Lafayette College

Math 400 Senior Seminar

final draft

Regression model and its applications

Author:
Shuyu Jia

Instructor:
Dr. Justin Corvino

April 28, 2018

LAFAYETTE COLLEGE

Contents

Acknowledgments

First of all, I would like to express my sincere gratitude to my instructor Prof. Justin Corvino for the continuous support of my mathematics study, for his patience, motivation, and immense knowledge. His guidance helped me in all the time of this semester's research study and writing of this paper. Besides my instructor, I would like to thank Prof. Jeffrey Liebner for his insightful advices, encouraging comments, as well as all the material he suggested me for self-studying. Last but not least, I want to thank my fellow classmates and friends John Jamieson, Christina Sirico, and Daniel Turchiano for their precious feedback and ideas on my paper.

1 Introduction

Regression models, in general, have two main objectives. The first is to establish if there exists a relationship between two or more variables. Positive relationships are observed when an increase of one variable results in an increase of the other. Conversely, in a negative relationship, we find that if one variable increases, the other variable tends to decrease. More specifically, regression models are used to establish if there is a statistically significant relationship between the variables. For instance, on average, we expect that people in families that earn higher income will generally spend more. In this case, a positive relationship exists between income and spending. Another example could be the relation between a student's height and his/her exam score. Of course we expect no relationship in this case and we can use regression models to test that.

The second objective is to forecast new observations. In other words, we are supposed to use what we know about the relationship to predict unobserved values. For example, if we know that our sales tend to grow and how fast our sales grow, we can use this information to predict what will our sales be over the next quarter.

There are two different roles that the variables play in regression models. The first one is the dependent variable. This is the variable whose values we want to explain or predict. We call it dependent variable because its value depends on something else. We usually denote it as Y. The other variable is the independent

variable, and this is the variable that affects the other one. We usually denote it as X.

There are two main parts of this paper. In the first part I will introduce different types of linear regression models and logistic regression models, give examples about them, and talk about how to estimate model parameters for each of the regression models. In the second part I will present a case study to display the capabilities and applications of regression models. The case study is the prediction of NBA playoff results using regular season's data. I will use one of the regression models that I introduced above, which is ordinal logistic regression. I will use *R Studio* to simulate the data and explain the result in the end.

2 Linear Regression

2.1 Simple Linear Regression

Simple linear regression uses a linear equation to model the relationship between two variables, an independent variable and a dependent variable. Data, in general, consists of a series of x and y observations, which on average may follow a linear pattern. We use a linear regression model to represent such trends. Here is an example of a simple linear regression.

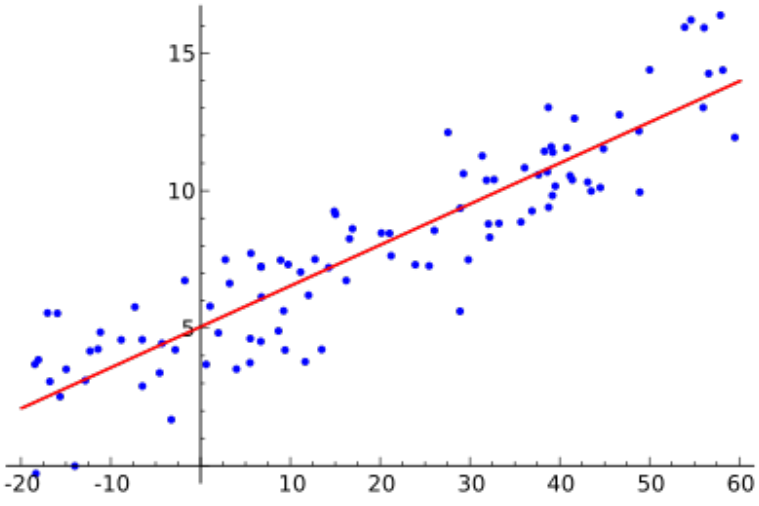

Figure 1: Example of simple linear regression[1]

The line generally does not intersect any of these observations. Rather, the distance between observations and the linear regression line measures the error that exists. The purpose of a linear regression is to find a line that minimizes these errors. Therefore, the linear regression model must include these errors. A simple linear regression model is shown below.[2][Ch.12]

$$y = \beta_0 + \beta_1 x + \epsilon. \tag{1}$$

y is the dependent variable whose value depends on all other parameters in the equation. x is the independent variable that affects the dependent variable y. β_0 is the constant term or intercept. β_1 is the slope coefficient for x. ϵ is the error term which we are trying to minimize.

Now let us have a look at an example applying actual data. Suppose there are 40 observations of 40 different families, and their weekly income and weekly consumption of a given product are known. Our simple linear regression model is as follows:

$$\text{Consumption} = \beta_0 + \beta_1 \cdot \text{Income} + \epsilon. \tag{2}$$

What we want to test is if income is good enough to explain the consumption. Suppose we simulate those data into *R Studio* and get the values of parameters as follows:

$$\text{Consumption} = 49.13 + 0.85 \cdot \text{Income} + \epsilon. \tag{3}$$

Let us interpret the coefficients. 49.13 could be interpreted as the consumption level of a family with 0 income. This makes little sense unless that family receives financial assistance from the government or whatsoever. Most generally, the intercept may not have an intuitive interpretation, so we usually ignore it. 0.85 is the marginal effect of one unit of income on consumption. In other words, for every additional unit of income a family has, we estimate its consumption grows by 0.85 units. Note that the slope always has an intuitive interpretation. It represents the sensitivity of the dependent variable on changes in the independent variable.

2.2 Multiple Linear Regression

A multiple linear regression is a lot like a simple linear regression, only instead of using a single predictor variable to make a prediction about a dependent variable, we use multiple predictor variables to make a prediction about one or more

dependent variables. Most of the same concepts from simple linear regression apply to multiple linear regression. In fact, multiple linear regression is still creating a best fit line through the data, but the data is multi-dimensional, so we usually use matrices in the calculation instead. However, the basic concepts are the same. Let us take a look at the equation for a multiple linear regression model:[2][Ch.12]

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k + \epsilon. \tag{4}$$

Consider an example related to basketball matches: we would like to analyze how turnovers and shooting percentage affect the total points scored. We intuitively think that turnovers are going to have a negative effect on points scored, and that shooting percentage is going to positively affect total points scored. However, we would like to know the results quantitatively. Here is our model:

$$\text{Points} = \beta_0 + \beta_1 \cdot \text{shooting percentage} + \beta_2 \cdot \text{turnovers} + \epsilon. \tag{5}$$

Suppose we have 100 data for this model and *R studio* gives the parameters as follows:

$$\text{Points} = -1.710 + 2.522 \cdot \text{shooting percentage} - 0.980 \cdot \text{turnovers} + \epsilon. \tag{6}$$

Based on the result, if a team increases turnovers by 1, we would expect the number of points scored by the team to decrease by 0.980. For each 1% a team's shooting percentage increases, we would expect that team to score 2.522 more points.

2.3 Estimating parameters: The method of least squares

When we are trying to estimate parameters of a linear regression, we usually use the method of least squares. Suppose our points are (x_i, y_i) and the line is $y = b_0 + b_1 x$. Then the vertical deviation (residual) for a given point (x_i, y_i) is as follows:[2][Ch.12]

$$\text{Residual} = y_i - (b_0 + b_1 x_i). \tag{7}$$

The sum of squared vertical deviations of all points is:[2][Ch.12]

$$S = \sum_{i=1}^{n} [y_i - (b_0 + b_1 x_i)]^2. \tag{8}$$

The method of least squares is trying to minimize the sum of squared vertical deviations S. To achieve that, we must take the partial derivatives of S with respect to both b_0 and b_1, then set both of them to zero and solve the equations.[2][Ch.12]

$$\frac{\partial S}{\partial b_0} = \sum_{i=1}^{n} 2(y_i - b_0 - b_1 x_i)(-1) = 0 \tag{9}$$

$$\frac{\partial S}{\partial b_1} = \sum_{i=1}^{n} 2(y_i - b_0 - b_1 x_i)(-x_i) = 0. \tag{10}$$

After solving the equations we get the critical point:[2][Ch.12]

$$b_1 = \widehat{\beta}_1 = \frac{\sum(x_i - \overline{x})(\sum(y_i - \overline{y})}{\sum(x_i - \overline{x})^2} \tag{11}$$

$$b_0 = \widehat{\beta}_0 = \overline{y} - \widehat{\beta}_1 \overline{x}. \tag{12}$$

There are two things we need to check. First, we need to make sure the critical point we get is not a saddle by using the second derivative test:

$$\frac{\partial^2 S}{\partial b_0^2}\frac{\partial^2 S}{\partial b_1^2} - \left(\frac{\partial^2 S}{\partial b_0 \partial b_1}\right)^2 = 2n \cdot \sum_{i=1}^{n} 2x_i^2 - \left(\sum_{i=1}^{n} 2x_i\right)^2 \tag{13}$$

$$= 4 \cdot \left[n \cdot \sum_{i=1}^{n} x_i^2 - \left(\sum_{i=1}^{n} x_i\right)^2 \right]. \tag{14}$$

By defining vector $\mathbf{u}$ and vector $\mathbf{v}$ as follows:

$$\mathbf{u} = \begin{pmatrix} x_1 & x_2 & \cdots & x_n \end{pmatrix}, \mathbf{v} = \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix}, \tag{15}$$

we can rewrite the second derivative test as follows:

$$\frac{\partial^2 S}{\partial b_0^2}\frac{\partial^2 S}{\partial b_1^2} - \left(\frac{\partial^2 S}{\partial b_0 \partial b_1}\right)^2 = 4 \cdot \left[|\mathbf{u}|^2|\mathbf{v}|^2 - (\mathbf{u} \cdot \mathbf{v})^2\right] \tag{16}$$

Recall the Cauchy-Schwarz inequality that if we have vector $\mathbf{u}$ and vector $\mathbf{v}$, the following holds:

$$|\mathbf{u} \cdot \mathbf{v}| \leq |\mathbf{u}| \cdot |\mathbf{v}|. \tag{17}$$

Since both sides are positive numbers, when squaring both sides, the inequality still holds:

$$(\mathbf{u} \cdot \mathbf{v})^2 \leq |\mathbf{u}|^2 |\mathbf{v}|^2. \tag{18}$$

Based on their definitions, row vector $\mathbf{u}$ and column vector $\mathbf{v}$ are linearly independent, so the equality never holds. Therefore we can confirm that the critical point we get is not a saddle point from Eq.(16):

$$\frac{\partial^2 S}{\partial b_0^2} \frac{\partial^2 S}{\partial b_1^2} - \left(\frac{\partial^2 S}{\partial b_0 \partial b_1} \right)^2 = 4 \cdot \left[|\mathbf{u}|^2 |\mathbf{v}|^2 - (\mathbf{u} \cdot \mathbf{v})^2 \right] > 0. \tag{19}$$

Next, we need to take the second partial derivatives with respect to both b_0, b_1 and to make sure the results are both positive. This ensures that the critical point actually minimizes S.

$$\frac{\partial^2 S}{\partial b_0^2} = \sum_{i=1}^{n} 2 = 2n > 0 \tag{20}$$

$$\frac{\partial^2 S}{\partial b_1^2} = \sum_{i=1}^{n} 2x_i^2 > 0. \tag{21}$$

The solutions are denoted by $\widehat{\beta}_0$ and $\widehat{\beta}_1$. They are called the least squares estimates which minimize S, the sum of squared vertical deviations. The estimated regression line or least squared line is then given by $y = \widehat{\beta}_0 + \widehat{\beta}_1 x$.

In multiple linear regressions, especially when we have multiple dependent variables, we are using matrices to calculate least squares. However the method is exactly the same. Without loss of generality, let us suppose that we have three dependent variables, each consisting of three predictors. Then we can write our Y matrix as follows:

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} \beta_0 + \beta_1 x_{11} + \beta_2 x_{12} + \beta_3 x_{13} + \epsilon_1 \\ \beta_0 + \beta_1 x_{21} + \beta_2 x_{22} + \beta_3 x_{23} + \epsilon_2 \\ \beta_0 + \beta_1 x_{31} + \beta_2 x_{32} + \beta_3 x_{33} + \epsilon_3 \end{bmatrix}. \tag{22}$$

The equations can be written more compactly using vectors and matrices. Therefore we would like to define a series of column vectors.

$$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}, \boldsymbol{\beta} = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \beta_3 \end{bmatrix}, \boldsymbol{\epsilon} = \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \end{bmatrix}. \tag{23}$$

The $\mathbf{X}$ matrix has a row for each observation, consisting of 1 and then the values of the three predictors.[2][Ch.12]

$$\mathbf{X} = \begin{bmatrix} 1 & x_{11} & x_{12} & x_{13} \\ 1 & x_{21} & x_{22} & x_{23} \\ 1 & x_{31} & x_{32} & x_{33} \end{bmatrix}. \tag{24}$$

Therefore we can express the equation very concisely as follows.[2][Ch.12]

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}. \tag{25}$$

We are using the same principle as in simple linear regression, so we are still trying to minimize the sum of squared vertical deviations of all points by setting the partial derivative with respect to each coefficient to zero, finding the critical points, and realizing those are minimum points. After all the calculations, the vector of estimated coefficients is:[2][Ch.12]

$$\boldsymbol{\beta} = [\mathbf{X}^\mathbf{T}\mathbf{X}]^{-1}\mathbf{X}^\mathbf{T}\mathbf{y}. \tag{26}$$

where $\mathbf{X}^\mathbf{T}$ refers to the transpose of $\mathbf{X}$.

3 Logistic Regression

3.1 General Logistic Regression

Linear regression models in general, have some problems dealing with binary outcomes. However, a technique called logistic regression is good at answering yes/no questions. Let me give you some examples of binary outcomes: (1) Should a bank give a person a loan or not? (2) Is a particular student is admitted into a school? (3) Is a person voting against a new law? For those questions there are only two outcomes: Yes and No, and we usually define a dummy variable to indicate if an observation is a Yes or a No in the following way.

$$y = \begin{cases} 1 & \text{if Yes} \\ 0 & \text{if No} \end{cases}. \tag{27}$$

Note that if we define our dummy variable the other way around, the coefficients will have the same magnitudes but opposite signs. For example, whatever helped me buy a product will have the exact opposite effect on helping me not buy that product, and thus we are going to have the opposite sign for that attribute.

Consider an example of customers' subscription to a magazine. There are only two outcomes in this example: subscribed to the magazine or not. Suppose we have data on 1000 random customers from a given city, and we want to know what determines their decision to subscribe to a magazine. In this case our dependent variable is going to be an indicator variable that tells us if a customer has subscribed to the magazine or not. The dependent variable is going to be a 1 if the subscription took place and it is going to be a 0 otherwise. We will be examining how age influences the likelihood of subscription in this example.

If we think about the problem, we do not see too many reasons why we could not use a linear model, because aside from being binary, there is really nothing special about our dependent variable y. In fact, if we want to change this binary variable from a zero to a one, we are changing its value to a higher value and thus anything that increases the value of y should favor the likelihood of a customer's subscribing to the magazine. Thus, we could run a simple linear regression model and suppose we get the following result:

$$\text{subscribe} = -1.700 + 0.064 \cdot \text{age}. \tag{28}$$

As we interpret the result, we are left wondering what makes it change from a 0 to 1. This can also be interpreted as what increase the likelihood of subscription, or $P(\text{subscribe} = 1)$, which we can also simply denote as p. Therefore the result can be also read as

$$P(\text{subscribe} = 1) = p = -1.700 + 0.064 \cdot \text{age}. \tag{29}$$

Now we can conclude that every additional year of age increases the probability of subscription by 6.4%. However, problems arise when we try to use this to forecast the probabilities of customers with given ages. We know that the probabilities are bounded whereby $0 \leq p \leq 1$, and suppose the range of age in the dataset is $20 \leq \text{age} \leq 55$. We can surely predict the probability that a 35 year-old person subscribes is

$$p = -1.700 + 0.064 \cdot 35 = 0.54. \tag{30}$$

However once we try to predict the probabilities for 25 and 45 years of age, we get

invalid values for both probabilities.

$$p = -1.700 + 0.064 \cdot 25 = -0.09 \tag{31}$$

$$p = -1.700 + 0.064 \cdot 45 = 1.20. \tag{32}$$

By the definition of the probability, any probability needs to stay between 0 and 1. In this model when the customers are young, say below 26 years of age or so, the estimated probabilities are negative. Meanwhile if the customer has more than 44 years of age, the probability is greater than 1. Both situations break the model, and we cannot use it.

There are two main attributes that must be satisfied. First, the probability must always be positive, since $p \geq 0$. An exponential form would satisfy this.

$$p = e^{\beta_0 + \beta_1 \cdot \text{age}}. \tag{33}$$

Second, the probability must be at most 1, since $p \leq 1$. Then we propose to use the expression we had before, and divided by something that is slightly larger, say 1.[2][Ch.12]

$$p = \frac{e^{\beta_0 + \beta_1 \cdot \text{age}}}{e^{\beta_0 + \beta_1 \cdot \text{age}} + 1}. \tag{34}$$

Note that we could have added any small or large values in the denominator, and the condition that having a value less than 1 would still be satisfied; however, we use 1 for reasons that will be explained in Section 3.2, where we introduce *log odds*.

Even though we have this more complex expression, the linear thinking is not completely gone. If we do some algebra, the previous expression can be rewritten as follows.

$$\ln\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 \cdot \text{age}. \tag{35}$$

Therefore, even if the probability of a customer subscribing (p) is not a linear function of age, we can perform a simple transformation on it such that it is now a linear function of age. This approach is also called log-linear modeling. [3][Ch.8] The above equation is the one used in logistic regressions. The result for our example is as follows:

$$\ln\left(\frac{p}{1-p}\right) = -26.52 + 0.78 \cdot \text{age}. \tag{36}$$

Written in terms of the probability p, instead, we have

$$p = \frac{e^{-26.52+0.78\cdot \text{age}}}{e^{-26.52+0.78\cdot \text{age}} + 1}. \tag{37}$$

As we try to predict probabilities using the logistic regression model mentioned above, the predicted probabilities for all ages are no longer below 0 or above 1. In this model, as customers grow older, the probability asymptotically gets closer to 1; and when the age approaches to the other direction, the function is asymptotically closer to 0; they never go below 0 or above 1. This is why we want to use logistic regression models.

3.2 Multinomial Logistic Regression

There are quantitative variables and categorical variables. Some examples of quantitative variables are age, income, and height. All of them have numerical values. Categorical variables, on the other hand, are simply variables that fall into categories. For instance, if we have categories of hair color, some of the categorical variables would be black, red, and brown. Note that hair color itself is not the categorical variable, it is the category. Within categorical variables, there are ordered categorical variables and unordered categorical variables. For example, we can use five ordered categorical variables on a questionnaire: strongly agree, agree, neutral, disagree, and strongly disagree. Another example of unordered category can be the choice of major. The corresponding unordered categorical variables can be mathematics, engineering, history and art. Indeed we can somehow order the choice of major by someone's preference, or in alphabetical order, however, if we want to model how gender and age affect the students' choices of major, there is no point in ordering the majors in any way. Therefore, whether or not we order categorical variables depends on what we are trying to model.

In binary logistic regression, we normally model categorical dependent variables which takes only two values, 0 and 1. In multinomial logistic regression, we extend that to multiple categories. In other words, a multinomial logistic regression is used to model categorical dependent variables which take more than two categories. In multinomial logistic regressions, the dependent categorical variables cannot be ordered, otherwise we need to use ordinal logistic regression which we will discuss in the next section.

What I am going to introduce next is the concept of *log odds*. In the last section we defined the binary dependent variable as follows:

$$\text{subscribe} = \begin{cases} 1 & \text{if subscribe} \\ 0 & \text{if not subscribe} \end{cases}. \tag{38}$$

We saw that the logit function $\ln\left(\frac{p}{1-p}\right)$ actually follows a linear pattern because our model is:

$$\ln\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 \cdot \text{age}. \tag{39}$$

We also defined that p is the probability of subscription, or $P(\text{subscribe} = 1)$, and let's simply denote it as $P(1)$. Therefore $1 - p$ should be the probability of not subscribe, we denote it as $P(\text{subscribe} = 0)$, or $P(0)$. Thus the logit function becomes

$$\ln\left[\frac{P(1)}{P(0)}\right] = \beta_0 + \beta_1 \cdot \text{age}. \tag{40}$$

The equation above is what we called a *log odd* in binary logistic regression. This also explains the problem we left in Section 3.1. Let us recall the model we introduced before:[2][Ch.12]

$$p = \frac{e^{\beta_0 + \beta_1 \cdot \text{age}}}{e^{\beta_0 + \beta_1 \cdot \text{age}} + 1}. \tag{41}$$

As we said in Section 3.1, we could add any positive number in the denominator. However, we can only get the exact log odd form for the model by adding 1 to the denominator.

In multinomial logistic regression, if there are N unordered categories, there will be $N-1$ log odds. For example, we would like to model how gender and age affect students' choices of major, and we assume there are four categories: Mathematics, Engineering, History and Art. First, if we define Art as our base category, then the three log odds are $\ln\left[\frac{P(\text{Maths})}{P(\text{Art})}\right]$, $\ln\left[\frac{P(\text{Engineer})}{P(\text{Art})}\right]$, and $\ln\left[\frac{P(\text{History})}{P(\text{Art})}\right]$.

From that we can build our multinomial logistic model.

$$\ln\left[\frac{P(\text{Maths})}{P(\text{Art})}\right] = a_0 + a_1 \cdot \text{gender} + a_2 \cdot \text{age} \tag{42}$$

$$\ln\left[\frac{P(\text{Engineer})}{P(\text{Art})}\right] = b_0 + b_1 \cdot \text{gender} + b_2 \cdot \text{age} \tag{43}$$

$$\ln\left[\frac{P(\text{History})}{P(\text{Art})}\right] = c_0 + c_1 \cdot \text{gender} + c_2 \cdot \text{age}. \tag{44}$$

where a, b, c are parameters we try to estimate.

3.3 Ordinal Logistic Regression

Contrary to multinomial logistic regression, ordinal logistic regression is used to model an ordered categorical scale. An example would be modeling whether a symptom is mild, moderate or severe. For general logistic regressions we discussed in Section 3.1, we have the logit function as follows:

$$\ln\left(\frac{p}{1-p}\right) = a + b_1x_1 + b_2x_2 + \cdots + b_kx_k. \tag{45}$$

where p represent the probability of the outcome. For an ordinal logistic regression, we have several outcomes. The key is to predict the partitions of several outcomes. In other words, we want to predict probabilities of outcome less or equal to i, and call it p_i. For example, to model if a symptom is mild, moderate or severe, we define p_1 as the probability of mild and p_2 be the probability of mild or moderate. Therefore the ordinal logistic regression model is as follows.

$$\ln\left(\frac{p_i}{1-p_i}\right) = a_i + b_1x_1 + b_2x_2 + \cdots + b_kx_k. \tag{46}$$

where a_i represents the intercept for an outcome less or equal to i, and $b_1, b_2, \cdots, b_k$ are parameters we want to estimate. The case study we will introduce later uses the ordinal logistic regression.

3.4 Estimating Parameters: Maximum Likelihood Estimation

In Section 2.3, we introduced the method to estimate the parameters of linear regressions. In logistic regressions, we normally use *Maximum Likelihood Estimation* (MLE) to get our model parameters. MLE is used with a wide range of

statistical analyses. I will use an example of the U.S. population to explain how the MLE works. In this example, we would like to know the probability that a given U.S. individual is male. Of course, since there are approximately 326 million individuals in the U.S., it is impossible to collect sex information from all of them. But let us suppose we have a sample of the U.S. population, and in the sample we have the sex information for each individual. Suppose we have N individuals in this sample and we are interested in evaluating what the probability is that an individual from the U.S is male, given that we only have a sample from the U.S. population. The idea is that there may be some probability distribution function which determines the probability that given individual was either going to be male, or female. We can call the probability density function $f(x_i|p)$. Here p is the probability that an individual is male, a population fraction which we are trying to estimate, but we just assume that we know p at this point. The probability density function (PDF) is as follows:

$$f(x_i|p) = p^{x_i}(1-p)^{1-x_i}. \tag{47}$$

x_i here is the categorical variable defined as follows.

$$x_i = \begin{cases} 1 & \text{male} \\ 0 & \text{female} \end{cases}. \tag{48}$$

When we try to substitute $x_i = 0$ and $x_i = 1$ in the PDF, we can find out that p is the probability that a given individual is male and $1-p$ is the probability that a given individual is female.

$$f(1|p) = p^1(1-p)^{1-1} = p \tag{49}$$

$$f(0|p) = p^0(1-p)^{1-0} = 1-p. \tag{50}$$

We can see that our function f is telling us the exact same probabilities as we defined before. That is the case for one observation. Now we want to add some supplemental observations. In that circumstance, we define our pdf as follows.

$$f(x_1, x_2, \cdots, x_N|p) = p^{x_1}(1-p)^{1-x_1}p^{x_2}(1-p)^{1-x_2}\cdots p^{x_N}(1-p)^{1-x_N} \tag{51}$$

$$= \prod_{i=1}^{N} p^{x_i}(1-p)^{1-x_i}. \tag{52}$$

Note that we are considering observations as random samples. In other words, they are all independent of one another. Therefore the pdf is just going to be individual pdfs multiplied together. Now we define some random variables $X_1, X_2, ..., X_N$, and each of them represents the sex of a corresponding individual. When we multiply each of the individual density functions, we obtain a joint probability in the discrete case as follows:

$$P(X_1 = x_1, X_2 = x_2, ..., X_N = x_N) = \prod_{i=1}^{N} p^{x_i}(1-p)^{1-x_i} = L. \tag{53}$$

This joint probability is what we define as the likelihood L. In general, we do not know the probability p, and indeed that is something we would like to estimate. The idea is that we want to maximize L, the probability of having observed the sample, given p. Therefore we want to differentiate this likelihood function with respect to p and set it to 0 in order to solve for the critical point of function L.

$$\frac{dL}{dp} = 0. \tag{54}$$

However, the problem with differentiating the function L is that the function L is a product, which is difficult to differentiate. We can try the product rule, but the function L has N parts of it, so we arrive at a dead end. But all is not lost, and we can rectify the natural log of the likelihood, which we can also call it log likelihood, l.

$$l = \ln(L) = \ln\left(\prod_{i=1}^{N} p^{x_i}(1-p)^{1-x_i}\right). \tag{55}$$

If we differentiate the log likelihood l with respect to p and set it equal to zero, then that gives us the same estimator for p that we would have obtained if we just differentiated the likelihood L rather than the log likelihood l. This is because of the monotonicity of the log transformation, and the idea that log function is always increasing in its argument. Since the log function is always increasing, then the value p that maximizes the log likelihood l is able to maximize the likelihood function L as well. Recall some of the log rules, so we can discover the benefits of taking the natural log.

$$\ln(a \cdot b) = \ln(a) + \ln(b) \tag{56}$$

$$\ln\left(a^b\right) = b \cdot \ln(a). \tag{57}$$

Since the likelihood function L is a product, taking the log of it turns it into a sum. Simplify it further as follows:

$$l = \ln\left(\prod_{i=1}^{N} p^{x_i}(1-p)^{1-x_i}\right) \tag{58}$$

$$= \sum_{i=1}^{N} \ln\left(p^{x_i}(1-p)^{1-x_i}\right) \tag{59}$$

$$= \sum_{i=1}^{N} \left[x_i \ln(p) + (1-x_i)\ln(1-p)\right] \tag{60}$$

$$= \ln(p) \cdot \sum_{i=1}^{N} x_i + \ln(1-p) \cdot \sum_{i=1}^{N}(1-x_i). \tag{61}$$

We may simplify this further if we recognize that $\sum_{i=1}^{N} x_i = N \cdot \overline{x}$ where $\overline{x}$ is the mean of all observations x_i.

$$l = \ln(p) \cdot N \cdot \overline{x} + \ln(1-p) \cdot N \cdot (1-\overline{x}). \tag{62}$$

Now the log likelihood function l is ready to be differentiated with respect to parameter p, and then we set it to zero in order to find the critical point.

$$\frac{dl}{dp} = \frac{N \cdot \overline{x}}{p} - \frac{N \cdot (1-\overline{x})}{1-p} = 0. \tag{63}$$

Using algebra, we get our critical point $p = \overline{x}$. Then we need to take the second derivative of l with respect to p and confirm that the result is negative.

$$\frac{d^2l}{dp^2} = -\frac{N \cdot \overline{x}}{p^2} - \frac{N \cdot (1-\overline{x})}{(1-p)^2} < 0. \tag{64}$$

Thus we know that this critical point p actually maximizes the log likelihood l. We denote the maximum likelihood estimator as $\widehat{p}_{ML}$ and we get $\widehat{p}_{ML} = \overline{x}$. Since the log function is monotonic, we know that $\widehat{p}_{ML}$ also maximizes our original likelihood function L. Finally we get our parameter p in the original probability density function (PDF) we defined before. The result shows that the maximum likelihood estimator for the population parameter p is just going to be the fraction of individuals who are male in the sample. That makes intuitive sense because p in the population is just the fraction of individuals who are male in the population, so the maximum likelihood estimator for the population parameter p is just going to be the fraction of individuals who are male in the sample.

4 Case Study: NBA playoff results estimation

4.1 Introduction of the problem

Every year, people go crazy about NBA playoffs. Plenty of people try to predict the winner before the series. Often they will predict that the team with a higher win rate in regular season will also win the series. However sometimes that is not the case. There are obviously more factors affecting the series, such as offense, defense, or even the play style of the team. Therefore, I use the following data of regular season taken from basketball-reference.com to estimate playoff results: (1) regular season total win rate; (2) points scored per game; (3) points allowed per game; (4) regular season matching result between the two teams.

4.2 Dataset

According to the NBA rules history, the 3-point line extension in 1997-98 season was one of the biggest rule changes, which could make a statistical difference in my predictors.[4] This is because 3-point line extension will likely affect the shooting percentage of all teams. In this way, both the points scored per game and the points allowed per game are going to be different. Aside from this, a lot of teams will change their play style according to the distance from which they shoot the 3-point basket. As a result, the match-ups between the teams are going to have different results, so I am only using playoff data since 1997-98 season. The first rounds of NBA playoffs were not best of 7 until 2002-03 season, so I did not use the first round NBA playoffs data from 1997-98 season to 2002-03 season. Finally I acquired a total of 260 playoff series within the 20 seasons in my dataset. On the next page there is a segment of my original dataset; there are supposed to be a total of 260 entries.

2014-15 season	result	W/L %	Point scored/Game	Points allowed/Game	matchup result
Golden State Warriors - Cleveland Cavaliers	(4-2)	0.817-0.646	110.0-103.1	99.9-98.7	(1-1)
Cleveland Cavaliers - Atlanta Hawks	(4-0)	0.646-0.732	103.1-102.5	98.7-97.1	(1-3)
Golden State Warriors - Houston Rockets	(4-1)	0.817-0.683	110.0-103.9	99.9-100.5	(4-0)
Atlanta Hawks - Washington Wizards	(4-2)	0.732-0.561	102.5-98.5	97.1-97.8	(3-1)
Cleveland Cavaliers - Chicago Bulls	(4-2)	0.646-0.610	103.1-100.8	98.7-97.8	(3-1)
Golden State Warriors - Memphis Grizzlies	(4-2)	0.817-0.671	110.0-98.3	99.9-95.1	(2-1)
Houston Rockets - Los Angeles Clippers	(4-3)	0.683-0.683	103.9-106.7	100.5-100.1	(2-2)
Atlanta Hawks - Brooklyn Nets	(4-2)	0.732-0.463	102.5-98.0	97.1-100.9	(4-0)
Chicago Bulls - Milwaukee Bucks	(4-2)	0.610-0.500	100.8-97.8	97.8-97.4	(3-1)
Cleveland Cavaliers - Boston Celtics	(4-0)	0.646-0.488	103.1-101.4	98.7-101.2	(2-2)
Washington Wizards - Toronto Raptors	(4-0)	0.561-0.598	98.5-104.0	97.8-100.9	(0-3)
Golden State Warriors - New Orleans Pelicans	(4-0)	0.817-0.549	110.0-99.4	99.9-98.6	(3-1)
Houston Rockets - Dallas Mavericks	(4-1)	0.683-0.610	103.9-105.2	100.5-102.3	(3-1)
Los Angeles Clippers - San Antonio Spurs	(4-3)	0.683-0.671	106.7-103.2	100.1-97.0	(2-2)
Memphis Grizzlies - Portland Trail Blazers	(4-1)	0.671-0.622	98.3-102.8	95.1-98.6	(4-0)
2015-16 season	result	W/L %	Point scored/Game	Points allowed/Game	matchup result
Cleveland Cavaliers - Golden State Warriors	(4-3)	0.695-0.890	104.3-114.9	98.3-104.1	(0-2)
Cleveland Cavaliers - Toronto Raptors	(4-2)	0.695-0.683	104.3-102.7	98.3-98.2	(1-2)
Golden State Warriors - Oklahoma City Thunder	(4-3)	0.890-0.671	114.9-110.2	104.1-102.9	(3-0)
Cleveland Cavaliers - Atlanta Hawks	(4-0)	0.695-0.585	104.3-102.8	98.3-99.2	(3-0)
Toronto Raptors - Miami Heat	(4-3)	0.683-0.585	102.7-100.0	98.2-98.4	(3-1)
Golden State Warriors - Portland Trail Blazers	(4-1)	0.890-0.537	114.9-105.1	104.1-104.3	(3-1)
Oklahoma City Thunder - San Antonio Spurs	(4-2)	0.671-0.817	110.2-103.5	102.9-92.9	(2-2)
Atlanta Hawks - Boston Celtics	(4-2)	0.585-0.585	102.8-105.7	99.2-102.5	(3-1)
Cleveland Cavaliers - Detroit Pistons	(4-0)	0.695-0.537	104.3-102.0	98.3-101.4	(1-3)
Miami Heat - Charlotte Hornets	(4-3)	0.585-0.585	100.0-103.4	98.4-100.7	(2-2)
Toronto Raptors - Indiana Pacers	(4-3)	0.683-0.549	102.7-102.2	98.2-100.5	(3-1)
Golden State Warriors - Houston Rockets	(4-1)	0.890-0.500	114.9-106.5	104.1-106.4	(3-0)
Oklahoma City Thunder - Dallas Mavericks	(4-1)	0.671-0.512	110.2-102.3	102.9-102.6	(4-0)
Portland Trail Blazers - Los Angeles Clippers	(4-2)	0.537-0.646	105.1-104.5	104.3-100.2	(1-3)
San Antonio Spurs - Memphis Grizzlies	(4-0)	0.817-0.512	103.5-99.1	92.9-101.3	(4-0)

Figure 2: A segment of original dataset[5]

I use an ordinal logistic regression to model the relationship between the playoff result and my four predictors, making several modifications to my original dataset. First, all independent variables (predictors) should be numbers, so I formally define the four predictors of my model as follows: (1) the difference of the win/loss percentage between the two teams; (2) the difference of points scored per game between the two teams; (3) the difference of points allowed per game between the two teams; (4) the net-win between the two teams in regular season. Take one of the series in 2015-16 season, the series of the Cleveland Cavaliers versus the Golden State Warriors, as an example. In my original dataset the win/loss fractions of the two teams are 0.695 and 0.890. Then I calculated the difference of the win/loss percentage between the two teams,-0.195. Similarly, calculations were performed on the difference of points scored per game between the two teams

and the difference of points allowed per game between the two teams. For the regular season matching result between the two teams, I simply calculated the difference and put the result in the net-win column. For example, the Cleveland Cavaliers went 0-2 versus the Golden State Warriors in the regular season of 2015-16, so the net-win is -2.

Second, the original dataset was ordered by the winners first, so only four different results exist in the dataset: 4-0, 4-1, 4-2 and 4-3. However, in reality we do not know which team is going to win before they play. Therefore I flipped half of the 4-0 series into 0-4, negating the 4 independent variables (predictors) at the same time. Then I performed the same operation to half of the 4-1, 4-2, and 4-3 series as well. Finally I got eight different results in my dataset.

Last but not least, since we are using ordinal logistic regression, we must define an ordered factor for all of our possible response. I changed the result column of my dataset to its corresponding factor as shown below:

Result	factor
(4-0)	0
(4-1)	1
(4-2)	2
(4-3)	3
(3-4)	4
(2-4)	5
(1-4)	6
(0-4)	7

Figure 3: Definition of factor

Now my dataset was properly prepared to be simulated in *R Studio*. Here is an excerpt of my final dataset:

factor	WLdiff	PSdiff	PAdiff	netwins
2	0	-4.3	-4.8	-2
3	0.049	0.7	-0.3	0
0	0.012	-4.5	-3.4	-2
1	0.134	0.1	-5	2
6	-0.183	-4.4	-0.8	-1
1	0	4.9	4.4	-2
6	-0.073	-8.5	-5.9	-2
1	0.2	6.4	-0.7	0
5	0.12	8.3	5.5	1
7	-0.04	2	3.8	-2
0	0.1	5	3.3	-1
7	0.08	-0.1	-2	1
2	-0.04	1.5	1.7	-1
0	0.12	-6.2	-11.3	3
5	-0.134	0.5	4.4	0
2	0.073	9.2	6	0
4	-0.097	-3.3	-1.3	0
5	-0.085	-6.5	-3.3	0
3	-0.024	-2.3	-0.6	-2
6	-0.171	-1.9	1.4	-4
1	0.049	1	-1	2
6	0	-5.9	-6.8	0
4	-0.049	6	6.5	0
7	0.024	-4.4	-8.8	0
3	0.073	8.8	7.1	-2
4	-0.11	2.9	5	2
0	0.012	-1.1	1.3	2
1	0.061	-4.3	-7.8	2
7	-0.073	-5.1	-2.1	0

Figure 4: An excerpt of final dataset

4.3 Simulation in R studio

Since there are data of 20 seasons in my dataset, I am going to feed in the data of the first 19 seasons (line 2 to line 246, a total of 245 series) and build the ordinal logistic regression model, and then test the model using the data of the last season (line 247 to line 261, a total of 15 series). The R code is as follows.

```
> rm(list=ls())
> require(MASS)
> library(readr)
> data <- read_csv("C:/Users/Shuyu Jia/Desktop/data.csv")
> View(data)
> table=data
> X = factor(table$Result, levels = 0:7, ordered = T)
> m = polr(X ~ WLdiff + PSdiff +PAdiff + netwins, data = table, Hess = TRUE)
> m
Call:
polr(formula = X ~ WLdiff + PSdiff + PAdiff + netwins, data = table,
    Hess = TRUE)

Coefficients:
    WLdiff     PSdiff     PAdiff    netwins
-2.7213527 -0.1900587  0.2051392 -0.1335234

Intercepts:
        0|1         1|2         2|3         3|4         4|5         5|6
-2.99289254 -1.73445824 -0.66940465 -0.08548189  0.47699628  1.57411600
        6|7
 2.86102526

Residual Deviance: 898.6684
AIC: 920.6684
```

Figure 5: The model built from first 19 seasons

Here is the R code for two tests. There are another 13 tests, not shown here..

```
> test247 = data.frame(WLdiff = 0.195, PSdiff = 5.6, PAdiff = -2.9, netwins =
 0)
> predict(m, test247, type = "probs")
          0           1           2           3           4           5
0.309385000 0.302546446 0.208679449 0.070720620 0.043713050 0.042290544
          6           7
0.016302206 0.006362685


> test248 = data.frame(WLdiff = 0.024, PSdiff = -2.3, PAdiff = -1.8, netwins
= -2)
> predict(m, test248, type = "probs")
         0          1          2          3          4          5          6
0.03687990 0.08189561 0.16232211 0.13104842 0.13951330 0.23493152 0.14371559
         7
0.06969354
```

Figure 6: Testing the data of the last season

In order to show the accuracy of this model, I performed two predictions. First, to predict which team is going to win the series, we add the 0, 1, 2, and 3 entries and we get the probability that team A will win the series. Similarly, by adding

the 4, 5, 6, and 7 entries we get the probability that team B will win. I compared the two probabilities, and then made the prediction. In Test 247, we already know the actual result is 1, which means that team A won by 4-1 last season. In the simulation, the sum of 0,1,2, and 3 entries is about 89.1%, then the sum of 4,5,6, and 7 entries has to be $1 - 89.1\% = 10.9\%$ (See Figure 6). Since 89.1% is greater than 10.9%, we predict that team A will win the series. We can see that this is a correct prediction based on the actual result.

Second, by performing each test, *R Studio* calculates the probability for each possible result. When we check the actual result of Test 248 which happened last season, it is 6, which means team A lost 1-4. Then, looking at the simulation results, the model tells us the probability of the factor being 6 is about 14.4% (See Figure 6), which is not small given the fact that we have eight possible results. Therefore, I am interested to see how frequently the actual result lands on a simulated probability greater than 10%. 10% is an arbitrary value, just to test how the model performs, and is not based on any metric. Below are the details of the predictions:

1	Actual result	W/L diff	PS/G diff	PA/G diff	netwins	win/lose prediction	if the result lands on a probability >10%
2	1	0.195	5.6	-2.9	0	yes	yes
3	6	0.024	-2.3	-1.8	-2	yes	yes
4	0	0.073	10.6	6.2	-1	yes	yes
5	4	-0.048	1.2	2	0	yes	yes
6	7	0	-3.4	-4.6	-2	yes	no
7	0	0.195	15.2	7.5	1	yes	yes
8	2	0.073	-10	-11.5	2	yes	yes
9	5	-0.146	-5.1	-3	0	yes	yes
10	7	-0.11	-5.2	-1.9	-2	yes	yes
11	2	0.11	3.3	-1.2	2	yes	yes
12	5	-0.074	-6	-3.4	-2	yes	yes
13	0	0.317	8	-4.2	4	yes	yes
14	1	0.098	8.7	3.8	2	yes	yes
15	2	0.22	4.8	-1.9	0	yes	yes
16	3	0	-8	-7.6	-2	no	yes

Figure 7: Prediction results

Out of 15 tests (the last season's data), 14 were correct win/loss predictions, and 14 actual results are predicted by the model with a probability over 10% (See Figure 7). The accuracy is over 93% for both predictions. It seems that the ordinal logistic regression model for NBA playoffs prediction performs very well.

The NBA regular season this year has just passed, now is the perfect time to predict this year's playoff results. To begin, I acquired this season's data on nba-reference.com:

2017-18 season 1st round (not start yet)	result	W/L %	Point scored/Game	Points allowed/Game	matchup result
Toronto Rapters - Washington Wizards		0.720-0.524	111.7-106.6	103.9-106.0	(2-2)
Boston Celtics - Milwaukee Bucks		0.671-0.537	104.0-106.5	100.4-106.8	(2-2)
Philadelphia 76ers - Miami Heat		0.634-0.537	109.8-103.4	105.3-102.9	(2-2)
Cleveland Cavaliers - Indiana Pacers		0.610-0.585	110.9-105.6	109.9-104.2	(1-3)
Houston Rockets - Minnesota Timberwolves		0.793-0.573	112.4-109.5	103.9-107.3	(4-0)
Golden State Warriors - San Antonio Spurs		0.707-0.573	113.5-102.7	107.5-99.8	(3-1)
Portland Trail Blazers - New Orleans Pelicans		0.598-0.585	105.6-111.7	103.0-110.4	(2-2)
Oklahoma City Thunder - Utah Jazz		0.585-0.585	107.9-104.1	104.4-99.8	(3-1)

Figure 8: Original data of this season[5]

Again, I modify the data as I did earlier:

matchups	WLdiff	PSdiff	PAdiff	netwins
Toronto Rapters - Washington Wizards	0.196	5.1	-2.1	0
Boston Celtics - Milwaukee Bucks	0.134	-2.5	-6.4	0
Philadelphia 76ers - Miami Heat	0.097	6.4	2.4	0
Cleveland Cavaliers - Indiana Pacers	0.025	5.3	5.7	-2
Houston Rockets - Minnesota Timberwolves	0.22	2.9	-3.4	4
Golden State Warriors - San Antonio Spurs	0.134	10.8	7.7	2
Portland Trail Blazers - New Orleans Pelicans	0.013	-6.1	-7.4	0
Oklahoma City Thunder - Utah Jazz	0	3.8	4.6	2

Figure 9: Modified data of this season

This time I will be using data from all 20 seasons to build the ordinal logistic regression model.

```
> rm(list=ls())
> require(MASS)
> library(readr)
> Math_400_data <- read_csv("C:/Users/Shuyu Jia/Desktop/Math 400 data.csv")
> View(Math_400_data)
> table = Math_400_data
> X = factor(table$Result, levels = 0:7, ordered = T)
> m = polr(X ~ WLdiff + PSdiff +PAdiff + netwins, data = table, Hess = TRUE)
> m
Call:
polr(formula = X ~ WLdiff + PSdiff + PAdiff + netwins, data = table,
    Hess = TRUE)

Coefficients:
    WLdiff     PSdiff     PAdiff    netwins
-2.1751424 -0.2197330  0.2232147 -0.1541273

Intercepts:
        0|1         1|2         2|3         3|4         4|5         5|6
-3.01276302 -1.76427192 -0.67184852 -0.08643246  0.47749794  1.57776497
        6|7
 2.82670210

Residual Deviance: 946.2874
AIC: 968.2874
```

Figure 10: The model built from all 20 seasons

Here are the probabilities provided by *R Studio*:

Toronto Raptors (East 1st seed) - Washington Wizards (East 8th seed):

```
> test1 = data.frame(WLdiff = 0.196, PSdiff = 5.1, PAdiff = -2.1, netwins =
0)
> predict(m, test1, type = "probs")
     0          1          2          3          4          5
 0.269522614 0.293009549 0.230593237 0.080044814 0.050494581 0.049568943
     6          7
 0.018940047 0.007826216
```

Boston Celtics (East 2nd seed) - Milwaukee Bucks (East 7th seed):

```
> test2 = data.frame(WLdiff = 0.134, PSdiff = -2.5, PAdiff = -6.4, netwins =
0)
> predict(m, test2, type = "probs")
     0          1          2          3          4          5          6
0.13680874 0.21900941 0.26637374 0.12511085 0.09134605 0.10117841 0.04214083
     7
0.01803196
```

Philadelphia 76ers (East 3rd seed) - Miami Heat (East 6th seed):

```
> test3 = data.frame(WLdiff = 0.097, PSdiff = 6.4, PAdiff = 2.4, netwins = 0)
> predict(m, test3, type = "probs")
     0          1          2          3          4          5          6
0.12661800 0.20903934 0.26536248 0.12908062 0.09611836 0.10836489 0.04573612
     7
0.01968018
```

Cleveland Cavaliers (East 4th seed) - Indiana Pacers (East 5th seed):

```
> test4 = data.frame(WLdiff = 0.025, PSdiff = 5.3, PAdiff = 5.7, netwins = -
2)
> predict(m, test4, type = "probs")
     0          1          2          3          4          5          6
0.03310589 0.07350022 0.15580710 0.12741192 0.13911326 0.24244694 0.15027245
     7
0.07834223
```

Figure 11: Eastern conference

Houston Rockets (West 1st seed) - Minnesota Timberwolves (West 8th seed):

```
> test5 = data.frame(WLdiff = 0.22, PSdiff = 2.9, PAdiff = -3.4, netwins = 4)
> predict(m, test5, type = "probs")
     0          1          2          3          4          5
0.372485911 0.301644034 0.186357881 0.056702010 0.033949786 0.032052571
     6          7
0.004879125 0.011928682
```

Golden State Warriors (West 2nd seed) - San Antonio Spurs (West 7th seed):

```
> test6 = data.frame(WLdiff = 0.134, PSdiff = 10.8, PAdiff = 7.7, netwins =
2)
> predict(m, test6, type = "probs")
     0          1          2          3          4          5          6
0.14695621 0.22819392 0.26643214 0.12113849 0.08689425 0.09475795 0.03901358
     7
0.01661346
```

Portland Trail Blazers (West 3rd seed) - New Orleans Pelicans (West 6th seed):

```
> test7 = data.frame(WLdiff = 0.013, PSdiff = -6.1, PAdiff = -7.4, netwins =
0)
> predict(m, test7, type = "probs")
     0          1          2          3          4          5          6
0.06458134 0.12936363 0.22377161 0.14526263 0.13066082 0.17821483 0.08769510
     7
0.04045005
```

Oklahoma City Thunder (West 4th seed) - Utah Jazz (West 5th seed):

```
> test8 = data.frame(WLdiff = 0, PSdiff = 3.8, PAdiff = 4.6, netwins = 2)
> predict(m, test8, type = "probs")
     0          1          2          3          4          5          6
0.05233624 0.10906762 0.20320951 0.14289104 0.13676660 0.20050696 0.10516098
     7
0.05006105
```

Figure 12: Western conference

We perform the exact same operations we did to get the two predictions earlier, and acquire the following predictions of this season's NBA first round playoff results:

Predictions	Winner	Likely results
Toronto Rapters - Washington Wizards	Rapters	4-0, 4-1, 4-2
Boston Celtics - Milwaukee Bucks	Celtics	4-0, 4-1, 4-2, 4-3
Philadelphia 76ers - Miami Heat	76ers	4-0, 4-1, 4-2, 4-3
Cleveland Cavaliers - Indiana Pacers	Pacers	3-4, 2-4, 1-4
Houston Rockets - Minnesota Timberwolves	Rockets	4-0, 4-1, 4-2
Golden State Warriors - San Antonio Spurs	Warriors	4-0, 4-1, 4-2, 4-3
Portland Trail Blazers - New Orleans Pelicans	Blazers	4-1, 4-2, 4-3
Oklahoma City Thunder - Utah Jazz	50%-50%	4-2, 4-3, 3-4, 2-4

Figure 13: Final predictions of the first round results

Up until the time this analysis was performed, four series in the first round have ended. We have three out of four correct win/loss predictions and three out of four actual results are predicted by the model with a probability over 20% [5] (See Figure 11 & Figure 12). The only failed prediction is the series between the Portland Trail Blazers and the New Orleans Pelicans. However, the actual result surprised every NBA data analyst. Every analyst thought that the Blazers would win the series but the result was the other way around, so we can treat this series as an exception.

4.4 Explanation of results and conclusion

In the previous section, with the help of R studio, we obtained different probabilities from factor 0 to factor 7 based on various combinations of test data. Now we would like to find out whether these probabilities, which we get by changing the test data, are statistically different from other probabilities. In other words, we would like to find out the importance of each predictor (win/loss difference, PS/G difference, PA/G difference, and net-wins).

For all regression models, we do not want the coefficient of any predictors to be zero. This is because the predictor would have no contributions to the response

if its coefficient is zero. Therefore, if zero lies in the confidence interval of a coefficient, then the corresponding predictor of that coefficient is not significant. The results of the 95% two-tailed confidence intervals are as follows:

```
> cbind(OR = coef(m), confint(m))
                OR       2.5 %      97.5 %
WLdiff  -2.1751424 -6.98374762  2.61894760
PSdiff  -0.2197330 -0.37312625 -0.06919947
PAdiff   0.2232147  0.06879573  0.37989541
netwins -0.1541273 -0.29630092 -0.01368086
```

Figure 14: 95% two-tailed confidence intervals

From Figure 14, we can see that zero is in the confidence interval of the coefficient of win/loss difference, therefore the predictor of win/lose difference is not significant. The other three confidence intervals of coefficients do not contain zero, so the points scored per game difference, the points allowed per game difference, and the net-wins are all significant predictors. OR stands for odds ratio, and the larger the odds ratio is, the more significant the predictor can be. From this we can see that points allowed per game are more important than points scored per game, which indicates that in NBA playoffs defense plays a more important role than offense.

Seeing as the accuracy of our prediction is over 93%, the ordinal logistic regression model is decent at predicting the NBA playoff results. However, there are still some other factors worth noticing that may affect the accuracy of the results. On one hand, from the perspective of my original data collection, using the points differential in the regular season match-ups instead of net-wins would be more convincing. In 2013-14 season, the Miami Heat went 0-4 versus the Brooklyn Nets in the regular season, which seems like a big level gap between the two teams. However, when you have a look at the point differential of these four games, the Miami Heat lost only one single point for each game.[5] This evidences that net-wins in the regular season can sometimes be a flawed predictor. On the other hand, injury is a very common situation in NBA. Sometimes a superstar on a team might be injured right before the playoff series. We again use the 2013-14 season to illustrate

the point. The San Antonio Spurs went 0-4 versus the Oklahoma City Thunder in the regular season, [5] however, their top defensive player Serge Ibaka was injured right before the series began. To sum it up, using points differential in the dataset and deleting exception points are two ways to make the model more accurate.

References

[1] wikipedia.org: *Simple Linear Regression*, Wikimedia Foundation, 10 Apr. 2018.

[2] Devore, Jay L., and Kenneth N. Berk.: *Modern Mathematical Statistics with Applications*, Springer-Verlag, New York, 2012

[3] Dobson, Annette J.: *An Introduction to Generalized Linear Models*, Chapman and Hall/CRC, 2002.

[4] NBA.com: *NBA Rules History*

[5] Basketball-Reference.com: *Basketball Statistics and History*