



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

2014

Building a Predictive Model for Baseball Games

Jordan Robertson Tait
Minnesota State University - Mankato

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

 Part of the [Mathematics Commons](#), [Numerical Analysis and Computation Commons](#), and the [Statistics and Probability Commons](#)

Recommended Citation

Tait, J. R. (2014). Building a Predictive Model for Baseball Games [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/382/>

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.

Building a Predictive Model for Baseball Games

by

Jordan Tait

A Thesis Submitted in Partial Fulfillment of the Requirements for

Masters of Science

In Mathematics and Statistics

Minnesota State University, Mankato

Mankato, Minnesota

December 2014

December 2014

Building a Predictive Model for Baseball Games

Jordan Tait

This thesis paper has been examined and approved by the following members of the thesis committee.

Examining Committee:

Dr. In-Jae Kim, Advisor

Dr. Deepak Sanjel

Dr. Han Wu

Acknowledgements

I would like to extend my utmost gratitude to Dr. In-Jae Kim for his support and guidance in completing this Thesis.

I would like to thank James R. Broomfield for his originally produced code which provided inspiration and structure for the code used in this paper. I would also like to thank Benjamin D. Sencindiver and Jacob O. Westman for their contributions to the research that went into this Thesis.

Building a Predictive Model for Baseball Games

TAIT, JORDAN ROBERTSON M.S. IN MATHEMATICS AND STATISTICS,
MINNESOTA STATE UNIVERSITY, MANKATO, MINNESOTA, DECEMBER
2014

Abstract. In this paper, we will discuss a method of building a predictive model for Major League Baseball Games. We detail the reasoning for pursuing the proposed predictive model in terms of social popularity and the complexity of analyzing individual variables. We apply a coarse-grain outlook inspired by Simon Dedeos' work on Human Social Systems, in particular the open source website Wikipedia [2] by attempting to quantify the influence of winning and losing streaks instead of analyzing individual performance variables. We will discuss initial findings of data collected from the LA Dodgers and Colorado Rockies and apply further statistical analysis to find optimal betting points using a coarse-grain approach. We will apply Bayes' Theorem to add predictive power to a naive model using winning and losing streaks. We will discuss possible shortcomings of the proposed using Bayes' approach and address the question as to whether or not baseball wins and losses can be produced using a random process.

Table of Contents

1	Introduction	1
1.1	Human Social Systems and Coarse-graining	2
2	Coarse Grain Approach	4
2.1	Linear Discriminant Analysis	4
2.1.1	Methodology and the Curse of Dimensionality	5
2.1.2	Results	5
2.2	Coarse-graining Data	7
2.3	Optimal Betting Point and Worse Case Scenario	8
3	Exploratory Analysis	10
3.1	Initial Findings: L.A. Dodgers	11
3.2	Further Analysis: Conditional Probability	13
3.2.1	Conditional Probability Analysis	17
4	Building a Predictive Model using Bayesian Approach	20
4.1	Bayes' Theorem	20
4.2	Maximizing Probability using Bayes' Theorem	22

5 Simulations and Conclusion	26
5.1 Simulating Random Strings	26
5.2 Coda	30
6 Appendix: R code	32
6.1 Code for Linear Discriminant Analysis in section 2.1.2	32
6.2 Code for Logistic Regression in section 2.1.2	35
6.3 Tabulating the frequency of LW^kL	37
6.4 Simulating Random Strings and creating Confidence Intervals in Chapter 5	39
Bibliography	48

Chapter 1

Introduction

It is undeniable that prediction and the act of betting is deeply rooted in human nature. The inherent thrill of successfully predicting the outcome of an event can be traced throughout human history. The sense of satisfaction after a correct prediction coupled with the often simplistic layout of possible outcomes make for an irresistible urge to partake. Of all possible forms of gambling in todays society, sports betting has been one of the most popular forms within the last century. The seemingly unpredictability of sporting events has undoubtedly assisted in it's rise in popularity within the betting community. The numerous variables, independent and dependent, involved in many sporting events contribute to the unpredictability. Baseball has been one of the most popular sporting events when it comes to gambling due to the many variables involved and the enormous amount of data available. One of the goals of this paper is to take a coarse-grain look on baseball game outcomes in order to create a predictive model that is primarily based on winning and losing streaks, as opposed to individual players performance variables. This coarse-grain outlook was inspired by Simon DeDeo's work on Collective Phenomena and Non-Finite State Computation in a Human Social System [2]. In the following section, background information will be provided detailing the Human social Systems and coarse-graining.

1.1 Human Social Systems and Coarse-graining

Human social systems exist in nearly every facet of everyday life. Knowing how these human social systems process information would allow for a rise in predictive power in areas such as economics [3], the formation of opinions and spread of representational information in organizations [4] or the propagation of ideas through a social network [1]. Human social systems tend to maintain a high level of complexity which allows them to process information on par with (and often better than) well engineered systems [2]. The high level of complexity tends to make understanding the structure of a social system difficult, hence lowering predictive power.

In his research, Simon DeDeo attempts to describe how social systems process information. Simon studied the particular phenomena of cooperation of the open source Wikipedia community with the goal of distinguishing between different classes of computational sophistication.

In order to gain insight of the cooperation within the Wikipedia community, he studied the edits made to Wikipedia pages by the Wikipedia community. As previously stated, brute force analysis of these edits could become quite complex due to a few factors. One such factor is that the number of possible edits that editors make is essentially unbounded. Another factor is that each edit may change, add and/or delete arbitrary amounts of text from the page being analyzed. These factors are examples of data at the individual level.

To avoid this computational complexity, it is natural to partition the possible edits into a manageable number of classes in order to perform the analysis. A dichotomy conveniently already exists between the types of possible edits; those that change the

Time	Type of Edit
02:01	C
02:30	R
07:12	C
12:21	C
18:06	R
23:43	C

Table 1.1: The above is a fictional example of a subset of edits. ie: the second entry refers to a revert that was made at 2:30.

current text and those that revert the text to the previous state. The latter action is called a *revert* while any other action is referred to as a non-revert or simply, a *collaboration*. Thus, the entire history of a pages edits can be partitioned into two disjoint classes, reverts (R) and collaborations (C). Since the edits are time dependent, a fictional yet useful example of a subset of edits is shown in Table 1.1.

The sequence of events in the table can be more conveniently expressed as a string of the types of edits in chronological order from left to right: “*CRCCRC*”. Simon focused on the most edited pages since they would provide the greatest amount of coarse-grain data while holding the page constant. A full string of edits could be much longer than that shown in Table 1.1 but would still be within the realm of computationally manageable. In the following chapter, we will discuss how this idea of coarse-graining will be adopted and applied to making a predictive model as well as discussing the possible outcomes. Chapter 3 will discuss the findings of the initial exploratory analysis and possible shortcomings.

Chapter 2

Coarse Grain Approach

The rise of gambling on baseball can be attributed to many factors, which include the collection of numerous variables that play roles in a games outcome and the massive amount of data available. Although the number of possible combinations of variables is very large, the amount of completed research and the number of previously constructed models both are anything but small. Variables such as home runs, runs scored, batting average and pitching ERA (a pitchers earned run average per game) have all been beaten by the test of time as proving to be insufficient at routinely predicting outcomes of baseball games. We will construct a model using *Linear Discriminant Analysis* to show these inconsistencies as far as predicting outcomes for baseball games. Afterwards, we will attempt to capture the influence of winning and losing streaks pertaining to predicting the outcome of baseball games.

2.1 Linear Discriminant Analysis

In this section, we will discuss Linear Discriminant Analysis and apply this method in an attempt to create a predictive model using individual performance variables. We will test the model and show the inability to predict wins and losses.

2.1.1 Methodology and the Curse of Dimensionality

Complexity of a system often rises as the number of features (dimensions) rises. This phenomena is often known as the *Curse of Dimensionality* [12]. It is useful to note that some features in a particular system can be more useful for prediction than others. The idea of focusing on more useful features is part of *dimension reduction*. In our case, we select only forty nine variables (dimensions) that we think are more relevant to predicting an outcome of a baseball game than others. Unlike many other methods, Linear Discriminant Analysis is a method of classification meaning it uses predictor variables to classify an outcome, not predict a numerical value. This is ideal in our case since our focus is the prediction of a win or loss, not a numerical value.

2.1.2 Results

The code for running Linear Discriminant Analysis in R can be found in the appendix section 6.1. After applying the Linear Discriminant Analysis on the LA Dodgers data, we will test this model by predicting the outcomes of a set of games within the span of the games played and compare the predictions with the actual outcomes. The results can be found in Table 2.1. From a set of 1000 games, the predictive model successfully predicted the outcomes of 551 games, incorrectly predicting a loss 269 times and incorrectly predicting a win 161 times. This particular model produces a success rate of $\frac{551}{1000} = 0.551$, leaving us with no competitive edge and very little predictive power.

We performed a similar analysis on the Colorado Rockies, applying Linear Discriminant Analysis on the Colorado Rockies data for the same variables and tested

	Predicted Loss	Predicted Win
Actual Loss	198	161
Actual Win	269	353

Table 2.1: The table above contains the prediction for the LA Dodgers using the Linear Discriminant Analysis on a set of 1000 games within the 1782 games. The diagonal entries represent correct predictions while the off diagonal entries represent incorrect predictions.

	Predicted Loss	Predicted Win
Actual Loss	369	241
Actual Win	158	210

Table 2.2: The table above contains the prediction for the Colorado Rockies using the Linear Discriminant Analysis on a set of 1000 games played by the Rockies. The diagonal entries represent correct predictions while the off diagonal entries represent incorrect predictions.

the model on a set of the games played by the Colorado Rockies. The results are in Table 2.2. The predictive model successfully predicted the outcomes of 579 games, incorrectly predicting a loss 158 times and incorrectly predicting a win 241 times. This particular model produces a success rate of $\frac{579}{1000} = 0.579$, leaving us with no competitive edge and very little predictive power.

We also used logistic regression to predict the outcomes using the same set of forty nine variables and the results are similar to those of Linear Discriminant Analysis. The R code for Logistic Regression may be found in appendix section 6.2.

2.2 Coarse-graining Data

Inspired by Simon Dedeo [2], we will use the idea of coarse graining instead of analyzing the individual level information. In our case, the individual level information pertains (but is not limited) to the performance of individual players, the interactions of players, the interactions between teams and the interactions between players and factors outside of the baseball community. Our goal is to capture the phenomena of winning and losing via the collective coarse-grain information of winning and losing streaks.

As described in section 1.1, the coarse-grain information of Wikipedia edits can be conveniently listed as strings of R's and C's, referring to *reverts* and *collaborations*. As wins and losses are also time dependent, we will conveniently list a series of winning and losing outcomes for a particular baseball team as a string of concatenated W's and L's in chronological order from left to right. A useful example of a subset of wins and losses is shown in Table 2.3. The information in Table 2.3 can be represented as the concatenated string "WLWLLL". The following notation will further facilitate handling such strings and finding a predictive model.

Notation. Let k be a positive integer. Then a winning streak of length k can be denoted as LW^kL , i.e.,

$$\dots LWWWLWLWWL\dots \iff \dots LW^3LW^1LW^2L\dots \quad (2.2.1)$$

Similarly, a losing streak of length k can be denoted as WL^kL , e.g.,

$$\dots WLLWLLLLLWLW\dots \iff \dots WL^2WL^4WL^1W\dots \quad (2.2.2)$$

Date	Outcome
March 31	W
April 1	L
April 2	W
April 3	L
April 4	L
April 5	L

Table 2.3: The above is an example of a subset of wins and losses from the LA Dodgers in 2003, e.g., the second entry refers to a Loss on April 1st.

It is useful to note how to count a winning or losing streak if they occur at the end of a string. For example,

$$LLWWW\dots LWWWW \longleftrightarrow L^2W^3\dots L^1W^4(L) \quad (2.2.3)$$

In which case, we say there exists a winning streak of length four at the end of the string even though the fourth win is not followed by a loss.

2.3 Optimal Betting Point and Worse Case Scenario

Our goal of gaining predictive power can be equivalently stated as being able to predict the outcome of a given trial with a large probability of success. Since we are not biased towards one outcome or the other (a particular team winning or losing), it is sufficient to search for a particular case where the probability of successfully predicting the outcome of a teams' performance, win or loss, is large. This particular case will be referred to as an *optimal betting point*. In regards to finding an optimal betting point, the worst case scenario occurs when the chance of winning a bet (

successfully predicting the outcome of a teams performance) is 50%. In terms of winning streaks (or losing streaks), this would refer to the case when k -game winning streaks decreases by a factor of 2 as the value of k increases.

For example, let the number of 1-game winning streaks be 2^n . The worst case scenario would occur if the number of 2-game winning streaks is 2^{n-1} . Since $2^n + 2^{n-1} + \dots + 2 + 1 = 2^{n+1} - 1$, the chance of winning a bet on loss for the game after the first win would be:

$$\frac{2^n}{2^n + 2^{n-1} + \dots + 2 + 1} = \frac{2^n}{2^{n+1} - 1} = \frac{1}{2 - (\frac{1}{2})^n} \quad (2.3.1)$$

$$\lim_{n \rightarrow \infty} \frac{1}{2 - (\frac{1}{2})^n} = \frac{1}{2} \quad (2.3.2)$$

If the number of k -game winning streaks decreases by a factor of 2 as the value of k increases, equation 2.3.2 implies the chance of a successful prediction is about 50%. Since the probability, in this case, of making a successful bet is no better than making a random bet, this is the worst case scenario. In the next chapter, we will discuss the findings of the initial exploratory analysis.

Chapter 3

Exploratory Analysis

In Baseball, as in many sports, the individual variables and their interactions exhibit highly irregular behavior. Each individual player's performance may change drastically from game to game or year to year. Yet, each individual player's performance is dependent on many internal and external factors that are changing from game to game and year to year.

The constant interaction of these numerous dependent variables is only one fold of difficulty, the second being the fairly frequent turnover of players. Each baseball team is allowed 25 players to be on the active roster. The active roster is for players who dress for each game and are allowed to play each game. Each team is also allowed 15 more players that, together with the active roster, comprise the expanded roster. The 15 additional players do not dress for each game but can be used as substitutes for any player on the active roster. Each year players are traded, players retire and new players are added to the roster. These transactions also occur between offseasons. This frequent turnover adds to the motivation for using coarse-graining.

3.1 Initial Findings: L.A. Dodgers

In order for our predictive model to successfully predict current outcomes, it should be built from the most current data that represents the current teams configuration. However, a large sample size is needed in order to capture possible present phenomena. In our exploratory analysis, we examine the game outcomes from the LA Dodgers from 2003-2013 [11]. The data is then coarse-grained and the strings of wins and losses are recorded. An arbitrary baseball team plays 162 games each year, often called the *regular season*. The regular season usually starts on the first Sunday in April and ends on the first Sunday in October. After the conclusion of the regular season, the post season, often referred to as the *playoffs*, begin. The post season is a tournament style series of games between ten selected teams. The last game of the post season is referred to as the *World Series* and is a best-of-seven game series between the top two teams used to declare the best team in the league. Since the post season does not include every team, we will analyze regular season data. Thus the coarse-grained data from 2003-2013 results in a string 1782 elements.

As it is noted in Table 3.1, roughly 80% of winning streaks of length k are of length three or less. This phenomena of the number three can be linked to many facets within and outside of baseball. There exist three primary colors that can produce the known color spectrum. Humans are often said to be comprised of three parts; the mind, the body and the soul. In the National Hockey League, teams decide a winner by competing for sixty minutes comprised of three twenty minute periods. Even within Baseball, a batter receives three strikes every time he faces the opposing pitcher before being declared “out” while each team must achieve three defensive

LW^kL	Frequency	Relative Frequency
$k = 1$	213	0.490
$k = 2$	103	0.237
$k = 3$	54	0.124
$1 \leq k \leq 3$	370	0.851
$k = 4$	27	.062
$k = 5$	13	0.030
$k = 6$	11	0.025
$k = 7$	4	0.009
$k = 8$	6	0.014
$k = 9$	1	0.002
$k = 10$	2	0.005
$k = 11$	1	0.002
$k = 12$	0	0

Table 3.1: The above table contains the frequencies and relative frequencies of winning streaks of length k for the LA Dodgers

“outs” before they get their chance to go on the offensive.

The phenomena of 80% can also be linked to many topics outside of baseball. The *Pareto Principle*, also known as the 80 – 20 rule, states that roughly 80% of effects come from roughly 20% of the causes. In economics, it is often observed that roughly 20% of given populations own roughly 80% of the entire populations income [10]. In Mathematics, the 80 – 20 rule can be represented by a *power law distribution*, also known as a *Pareto Distribution* [5]. In computer science, Microsoft noted that when 20% of the most frequently reported bugs were fixed, 80% of the related errors in a system would be eliminated [6]. In occupational health and safety, it is common practice to assume that 20% of the possible hazards cause 80% of injuries [8], allowing for more directed targeting of hazards. An example of a Pareto Distribution is shown in Figure 3.1

This 80% phenomena unraveled in the initial exploratory analysis can be naively

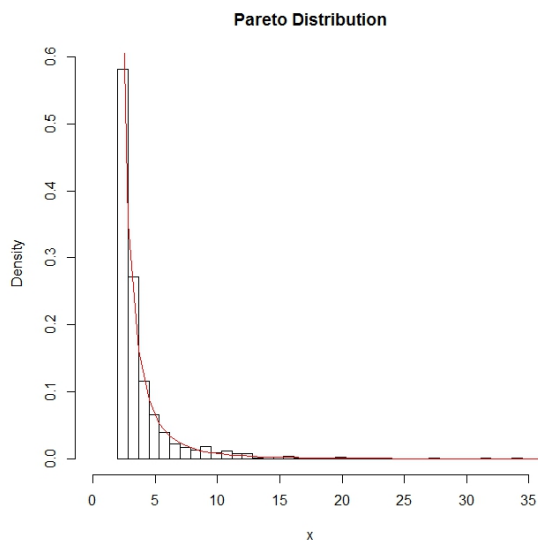


Figure 3.1: Above is an example of a Pareto Distribution.

interpreted as the existence of an 80% chance to successfully predict a loss for the LA Dodgers after they have won three games in a row, ending their three game winning streak. In the following section, we will discuss the validity of this naive interpretation.

3.2 Further Analysis: Conditional Probability

To test the validity of our initial findings, we will develop and use some statistical tools. The probability of an event, say B , occurring given the knowledge that an event, say A , has already occurred is referred to as the *conditional probability of event B* . The following definitions will prove to be useful in describing the post-initial analysis.

Definition. The *sample space* of an experiment or random trial is the set of all

possible outcomes of that experiment or random trial.

Definition. A *simple event* is a single possible outcome in the sample space of an experiment or random trial.

Definition. An *event* is a set of outcomes, a subset of the Sample space, of an experiment or random trial.

Definition. If A and B are events in a sample space S , then events A and B are said to be *mutually exclusive* if they share no common simple events.

Definition. For events A and B , the probability of the *intersection of A and B* is defined by:

$$P(A \cap B) = P(A) \cdot P(B|A) \quad (3.2.1)$$

From this definition, the *conditional probability* $P(B|A)$ can be obtained to find the following:

$$P(B|A) = \frac{P(A \cap B)}{P(A)} \quad (3.2.2)$$

If it is known that an event A has occurred, this can be interpreted as a shrinking of the sample space to only the simple events contained within the event A . This

interpretation will prove to be very useful in our application. It is also worth noting that if events A and B are independent, then knowledge of knowing event A has occurred does not affect the probability of event B occurring, hence the following holds:

$$P(B|A) = P(B) \tag{3.2.3}$$

$$P(A \cap B) = P(A) \cdot P(B) \tag{3.2.4}$$

In our case, we will let A be the event that a team achieves a winning streak of length k , losing the next game. Letting B be the event that a team achieves a winning streak of at least length k , we can express A and B as the following:

$$A = LW^kL \tag{3.2.5}$$

$$B = LW^k \tag{3.2.6}$$

We will use these defined events to observe the restricted sample space of instances where only event B occurs to find the conditional probability $P(A|B)$, the probability of losing the $k + 1$ game after winning k games in a row.

Although the following relationship is generally not true, in our case the intersection of events A and B results in exactly the event A . Hence, the following holds

$$P(A \cap B) = P(A) \tag{3.2.7}$$

We are then left with

$$P(A|B) = \frac{P(A \cap B)}{P(B)} = \frac{P(A)}{P(B)} = \frac{P(LW^k L)}{P(LW^k L) + \sum_{j \geq 0} P(LW^{k+j} L)} \quad (3.2.8)$$

The following notation gives an alternative way to express the desired probability,

Definition. If A is an event, then let the *frequency* of event A be represented by the following notation

$$n(A) = \text{frequency of event } A$$

Hence the equation 3.2.8 can be rewritten as

$$P(A|B) = \frac{n(LW^k L)}{\sum_{j \geq k} n(LW^j L)} \quad (3.2.9)$$

We can directly apply (3.2.9) to the data in Table 3.1 by defining the following events

$$A = LW^3 L \quad (3.2.10)$$

$$B = LW^j \quad (3.2.11)$$

for $j = 3$. Hence, we have

$$P(A|B) = \frac{n(LW^3 L)}{\sum_{j \geq 3} n(LW^j L)} = \frac{54}{119} \approx 0.4538 \quad (3.2.12)$$

Contrary to the naive interpretation in section 3.1, we aren't left with any compet-

itive edge when trying to predict the outcome of the fourth game for the LA Dodgers given the knowledge that the LA Dodgers have already won three games in a row.

3.2.1 Conditional Probability Analysis

Since we do not gain predictive power via the method of analyzing streak lengths when betting on the fourth game after a three game winning streak, a logical question is to ask if there exists optimal betting points for some other events than the events defined in (3.2.10) and (3.2.11). In order to gain perspective on possible optimal betting points, we use equation (3.2.9) to find the conditional probabilities for the possible events. These conditional probabilities can be found in Tables 3.2 and 3.3 below.

In order to reveal more possible optimal betting points, similar analysis is performed regarding losing streaks; these results can be found in Tables 3.4 and 3.5 below. Note that in both cases of winning streaks and losing streaks there lacks existence of an optimal betting point for winning a bet for streaks of length three. Intriguingly, there is existence of optimal betting points. In the following chapter, we will discuss a Bayesian approach to maximize the chances of winning a bet regarding these existing optimal betting points.

k	$n(LW^kL)$	$\sum_{j \geq k} n(LW^jL)$	$\frac{n(LW^kL)}{\sum n(LW^jL)}$
1	210	390	0.538
2	84	180	0.215
3	48	96	0.123
4	28	48	0.072
5	12	20	0.031
6	2	8	0.005
7	2	6	0.005
8	1	4	0.003
9	0	3	0
10	1	3	0.003
11	2	2	0.005

Table 3.2: The above table contains winning streak data from the Colorado Rockies [11] applying an earlier definition and equation (3.2.9)

k	Cumulative Relative Frequency	$\frac{n(LW^kL)}{\sum_{j \geq k} n(LW^jL)}$	$1 - \frac{n(LW^kL)}{\sum_{j \geq k} n(LW^jL)}$
1	0.538	0.538	0.462
2	0.754	0.0467	0.533
3	0.877	0.5	0.5
4	0.949	0.583	0.417
5	0.979	0.6	0.4
6	0.985	0.25	0.75
7	0.990	0.333	.0667
8	0.992	0.25	0.75
9	0.992	0	1
10	0.995	0.333	0.667
11	1	1	0

Table 3.3: The above table is a continuation of Table 3.2. Note that column three is the probability of winning a bet on L given a winning streak of length at least k and the last column is the probability of winning the bet on W given a winning streak of length at least k .

k	$n(WL^kL)$	$\sum_{j \geq k} n(WL^jW)$	$\frac{n(WL^kL)}{\sum n(WL^jW)}$
1	210	390	0.538
2	84	180	0.215
3	48	96	0.123
4	28	48	0.072
5	12	20	0.031
6	2	8	0.005
7	2	6	0.005
8	1	4	0.003
9	0	3	0
10	1	3	0.003
11	2	2	0.005

Table 3.4: The above table contains losing streak data from the Colorado Rockies applying an earlier definition and equation (3.2.9)

k	Cumulative Relative Frequency	$\frac{n(LW^kL)}{\sum_{j > k} n(LW^jL)}$	$1 - \frac{n(LW^kL)}{\sum_{j > k} n(LW^jL)}$
1	0.538	0.538	0.462
2	0.754	0.0467	0.533
3	0.877	0.5	0.5
4	0.949	0.583	0.417
5	0.979	0.6	0.4
6	0.985	0.25	0.75
7	0.990	0.333	.0667
8	0.992	0.25	0.75
9	0.992	0	1
10	0.995	0.333	0.667
11	1	1	0

Table 3.5: The above table is a continuation of Table 3.4. Note that column three is the probability of winning a bet on W given a losing streak of length at least k and the last column is the probability of winning the bet on L given a losing streak of length at least k .

Chapter 4

Building a Predictive Model using Bayesian Approach

In this chapter, we attempt to maximize the chance of winning a bet. In order to do so, we will develop a *Bayesian Approach* to the events discussed in section 3.2.1. A traditional Bayesian approach can be interpreted as an updating of the chance of an event occurring. This method effectively recalculates the probability of an event occurring after taking prior evidence into account.

4.1 Bayes' Theorem

Recall the probability of an event B occurring given the knowledge that an event A has already occurred is referred to as the *conditional probability of event B* . The *conditional probability of event B* and the *intersection of A and B* are linked by the following relationships

$$P(A \cap B) = P(A) \cdot P(B|A) \tag{4.1.1}$$

$$P(B|A) = \frac{P(A \cap B)}{P(A)} \tag{4.1.2}$$

Similarly, the *conditional probability of event A* and the *intersection of A and B* are linked by

$$P(A|B) = \frac{P(A \cap B)}{P(B)} \quad (4.1.3)$$

Rearranging equations (4.1.2) and (4.1.3) we are left with two ways to express the intersection of events A and B .

$$P(A \cap B) = P(A|B) \cdot P(B) = P(B|A) \cdot P(A) \quad (4.1.4)$$

The latter part of equation (4.1.4) is often called *Bayes Rule* and can be written the following ways

$$P(A|B) \cdot P(B) = P(B|A) \cdot P(A) \quad (4.1.5)$$

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)} \quad (4.1.6)$$

The sample space S can be partitioned in terms of mutually exclusive and exhaustive events E_1, E_2, \dots, E_n (that is $P(E_i \cap E_j) = 0, i \neq j$; $\cup_{i=1}^n E_i = S$) and the partition can be used to express the probability of event B as follows

$$P(B) = \sum_{i=1}^n P(B|E_i) \cdot P(E_i) \quad (4.1.7)$$

Definition. If A is an event contained in a sample space S , then the *complement of event A* is the accumulation of events included in S but not in A . The *complement of an event A* is denoted as A^c .

Note that an event A and its complement A^c are mutually exclusive and exhaustive. We can use equation (4.1.7) to express equation (4.1.6) as

$$P(E|B) = \frac{P(B|E) \cdot P(E)}{\sum_{i=1}^n P(B|E_i) \cdot P(E_i)} \quad (4.1.8)$$

In terms of A and A^c , we are left with a traditional form of *Bayes' Theorem*

Theorem 4.1.1. [9, Equation 1.4.1] *The probability of an event A , given that an event B has subsequently occurred is*

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B|A) \cdot P(A) + P(B|A^c) \cdot P(A^c)} \quad (4.1.9)$$

This theorem gives us a way of updating the probability of an event A with given information, event B . In the following section, we will discuss how we can use Bayes' Theorem to possibly give us more predictive power.

4.2 Maximizing Probability using Bayes' Theorem

In this section, we will develop a technique using Bayes' Theorem. This technique can influence the probability of existing betting points, making them more optimal if used strategically. We will start with Theorem (4.1.1) and apply traditional maximizing techniques, then link the general situation with our particular situation.

Consider a function of the following form

$$f(x) = \frac{g(x)}{g(x) + h(x)}, \quad (4.2.1)$$

where $g(x)$ is positive and $h(x)$ is nonnegative.

If the goal were to find a value x making $f(x)$ maximized, it would be sufficient to find a value x where $\frac{h(x)}{g(x)}$ attains its minimum. As the value of $\frac{h(x)}{g(x)}$ approaches zero, the value of $f(x)$ approaches one.

In our case, our goal is to maximize the conditional probability $P(A|B)$ expressed as

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B|A) \cdot P(A) + P(B|A^c) \cdot P(A^c)} \quad (4.2.2)$$

We can exhibit a similar technique as maximizing equation (4.2.1) and find an event B such that the value $P(B|A^c) \cdot P(A^c)$ is minimized. We then would be left with

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B|A) \cdot P(A) + P(B|A^c) \cdot P(A^c)} \longrightarrow \frac{P(B|A) \cdot P(A)}{P(B|A) \cdot P(A)} = 1 \quad (4.2.3)$$

as $\frac{P(B|A^c) \cdot P(A^c)}{P(B|A) \cdot P(A)}$ approaches zero.

For example, let A be the event that we get a loss after a winning streak of six, producing LW^6L . Let B be the event that a game is played in either June or July. Using the data collected from the Colorado Rockies, $P(A) = 0.25$, $P(B|A) = 1$ and $P(B|A^c) = 0.167$. Applying Theorem (4.1.1), we find

$$P(A|B) = \frac{(1) \cdot (0.25)}{(1) \cdot (0.25) + (0.167) \cdot (0.75)} = 0.667 \quad (4.2.4)$$

Although this does not result in an optimal betting point, it successfully demonstrates the ability of Bayes' Theorem to update the probability of an event occurring given additional information, event B .

We next consider another pair of events A and B to update the conditional probability $P(A|B)$ using a strategic selection of an event B . Consider the event A that we get a win after a winning streak of length six, producing LW^6W or equivalently LW^7 . Let B be the event that a game is played in neither June nor July. From the Colorado Rockies data, $P(\text{Neither June nor July}|LW^6L) = P(B|A^c) = 0$. Applying Theorem (4.1.1), we find the following

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B|A) \cdot P(A) + P(B|A^c) \cdot P(A^c)} = \frac{P(B|A) \cdot P(A)}{P(B|A) \cdot P(A)} = 1 \quad (4.2.5)$$

Given the Colorado Rockies are playing in neither June nor July, the probability of winning a seventh game given they have already won six games in a row, is one.

The idea to strategically pick B as a particular time of the regular season was derived from the common conception that certain sports teams have more wins in certain times of the regular season. It is observed in the data that the Colorado Rockies were much more likely to continue their winning streak if the seventh game was played in neither June nor July, that is, early on in the regular season or near the end of the regular season.

In the final chapter, we will discuss whether or not the strings of wins and losses can be produced by a random process. We will also recap what has been accomplished in this paper and discuss possible follow up research.

Chapter 5

Simulations and Conclusion

It was found in the Section 3.2.1 that there were many non-advantageous betting points. We were able to apply a Bayesian Approach in Chapter 4 to improve existing optimal betting points given prior information. However, because of the nature of data which are coarse grained strings of wins and losses we can ask the question of whether or not baseball wins and losses can be predicted via a random process.

5.1 Simulating Random Strings

To help answer this question, we will create random processes that will produce strings of ones and zeros, representing wins and losses respectively. For the Dodgers, each element will be represented with a one with probability 0.525 and zero with probability 0.475. For the Rockies, each element will be represented with a one with probability 0.469 and zero with probability 0.531. These probabilities represent the historical relative frequencies of wins and losses for the Dodgers and Rockies.

We will look particularly at the frequency of the event $A = LW^kL$. The frequency of event A can be found in Tables 3.1 and 3.2 for the Dodgers and Rockies, respectively. Using multiple simulations, we will create confidence intervals for the average number of occurrences of the event $A = LW^kL$ at a 95% confidence level,

then compare them to the frequencies in Tables 3.1 and 3.2. We will need the following Theorem and definitions in order to appropriately create the desired confidence intervals. The code can be found in the Appendix.

Theorem 5.1.1. [9, Theorem 4.2.1] *Let X_1, X_2, \dots, X_n denote a random sample with sample mean \bar{X} and sample variance S^2 from a distribution that has mean μ and finite variance σ^2 . Then the variable $W = \frac{\bar{X} - \mu}{S/\sqrt{n}}$ converges to the Standard Normal Distribution as n approaches infinity.*

Definition. Let X_1, X_2, \dots, X_n be a random sample on a random variable X with mean μ and variance σ^2 . Then the following holds for sufficiently large n

$$1 - \alpha \approx P(-z_{\alpha/2} < \frac{\bar{X} - \mu}{S/\sqrt{n}} < z_{\alpha/2}), \quad (5.1.1)$$

which is algebraically equivalent to the following

$$1 - \alpha \approx P(\bar{X} - z_{\alpha/2} \cdot S/\sqrt{n} < \mu < \bar{X} + z_{\alpha/2} \cdot S/\sqrt{n}) \quad (5.1.2)$$

The above is referred to as a $(1 - \alpha)\%$ *Confidence Interval for the population mean μ .*

Applying Theorem (5.1.1) and equation (5.1.2) to output from the simulations, the confidence intervals for the population mean frequency of $A = LW^2L$ were found to be the following.

$$(105.353, 112.247) \quad (5.1.3)$$

Equation (5.1.3) represents a 95% confidence interval for the population mean fre-

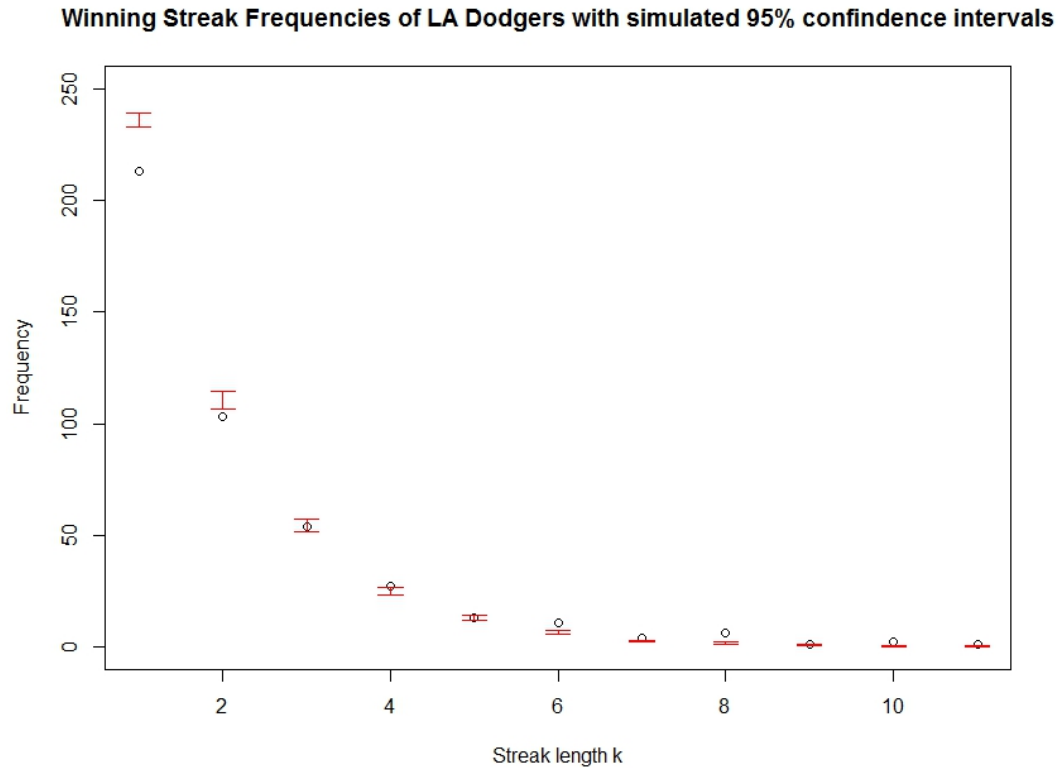


Figure 5.1: Above is a graph with the actual frequencies of LW^kL of the LA Dodgers from Table 3.1 with confidence intervals created from the simulations for the LA Dodgers discussed above.

quency of $A = LW^2L$ for the LA Dodgers.

$$(97.0291, 102.6709) \tag{5.1.4}$$

Equation (5.1.4) represents a 95% confidence interval for the population mean frequency of $A = LW^2L$ for the Colorado Rockies.

Similarly constructed confidence intervals are shown in figures 5.1 and 5.2.

Although the generated strings are visually close to the actual data, there are values of k for LW^kL where the observed data are not close to the randomly generated

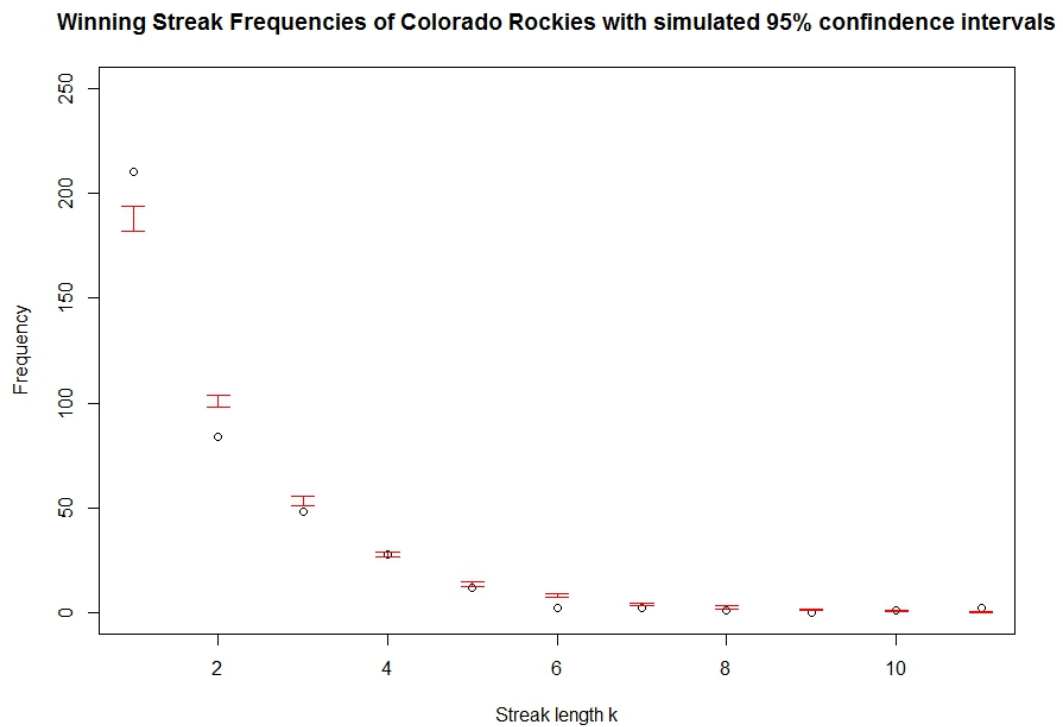


Figure 5.2: Above is a graph with the actual frequencies of LW^kL of the Colorado Rockies from Table 3.2 with confidence intervals created from the simulations for the Colorado Rockies discussed above.

data and where there lacks a general pattern for particular values of k .

5.2 Coda

In this subsection, we will recap what has been discussed in this paper, possible shortcomings and potential further research.

Thus far, we have used Linear Discriminant Analysis on individual performance variables to show inconsistencies of predicting baseball game outcomes using said variables. We then adopted a coarse graining technique in an attempt to create a predictive model for baseball games. Instead of looking at individual information, we analyzed the coarse-grained information of wins and losses as a concatenated string of W's and L's.

We initially found that roughly 80% of all streaks are of length three or less. We then used conditional probability to find that the chance of winning a bet on a fourth game given a winning streak of length three was not an optimal betting point. We did in fact find optimal betting points. We were able to successfully take a Bayesian approach in order to improve the existing optimal betting points by strategically selecting specific events. We raised the question as to whether or not wins and losses could be produced using a random process. We were able to give some insight regarding this question by simulating win and loss strings to create appropriate confidence intervals for the population mean of the events $A = LW^kL$.

Possible shortcomings and topics of follow-up research for application should also be discussed. The predictive models we have discussed were constructed from the data of Colorado Rockies and LA Dodgers, hence the optimal betting points that we

have found are specific for these two teams only. Predictive models for other teams could be built in a similar fashion but it is likely that optimal betting points for other teams could be different than those found for the Colorado Rockies and LA Dodgers. Comparing models between teams, as well as optimal betting points could be worthwhile future research.

It should be noted that the optimal betting points for the LA Dodgers and Colorado Rockies are rare and hence the opportunities to place winning bets may only occur a few times over upcoming seasons.

This type of predictive model could be extended outside of Baseball, as the idea of winning and losing streaks is not unique to Baseball. Our focus was directed toward Baseball partially due to the amount of readily available data. Along this line, sports such as Basketball, Soccer, Handball or Cricket could be possible future research topics.

Chapter 6

Appendix: R code

In the appendix, the R codes used in the analysis and simulations of this paper are provided below.

6.1 Code for Linear Discriminant Analysis in section 2.1.2

```
install.packages("PASWR")  
  
library(PASWR)  
  
# Some Variable names are duplicated between offensive and defensive statistics  
# For said variables, "def" is concatenated to the end of the  
# defensive variable to distinguish between offense and defensive  
rock<-read.csv("C:\\Users\\Jordan\\Jordan's Viao\\School\\  
Thesis_F14\\LDA Data\\Rockies_LDA_Data.csv", header = TRUE)  
attach(rock)  
  
rockiesLDA<-lda(Class~PA+AB+R+H+doubles+triples+HR+RBI+BB+IBB
```

```

+SO+HBP+SH+SF+ROE+GDP+SB+CS+BA+OBP+SLG+OPS
+LOB+IP+Hdef+Rdef+ER+UER+BBdef+S0def+HRdef+HBPdef+ERA+
BF+Pit+Str+IR+IS+SBdef+CSdef+ABdef+doublesdef+triplesdef+
IBBdef+SHdef+SFdef+ROEdef+GDPdef,data=rock)

# rock[c(2:1002),c(2:49)] indicates the range of data to attempt
# prediction with. Hence the following line attempts to predict
# the class variable using rows 2 to 1002, 1000 games
# and columns 2 to 49, the variables minus the "class" variable

prediction<-predict(rockiesLDA,newdata=rock[c(2:1002),c(2:49)])$class

# this codes gives a tabluar visual of correctly
# predicted games.
# the diagonals represent correctly predicted outcomes
# the off diagonals represent incorrectly predicted games outcomes

table(prediction,rock[c(2:1002),1])

# The following is the corresponding code for the LA Dodgers

dodgers<-read.csv("C:\\Users\\Jordan\\Jordan's Viao\\School\\
Thesis_F14\\LDA Data\\Dodgers_LDA_Data.csv", header = TRUE)

```



```

attach(dodgers)

dodgersLDA<-lda(Class~PA+AB+R+H+doubles+triples+HR+RBI+BB+IBB
+SO+HBP+SH+SF+ROE+GDP+SB+CS+BA+OBP+SLG+OPS
+LOB+IP+Hdef+Rdef+ER+UER+BBdef+S0def+HRdef+HBPdef+ERA+
BF+Pit+Str+IR+IS+SBdef+CSdef+ABdef+doublesdef+triplesdef+
IBBdef+SHdef+SFdef+ROEdef+GDPdef,data=dodgers)

# dodgers[c(2:1002),c(2:49)] indidcates the range of data to attempt
# prediction with. Hence the following line attempts to predict
# the class variable using rows 2 to 1002, 1000 games
# and columns 2 to 49, the variables minus the "class" variable

prediction<-predict(dodgersLDA,newdata=dodgers[c(2:1002),c(2:49)])$class

# this codes gives a tabluar visual of correctly
# predicted games.
# the diagonals represent correctly predicted outcomes
# the off diagonals represent incorrectly predicted games outcomes

table(prediction,dodgers[c(2:1002),1])

```

6.2 Code for Logistic Regression in section 2.1.2

```

# Logistic Regression for LA Dodgers

dodgers<-read.csv("C:\\Users\\Jordan\\Jordan's Viao\\School\\
Thesis_F14\\LDA Data\\Dodgers_LDA_Data.csv", header = TRUE)

attach(dodgers)

# The following code performs a logistic regression proces
# on the 49 variables in the data for the LA Dodgers

dodgerslogit<-glm(Class~PA+AB+R+H+doubles+triples+HR+RBI+BB+IBB
+SO+HBP+SH+SF+ROE+GDP+SB+CS+BA+OBP+SLG+OPS
+LOB+IP+Hdef+Rdef+ER+UER+BBdef+S0def+HRdef+HBPdef+ERA+
BF+Pit+Str+IR+IS+SBdef+CSdef+ABdef+doublesdef+triplesdef+
IBBdef+SHdef+SFdef+ROEdef+GDPdef,data=dodgers,family="binomial")

# The following code gives the summary of the logist regression process

summary(dodgerslogit)

# The following code gives the odds ratios

```

```
# from the coefficients of regression from the above code

exp(coef(dodgerslogit))

# Logistic Regression for Colorado Rockies

rock<-read.csv("C:\\Users\\Jordan\\Jordan's Viao\\School\\
Thesis_F14\\LDA Data\\Rockies_LDA_Data.csv", header = TRUE)

attach(rock)

# The following code performs a logistic regression proces
# on the 49 variables in the data for the Colorado rockies

rockieslogit<-glm(Class~PA+AB+R+H+doubles+triples+HR+RBI+BB+IBB
+SO+HBP+SH+SF+ROE+GDP+SB+CS+BA+OBP+SLG+OPS
+LOB+IP+Hdef+Rdef+ER+UER+BBdef+S0def+HRdef+HBPdef+ERA+
BF+Pit+Str+IR+IS+SBdef+CSdef+ABdef+doublesdef+triplesdef+
IBBdef+SHdef+SFdef+ROEdef+GDPdef,data=rock,family="binomial")

# The following code gives the summary of the logist regression process

summary(rockieslogit)
```

```
# The following code gives the odds ratios
# from the coefficients of regression from the above code

exp(coef(rockieslogit))
```

6.3 Tabulating the frequency of LW^kL

```
# The following function calculates the frequencies
# of continuous strings of "ones" in a string of
# "ones" and "zeroes". The input, x, must be a string of "ones"
# and "zeros". In our case, the string of W's and L's
# are replaced with "ones" and "zeros", then inputed
# directly into the function below which outputs a vector w
# Where the ith entry in the vector w is the number of continuous
# strings of the number "one" of length i.
# There are print options pre-commented that can be un-commented
# in order to print the results at each step of the counting
# process.

streakfrequency<-function(x)
{
  L<-length(x)
  i<-1
```

```
y<-0
w<-rep(0,15)
while (i<L+1){
  if (x[i]==1){
    y<-y+1
    i<-i+1
    # print('y')
    # print(y)
    # print('i')
    # print(i)
  }
  else if (x[i]==0) {
w[y]<-w[y]+1
i<-i+1
y<-0
    # print('y')
    # print(y)
    # print('i')
    # print(i)
    # print('w')
    # print(w)
  }
}
return(w)
```

```
}
```

6.4 Simulating Random Strings and creating Confidence Intervals in Chapter 5

```
# This part of the code returns an array of numbers.
# The random variable used to generate each element in the array takes on
# values 1 or 0 with probability p and 1-p, respectively.
# For the LA Dodgers historically: p=0.525
# For the Colorado Rockies historically: p=0.469

random_string<-function(n)
{
a<-sample(c(0,1),n,prob=c(1-p,p),replace=TRUE)
return(a)
}

# This part of the code counts
# the number of win streaks of length k
# and returns them in a vector , w, where the ith
# entry in w corresponds to the number of winning
# streaks of length i.
```

```
#  
  
# There are optional print commands built into the code  
  
# in order to see the results from each step  
  
# in the counting process.  
  
#  
  
#  
  
randomwins<-function(x)  
{  
  L<-length(x)  
  i<-1  
  y<-0  
  w<-rep(0,15)  
  while (i<L+1){  
    if (x[i]==1){  
      y<-y+1  
      i<-i+1  
      # print('y')  
      # print(y)  
      # print('i')  
      # print(i)  
    }  
    else if (x[i]==0) {  
      w[y]<-w[y]+1
```

```
    i<-i+1
    y<-0
#   print('y')
#   print(y)
#   print('i')
#   print(i)
#   print('w')
#   print(w)
  }
}
return(w)
}

# returns a vector, y, of length "n"
# where "n" is the number of simulations
# to run and "m" is the length of each simulation
# the ith entry of y is the number of streaks
# of exactly length two in the ith simulation

samplewins<-function(n,m)
{
  i<-1
  y<-rep(0,n)
```



```
while (i<n+1){
a<-random_string(m)
b<-randomwins(a)
y[i]<-b[2]
i<-i+1
}
return(y)
}

# 30 simulations of random strings of length 1627
# length 1627 is used to compare to the rockies data, of which
# contains a string of 1627 elements.

rockiesvector<-samplewins(30,1627)

# sample size, n
n<-30

xbar<-mean(rockiesvector)
var<-var(rockiesvector)

# 95 percent CI for population mean of  $LW^2L$ 
# for Colorado Rockies

lower<-xbar-1.96*sqrt(var/n)
```

```
upper<-xbar+1.96*sqrt(var/n)

> lower

[1] 97.0291

> upper

[1] 102.5709

>

# 99 percent CI for population mean of LW^2L
# for Colorado Rockies

lower<-xbar-2.807*sqrt(var/n)
upper<-xbar+2.807*sqrt(var/n)

> lower

[1] 95.83167

> upper

[1] 103.7683

>

#30 simulations of random strings of length 1782
# length 1627 is used to compare to the Dodgers data, of which
# contains a string of 1627 elements.

dodgersvector<-samplewins(30,1782)
xbar<-mean(dodgersvector)
var<-var(dodgersvector)
```

```
# 95 percent CI for population mean of LW^2L
# for LA Dodgers
lower<-xbar-1.96*sqrt(var/n)
upper<-xbar+1.96*sqrt(var/n)
> > lower
[1] 105.353
> upper
[1] 112.247

# 99 percent CI for population mean of LW^2L
# for LA Dodgers
lower<-xbar-2.807*sqrt(var/n)
upper<-xbar+2.807*sqrt(var/n)
> >
> lower
[1] 103.8635
> upper
[1] 113.7365
>

# The following code computes 95% Confidence interval for
# streaks of # length K for the Rockies and Dodgers respectively
# k is the length of winning streaks
```

```
# n is the number of simulations to run  
# m is the length of each simulation
```

```
CItwoSDR<-function(k,n,m){  
  x<-samplewinsR(k,n,m)  
  xbar<-mean(x)  
  var<-var(x)  
  lower<-xbar-1.96*sqrt(var/n)  
  upper<-xbar+1.96*sqrt(var/n)  
  j<-c(lower,upper)  
  return(j)  
}
```

```
CItwoSDD<-function(k,n,m){  
  x<-samplewinsD(k,n,m)  
  xbar<-mean(x)  
  var<-var(x)  
  lower<-xbar-1.96*sqrt(var/n)  
  upper<-xbar+1.96*sqrt(var/n)  
  j<-c(lower,upper)  
  return(j)
```

```
}

# The following code plots the raw data
# with super imposed confidence intervals
# Note that the confidence interval limits must be manually
# entered for each streak length k
# This requires the package "gplots"

# Plot of the Colorado Rockies data with superimposed 95% Confidence intervals

actualrockies<-c(210,84,48,28,12,2,2,1,0,1,2)
x<-seq(1,11,1)
plot(x,actualrockies,ylim=c(0,250),xlab="Streak length k",ylab="Frequency",
main="Winning Streak Frequencies of
Colorado Rockies with simulated 95% confidence intervals")
arrows(x0=1,y0=181.7083,x1=1,y1=193.6917,col=2,angle=90,length=0.1,code=3)
arrows(x0=2,y0=98.08678,x1=2,y1=103.91322,col=2,angle=90,length=0.1,code=3)
arrows(x0=3,y0=50.73404,x1=3,y1=55.59930,col=2,angle=90,length=0.1,code=3)
arrows(x0=4,y0=26.64033,x1=4,y1=28.75967,col=2,angle=90,length=0.1,code=3)
arrows(x0=5,y0=12.36734,x1=5,y1=14.89932,col=2,angle=90,length=0.1,code=3)
arrows(x0=6,y0=7.275193,x1=6,y1=8.858141,col=2,angle=90,length=0.1,code=3)
arrows(x0=7,y0=3.256106,x1=7,y1=4.610561,col=2,angle=90,length=0.1,code=3)
arrows(x0=8,y0=1.904272,x1=8,y1=3.229061,col=2,angle=90,length=0.1,code=3)
```

```

arrows(x0=9,y0=0.9112685,x1=9,y1=1.8220649,col=2,angle=90,length=0.1,code=3)
arrows(x0=10,y0=0.612646,x1=10,y1=1.187354,col=2,angle=90,length=0.1,code=3)
arrows(x0=11,y0=0.0793943,x1=11,y1=0.3872724,col=2,angle=90,length=0.1,code=3)

# Plot of raw LA Dodgers data with superimposed 95% Confidence Intervals

x<-seq(1,11,1)
actualdogers<-c(213,103,54,27,13,11,4,6,1,2,1)
plot(x,actualdogers,ylim=c(0,250),xlab="Streak length k",ylab="Frequency",
main="Winning Streak Frequencies
of LA Dodgers with simulated 95% confidence intervals")
arrows(x0=1,y0=232.9351,x1=1,y1=238.9316,col=2,angle=90,length=0.1,code=3)
arrows(x0=2,y0=106.5535,x1=2,y1=114.1799,col=2,angle=90,length=0.1,code=3)
arrows(x0=3,y0=51.48277,x1=3,y1=56.98390,col=2,angle=90,length=0.1,code=3)
arrows(x0=4,y0=23.44766,x1=4,y1=26.41901,col=2,angle=90,length=0.1,code=3)
arrows(x0=5,y0=11.84524,x1=5,y1=14.28810,col=2,angle=90,length=0.1,code=3)
arrows(x0=6,y0=5.437938,x1=6,y1=7.162062,col=2,angle=90,length=0.1,code=3)
arrows(x0=7,y0=1.970479,x1=7,y1=3.029521,col=2,angle=90,length=0.1,code=3)
arrows(x0=8,y0=1.275418,x1=8,y1=2.057915,col=2,angle=90,length=0.1,code=3)
arrows(x0=9,y0=0.4969404,x1=9,y1=1.1030596,col=2,angle=90,length=0.1,code=3)
arrows(x0=10,y0=0.05441485,x1=10,y1=0.34558515,col=2,angle=90,length=0.1,code=3)
arrows(x0=11,y0=0.05297677,x1=11,y1=0.41368990,col=2,angle=90,length=0.1,code=3)

```

Bibliography

- [1] Kim, I.-J., Barthel, B. P., Park, Y., Tait, J. R., Dobmeier, J. L., Kim, S. and Shin, D. (2014), Network analysis for active and passive propagation models. *Networks*, 63: 160169. doi: 10.1002/net.21532
- [2] Dedeo, Simon. 2013. Collective Phenomena and Non-Finite State Computation in a Human Social System.
- [3] Hayek FA (1945) The use of knowledge in society. *The American Economic Review* XXXV: 519530.
- [4] DeCanio SJ, Watkins WE (1998) Information processing and organizational structure. *Journal of Economic Behavior and Organization* 36: 275294.
- [5] The Pareto Law and the Distribution of Income G. Findlay Shirras *The Economic Journal*, Vol. 45, No. 180 (Dec., 1935), pp. 663-681
- [6] Koch, Richard. *Living the 80/20 Way: Work Less, Worry Less, Succeed More, Enjoy More*. Nicholas Brealey Publishing. 2014.
- [7] "Baseball Almanac." *Baseball Almanac*. N.p., n.d. Web. Mar.-Apr. 2014.

- [8] Woodcock, Kathryn (2010). Safety Evaluation Techniques. Toronto, ON: Ryerson University. p. 86.
- [9] Hogg, Robert V, Joseph W. McKean, and Allen T. Craig. Introduction to Mathematical Statistics. Upper Saddle River, N.J: Pearson Education, 2005. Print.
- [10] Pareto, Vilfredo; Page, Alfred N. (1971), Translation of *Manuale di economia politica* ("Manual of political economy"), A.M. Kelley, ISBN 978-0-678-00881-2
- [11] "Baseball-Reference.com - MLB Stats, Standings, Scores, History." Baseball-Reference.com. N.p., n.d. Web. Mar.-Apr. 2014.
- [12] Koppen, Mario; Berlin, Fraunhofer IPK. The curse of dimensionality.
- [13] Mika, Sebastion; Ratsch, Gunnar; Weston, Jason; Scholkopf, Bernhard; Muller, Klaus-Robert. Fisher Discriminant Analysis with Kernals.