted on PC1, PC2
3816
3817 #install.packages("ggplot2", dependencies = TRUE)
3818 #library("ggplot2")
3819
3820 KM3_PC1_PC2_PC3_DIM1_DIM2 <- ggplot(CRIME_2018_ASSIGN,
3821
                                            aes(x=y1, y=y2, color=KM3_PC1_PC2_PC3_ASSIGN)) +
3822
        geom_text(label=rownames(CRIME_2018_ASSIGN)) +
        ggtitle("k-Means, k=3, Input y1, y2, y3") +
labs(x = "y1 (63.05%)", y = "y2 (10.99%)", col = "cluster") +
3823
3824
        scale_color_manual(values=c("#FE3A07", "#0080FF", "#009788")) +
3825
3826
        theme(plot.title = element_text(hjust = 0.5))
3827 # theme(legend.position = "none")
3828 # theme(legend.position = "bottom")
3829
3830
3832
3833 # 3 group WARD.D2 scatterplot with CITY labels input scaled crime 2018 data.
3834 # Plotted on PC1, PC2
3835
3836 #install.packages("ggplot2", dependencies = TRUE)
3837 #library("ggplot2")
3838
3839 K3_SCALED_WARD.D2_DIM1_DIM2 <- ggplot(CRIME_2018_ASSIGN,
                                            aes(x=y1, y=y2, color=K3_SCALED_WARD.D2)) +
3840
3841
        geom_text(label=rownames(CRIME_2018_ASSIGN)) +
3842
        ggtitle("Wards, k=3, Input Standardized Crime 2018") +
       labs(x = "y1 (63.05%)", y = "y2 (10.99%)", col = "cluster") +
scale_color_manual(values=c( "#FE3A07", "#0080FF", "#009788")) +
3843
3844
       theme(plot.title = element_text(hjust = 0.5))
3845
3846 # theme(legend.position = "none")
3847 # theme(legend.position = "bottom")
3848
```

```
3851 # 3 group WARD.D2 scatterplot with CITY labels, input PC1, PC2, PC3
3852 # Plotted on PC1, PC2
3853
3854 #install.packages("ggplot2", dependencies = TRUE)
3855 #library("ggplot2")
3856
3857 K3_PC1_PC2_PC3_WARD.D2_DIM1_DIM2 <- ggplot(CRIME_2018_ASSIGN,
3858
                   aes(x=y1, y=y2, color=K3_PC1_PC2_PC3_WARD.D2)) +
       geom_text(label=rownames(CRIME_2018_ASSIGN)) +
3859
       ggtitle("ward, k=3, Input y1, y2, y3") +
labs(x = "y1 (63.05%)", y = "y2 (10.99%)", col = "cluster") +
scale_color_manual(values=c("#FE3A07", "#0080FF", "#009788")) +
3860
3861
3862
        theme(plot.title = element_text(hjust = 0.5))
3863
3864 # theme(legend.position = "none")
3865 # theme(legend.position = "bottom")
3866
3869 # 3 group AVERAGE scatterplot with CITY labels input scaled crime 2018 data.
3870 # Plotted on PC1, PC2
3871
3872 #install.packages("ggplot2", dependencies = TRUE)
3873 #library("ggplot2")
3874
3875 K3_SCALED_AVERAGE_DIM1_DIM2 <- ggplot(CRIME_2018_ASSIGN,
3876
                  aes(x=y1, y=y2, color=K3_SCALED_AVERAGE)) +
3877
        geom_text(label=rownames(CRIME_2018_ASSIGN)) +
       ggtitle("Average, k=3, Input Scaled Crime 2018") +
labs(x = "y1 (63.05%)", y = "y2 (10.99%)", col = "cluster") +
scale_color_manual(values=c( "#FE3A07", "#0080FF", "#009788")) +
3878
3879
3880
        theme(plot.title = element_text(hjust = 0.5))
3881
3882 # theme(legend.position = "none")
3883 # theme(legend.position = "bottom")
3884
3887 # 3 group AVERAGE scatterplot with CITY labels, input PC1, PC2, PC3
3888 # Plotted on PC1, PC2
3889
3890 #install.packages("ggplot2", dependencies = TRUE)
3891 #library("ggplot2")
3892
3893 K3_PC1_PC2_PC3_AVERAGE_DIM1_DIM2 <- ggplot(CRIME_2018_ASSIGN,
3894
                       aes(x=y1, y=y2, color=K3_PC1_PC2_PC3_AVERAGE)) +
3895
        geom_text(label=rownames(CRIME_2018_ASSIGN)) +
3896
        ggtitle("Average, k=3, Input y1, y2, y3") +
        labs(x = "y1 (63.05%)", y = "y2 (10.99%)", col = "cluster") +
3897
        scale_color_manual(values=c("#FE3A07", "#0080FF", "#009788")) +
3898
3899
        theme(plot.title = element_text(hjust = 0.5))
3900 # theme(legend.position = "none")
3901 # theme(legend.position = "bottom")
3902
```

```
3905 # combined plots
3906
3907 # Scaled Crime 2018 Features
3908
3909 # PC1, PC2, PC3 from Scaled Crime 2018 Features
3910
3911 grid.arrange(
3912
      KM3_SCALE_CRIME_DIM1_DIM2,
3913
      KM3_PC1_PC2_PC3_DIM1_DIM2,
3914
      K3_SCALED_WARD.D2_DIM1_DIM2,
3915
      K3_PC1_PC2_PC3_WARD.D2_DIM1_DIM2,
3916
      K3_SCALED_AVERAGE_DIM1_DIM2,
3917
      K3_PC1_PC2_PC3_AVERAGE_DIM1_DIM2,
3918
      nrow = 3) \#,
3919 # top = text_grob("K=3, K-Means Row 1, WARD Row 2, AVERAGE Row 3",
                  color = "black", face = "bold", size = 14))
3920 #
3921
3924 # K=3, KMEANs, Cluster assignments
3925
3926 summary(CRIME_2018_ASSIGN[,c(12,13)])
3927
3928 # K=3, WARD, Cluster assignments
3929
3930 summary(CRIME_2018_ASSIGN[,c(14,15)])
3931
3932 # K=3, AVERAGE, Cluster assignments
3933
3934 summary(CRIME_2018_ASSIGN[,c(16,17)])
3935
3937
3938 CRIME_2018_ASSIGN[CRIME_2018_ASSIGN%KM3_SCALED_ASSIGN == "1",]
3939
3945 # conclusion and future study
3946
3947 High_Crime <- c("Albuquerque",
3948
                 "Anchorage",
                 "Baltimore",
3949
                 "Chicago",
"Detroit",
3950
3951
3952
                 "Houston",
3953
                 "Lake_Charles",
                 "Little_Rock",
3954
                 "Memphis",
3955
3956
                 "Myrtle_Beach",
                 "Nashville",
3957
3958
                 "New_Orleans",
                 "St_Louis")
3959
3960 High_Crime
3961
3962 # original crime variables
3963
3964 CRIME_2018_ASSIGN[CRIME_2018_ASSIGN$CITY == High_Crime, 2:8]
3965
3966 # y1, y2, y3
3967
3968 CRIME_2018_ASSIGN[High_Crime,9:11]
```