INST 737 Project Write-up

Zac Beitzel

Ty Brooks

Siwei Zhang

Need to Complete:  
1) Need to make sure these points are answered in the writeup when we are finished writing **(Everyone)**

* Why this is a good idea
* What you did
* Who did what
* Whether your technique worked or not

2) Need to generate R Markdown file, instead of the just the script that we currently have, and then save that to GitHub along with the R Script as well I guess **(Zac)**

As part of this….-Include code that checks/installs for the installation of R libraries that are required in the code.

-in your .Rmd submission include setwd at the top of your file, and don’t use paths for the rest of the code, just filenames. (That way for testing I can change the setwd line and run your code.)

3) Create a README file that indicates what is the purpose of each included file, and save to GitHub **(Ty, once all files are completed)**

4) Save a copy of the PowerPoint presentation, and save to GitHub **(Siwei) Done**  
5) Each team-member write the assigned sections in the write-up **(Everyone)**6) Save this write-up and a Word Doc and save to GitHub Repository **(Ty)**7) ZIP all completed files mentioned/needed in the project submission page (presentation slides, code, data, and any supporting materials, especially those required to run your code), and then submit that zipped file **(Ty, once all files are completed)**8) Once this write-up is completed, save the necessary parts as a Word Doc and save to GitHub **(Ty)**

**Project introduction (Zac Write)**

Our project goal was to create a predictive analytics model for NCAA Division 1 Men's Basketball games, in order to predict the winner when two teams play against each other during the Division 1 regular season and also during the postseason single elimination tournaments (March Madness and NIT).   
 There are a few practical applications for this type of predictive model. Recently in the college basketball ranks, team are devoting staff members to analytics positions in order to best understand what factors are the most significant forces in determining whether the outcome will be a win. NBA teams are very notorious for their analytics staff teams. This type of predictive model could also be used in order to strategically schedule the opponents for the upcoming season, in order to maximize the potential for number of wins. The gambling industry, which is very lucrative, is also interested in these type of predictive models, in order to generate odds for betting.   
 We decided that for our project, we would use various season-long statistical categories for teams, in order to predict which teams would win for any particular matchup. This specific type of model would be very useful for games that occur at the end of the season, such as in March Madness or The NIT.

**Data Used For Project (Siwei Write)**

For this project , we used two datasets (all of our data are from the website sports-reference.com):

Season Long Team Statistics for All NCAA Division 1 Schools (2016-17 and 2017-18 Seasons)

Data sets links: <https://github.com/brooksrelyt/INST737/blob/master/NCAA_Basketball_Season_Stats_2016_17.csv>

<https://github.com/brooksrelyt/INST737/blob/master/NCAA_Basketball_Season_Stats_2017_18.csv>

Team Matchups/Results for Regular Season and Postseason (2016-17 and 2017-18 Seasons)

Data sets links: <https://github.com/brooksrelyt/INST737/blob/master/NCAA_Matchups_2016_2017.csv>

<https://github.com/brooksrelyt/INST737/blob/master/NCAA_Matchups_2017_2018.csv>

(These data sets can also be accessed from out R script.)

Here are some examples of the features in these data sets:

Strength of Schedule, Conference, Home and Away - Wins and Loses, Pace Factor, Offensive Rating, Free Throw Attempt Rate, 3-Point Attempt Rate, True Shooting Percentage, Total Rebound Percentage, Assist Percentage, Steal Percentage, Block Percentage, Effective Field Goal Percentage, Turnover Percentage, Offensive Rebound Percentage, Free Throws Per Field, Goal Attempt…

**Data Preparation (Siwei Write)**

There are two kinds of data for the School Statistics---basic school stats and advanced school stats. The first thing we did was to merge these two kinds of data together by matching the school ID. Then we got the Season Long Team Statistics for All NCAA Division 1 Schools (2016-17 and 2017-18 Seasons). We decided to use the data set from 2016-17 season to train the model, and 2017-18 season to test our model.

The second data set is Team Matchups/Results data set, whose features are shown below:



In order to indicate which team won, we create a column called “Team\_A\_Won”. Because the results can only be “won” or “lost”, so it’s a binary outcome. If Team A won the matchup, we put “1” in this column, otherwise, put “0”. Then we duplicated the results from the training dataset (2016-17) by reversing the order of Team A and Team B to reduce the order effect.

Next step was to merge the Season Long Team Statistics data set with the Team Matchups/Results data set, since we need to contain the result of each matchup as well as the statistics of each team in the same file to create a prediction model. We removed all non-division 1 schools from the Matchups/Results dataset, and then made sure both datasets used the exact same wording for school names, as well as ensured that they used the same school ID. After that, we renamed the headers in the Season Long Team Statistics data set in order to make them easier to understand and more readable. All of these were done with R and Excel.

Here are the columns after merged the two data sets:









****

****

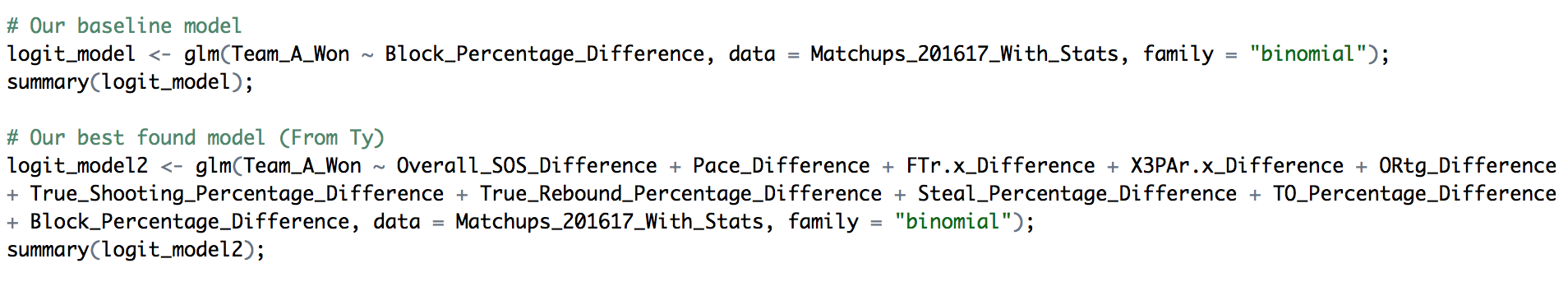
****

****

**Creating Models (Ty Write)**

We chose categories we thought would have an impact on the outcome of a game. We used these categories to create features for our model. After selecting a few categories, we calculated the difference between the two teams statistics. We created a model to help determine what statistical categories help predict which team wins. The features we chose included Offensive Rating, True Shooting Percentage, True Rebound Percentage, Steal Percentage, Turnover Percentage, Block Percentage, Three Point Attempt Rate, Free Throw Attempt Rate, Pace, and Overall Strength of Schedule.

We used logistic regression to build our model. Team\_A\_Won is our binary factor and we used each feature above as our continuous predictors. Our baseline and best model:



We next created a data frame that held the matchups for the 2017-2018 season, as well as the season-long statistics for the two teams in the matchup. We created new columns to compare the statistical categories between the two teams just like we did for the 2016-2017 season. We then tested our baseline model on the 2017-2018 season data. Using the predict function we created a new column with a prediction of the probability that Team A wins (1) or loses (0).



Using the probability created above, we made a guess of whether Team A Wins (1) or Team B Loses (0). We did this by giving our prediction values that are greater than 0.5 a value of 1 and assigning any values that are less than 0.5 a value of 0. We then displayed the results in a contingency table.

As a result, our baseline model predicted 57.7% and our best model predicted 72.7%. An overall increase of 15% of the total 5,540 games played during the 2017-2018 Regular Season and Postseason.

**Which Group Members Did What Tasks (Everyone Fill In Items Here)**

* Find and Download Datasets  
   -Ty, Zac and Siwei downloaded the datasets to be used from Sports-Reference.com, they also originally manually compiled postseason matchups/results, before later they discovered that all season matchups/results were available on Sports-Reference
* Create GitHub Repository To Store All Data and Files  
   -Ty created the repository using his GitHub account, and all three group members edited the files on GitHub as needed, when they were edited
* Clean the Datasets /Data Preparation  
   -Zac removed all non-division 1 schools from the matchups/results dataset, and then made sure both datasets used the exact same wording for school names, as well as ensured that they used the same school ID  
   -Siwei and Zac renamed the headers in the season-long stats dataset in order to make them easier to understand and more readable  
   -Ty duplicated the results from the training dataset by reversing the order of Team A and Team B in order to reduce the effect of bias of the order of Team A and Team B
* Write R Script To Import Datasets  
   -Zac wrote R script to import the csv files
* Write R Script To Merge Datasets  
   -Zac wrote R script to merge the Season Long Team Statistics dataset and the matchups dataset together, by matching the Team ID  
   -Ty and Zac created a column indicating the winner of each matchup
* Write R Script To Create Model  
   Write R Script To Create A Baseline Mode  
   -Zac and Siwei evaluated the variables and chose one of them that was used in the baseline model  
   -Zac created the baseline model in R  
   Write R Script To Create Our Best Model  
   -Ty, Siwei and Zac evaluated all the features which important for making the prediction, then tested to see if they are significant in our model   
   -Zac wrote R script to calculate the differences of these significant features between Team A and Team B   
   - All group member created and tested different logistic regression models with different combinations of multiple variables.
* Write PowerPoint Slides  
   -All group member collaboratively helped with the writing, evenly splitting up the duties  
   -Siwei saved it as a PowerPoint file, from our collective Google Slides document
* Write Final Project Write-up  
   -All group member collaboratively helped with the writing, evenly splitting up the duties
* Generate R Markdown File From R Script  
   -Zac will convert the R script that we used for the project to an R Markdown file that is needed for project submission
* ZIP All Project Items Together and Submit  
   -Ty is responsible for collecting all of the project files, making sure they are on GitHub, zipping them into one file, and then submitting the project files to Patrice