PSTAT 131 Term Project Draft

Sophia Sternberg ([redacted]), Nicolette Phillips ([redacted]), Sebastian Naibaho ([redacted])

6/4/2020

AVP Beach Volleyball Data Analysis

Introduction

What makes a winning player and team in the AVP beach volleyball tournaments? This is an area of interest for many volleyball fans and players who would like to predict winning teams in a tournament. There are many factors that go into the success of a player or team, and we hope to look at variables such as player age, country of teams, and a team's overall win-loss ratio to develop a model to answer this question. Through the model building process, our goal is to develop a model that will accurately predict whether a team has more wins than losses given our predictor variables while valuing models with low test errors.

Data Overview

https://github.com/BigTimeStats/beach-volleyball

The data for our project comes from Adam Vagner's public Github page for anyone to access. The main csv file contains tournament data from 2002 to 2017. The file we downloaded from the Github page had information on two different tournaments (FIVB and AVP). However, we only wanted AVP data, so we subsetted it further to only include that tournament series. We also altered the dataset slightly in order to extract individual and team statistics. The only analyses that we were able to find using Vagner's public datasets were his own Tableau visualizations.

Our outcome variable will be a binary value that indicates whether the winning team had more wins than losses. There are a total of 33 potential predictors in the original dataset, comprising 19 numerical and 14 categorical variables. However, most of these predictors were unusable due to missingness or overlap with the outcome variable. For example, the outcome variable winning_record_w is based upon the number of team wins and losses, so the variables team_wins_w, team_wins_l, team_losses_w, and team_losses_l could not be used. Examining a summary table of missing values revealed a large number of missing values for the game statistics columns, so data cleaning is necessary before we begin our analysis. The dimensions of the original data are 1895 by 80.

```
## [1] 1895   80
```

To prepare our data for the model building, we first created a binary variable that will be our response. This binary variable identifies whether the winning team has more wins than losses overall. A win/loss ratio greater than 1 is indicated by 1, and the remaining entries are indicated with a 0. Therefore we are looking to predict whether a team will have more wins than losses(1) or more losses/the same number of wins and losses(0). After adding this variable to our original data, we began to remove columns with missing values as well as variables that we deemed unnecessary. The output below displays the 10 variables that remain in our cleaned dataset.

```
##  [1] "year"           "gender"         "w_p1_country"   "w_p2_country"
##  [5] "l_p1_country"   "l_p2_country"   "avg_age_w"      "avg_age_l"
##  [9] "winning_record_w" "winning_record_l"
```

After creating our cleaned dataset, we once again checked for missing values to ensure all variables can be used. The output below shows that the remaining variables contain no missing values.

```
##             year           gender     w_p1_country     w_p2_country 
##                0                0                0                0 
##     l_p1_country     l_p2_country        avg_age_w        avg_age_l 
##                0                0                0                0 
## winning_record_w winning_record_l 
##                0                0
```

Next, we looked into the class of our remaining variables in order to give us an idea about which models will work best with our model. This also allows us to identify any data types that should be changed before moving onto the model building process.

We decided to change the class of columns of numbers into numeric for future analysis. Following this change, the class types are as follows:

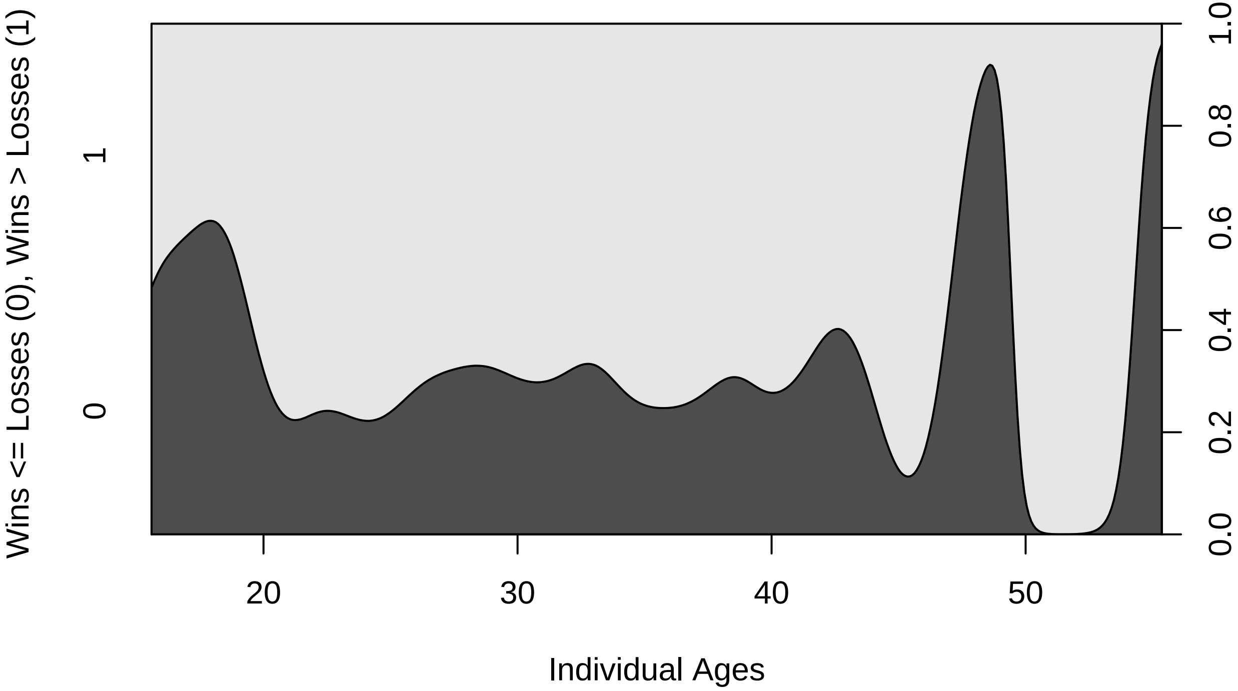
```
##             year           gender     w_p1_country     w_p2_country 
##        "integer"         "factor"         "factor"         "factor" 
##     l_p1_country     l_p2_country        avg_age_w        avg_age_l 
##         "factor"         "factor"        "numeric"        "numeric" 
## winning_record_w winning_record_l 
##        "numeric"        "numeric"
```

After these steps the dataset is clean and ready for analysis, so we decided to do some preliminary exploration into the data. The new dimensions of the data are 1895 by 10, which reflects the removal of columns when cleaning the data. The table summarizes the statistics of each of the 10 variables, to give an idea about what each variable looks like and how we can use it in our analysis.

```
## [1] 1895   10
```

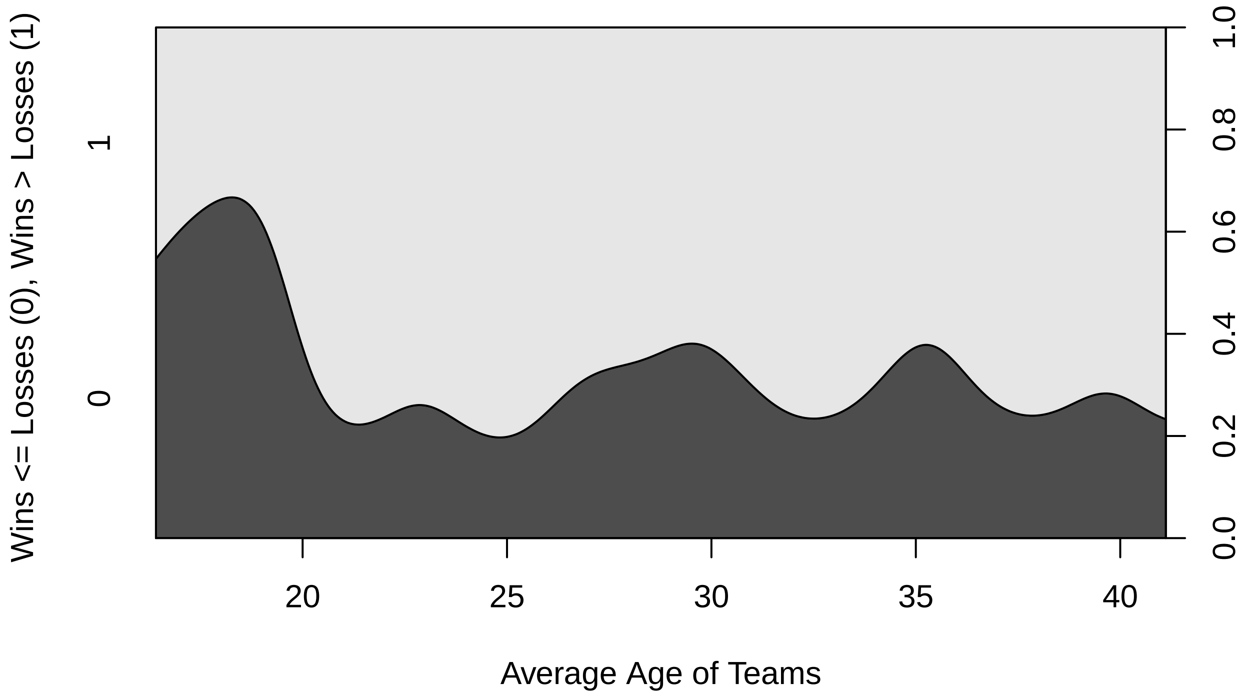
```
## avpdata_clean 
## 
##  10  Variables      1895  Observations
## --------------------------------------------------------------------------------
## year 
##        n  missing distinct     Info     Mean      Gmd 
##     1895        0        2    0.749     2016   0.4998 
## 
## Value       2016  2017
## Frequency    975   920
## Proportion 0.515 0.485
## --------------------------------------------------------------------------------
## gender 
##        n  missing distinct 
##     1895        0        2 
## 
## Value          M     W
## Frequency   1010   885
## Proportion 0.533 0.467
## --------------------------------------------------------------------------------
## w_p1_country 
##        n  missing distinct 
##     1895        0       14 
## 
## lowest :                 Belarus        Brazil         Canada         England
## highest: Poland         Puerto Rico    Spain          United States  Virgin Islands
##
## (1, 0.001), Belarus (2, 0.001), Brazil (3, 0.002), Canada (22, 0.012), England
## (5, 0.003), France (4, 0.002), Germany (3, 0.002), Guatemala (5, 0.003), New
## Zealand (1, 0.001), Poland (31, 0.016), Puerto Rico (5, 0.003), Spain (1,
## 0.001), United States (1805, 0.953), Virgin Islands (7, 0.004)
## ---------------------------------------------------------------------------------
## w_p2_country
##        n  missing distinct
##     1895        0       16
##
## lowest : Brazil          Canada          China           Czech Republic  England
## highest: Slovak Republic Spain           United States   Venezuela       Virgin Islands
##
## Brazil (46, 0.024), Canada (10, 0.005), China (19, 0.010), Czech Republic (6,
## 0.003), England (11, 0.006), Germany (6, 0.003), Greece (1, 0.001), Guatemala
## (5, 0.003), Japan (5, 0.003), Poland (7, 0.004), Puerto Rico (30, 0.016),
## Slovak Republic (1, 0.001), Spain (3, 0.002), United States (1733, 0.915),
## Venezuela (2, 0.001), Virgin Islands (10, 0.005)
## ---------------------------------------------------------------------------------
## l_p1_country
##        n  missing distinct
##     1895        0       15
##
## lowest :                Belarus        Brazil         Canada         England
## highest: Poland         Puerto Rico    Spain          United States  Virgin Islands
##
## (2, 0.001), Belarus (1, 0.001), Brazil (2, 0.001), Canada (7, 0.004), England
## (4, 0.002), France (2, 0.001), Germany (2, 0.001), Greece (1, 0.001), Guatemala
## (3, 0.002), New Zealand (2, 0.001), Poland (37, 0.020), Puerto Rico (6, 0.003),
## Spain (1, 0.001), United States (1817, 0.959), Virgin Islands (8, 0.004)
## ---------------------------------------------------------------------------------
## l_p2_country
##        n  missing distinct
##     1895        0       18
##
## lowest : Brazil          Canada          China           Cuba            Czech Republic
## highest: Slovak Republic Spain           United States   Venezuela       Virgin Islands
##
## Brazil (29, 0.015), Canada (9, 0.005), China (16, 0.008), Cuba (1, 0.001),
## Czech Republic (7, 0.004), England (9, 0.005), Germany (5, 0.003), Greece (1,
## 0.001), Guatemala (4, 0.002), Iceland (1, 0.001), Japan (4, 0.002), Poland (11,
## 0.006), Puerto Rico (23, 0.012), Slovak Republic (1, 0.001), Spain (2, 0.001),
## United States (1761, 0.929), Venezuela (2, 0.001), Virgin Islands (9, 0.005)
## ---------------------------------------------------------------------------------
## avg_age_w
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##     1895        0      896        1    29.67    5.631    21.31    23.16
##      .25      .50      .75      .90      .95
##    25.77    30.33    33.13    36.00    37.50
##
## lowest : 16.35592 16.43258 16.51745 16.72964 16.87611
## highest: 40.67214 41.02669 41.12936 41.16769 41.17317
## ---------------------------------------------------------------------------------
## avg_age_l
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##     1895        0     1427        1    29.63    5.934    20.90    22.63
##      .25      .50      .75      .90      .95
##    26.17    29.84    33.18    36.23    38.22
##
## lowest : 14.86242 15.30185 16.28611 16.35592 16.43258
## highest: 43.59343 44.62012 48.63792 49.74401 53.28131
## ---------------------------------------------------------------------------
## winning_record_w
##        n  missing distinct     Info      Sum     Mean      Gmd
##     1895        0        2    0.621     1340   0.7071   0.4144
##
## ---------------------------------------------------------------------------
## winning_record_l
##        n  missing distinct     Info      Sum     Mean      Gmd
##     1895        0        2    0.705      715   0.3773   0.4701
##
## ---------------------------------------------------------------------------
```

One area that we would like to look at more in depth is the age variables. We can either use the individual player ages or the average team ages when predicting our outcome variable, so we would like to determine which might be more significant. To do this we plotted smoothed proportions of the binary response within various levels of the age variables. The conditional density graph using individual ages of winning players is shown below.



In the graph, we can see a spike in the proportion of 0s(more losses than wins) for ages under 20, and also some spikes in the higher ages. The higher ages have extreme spikes in both 0s and 1s, which is likely because there are few individuals that have ages between 45 and 60. the fluctuation in this range is not ideal for our analysis since the proportion is likely based on only 1 or 2 players, rather than it being representative of the overall age group. There is also very little fluctuation between ages 20 and 45. This is also not very ideal since it suggests that indivdual age may not have a great effect on our outcome variable. The spike in the 0-20 year old age range is likely due to inexperience for players just starting out in the sport.

We believe this proves a very interesting point with AVP data. This shows that age does not necessarily warrant a competitive advangtage in the sport beyond ages under 20 and potential outliers in the older age range. This may also be caused by older(around 40 years old) players partnering up with younger(around 20-30 years old) players to get the best of both worlds (knowledge and expertise from an older player and youthfulness/inventiveness from the younger).



Looking at the graph above, which uses the average team age instead of the individual player ages, it appears to be more ideal for our analysis. We can see that there are more clear peaks across the entire range of ages, and this solves the potential issue of outliers in the older ages. After comparing the two graphs, we have decided to use average team age as a predictor instead of individual ages moving forward.

Next we looked into the outcome variable. The mean(.707124) and counts of each outcome show that the outcome 1 is more common than 0. Therefore, it is more likely that the winning team of an individual game has more wins than losses, which intuitively makes sense.

```
##
##    0    1
##  555 1340
```

We also looked at the outcome variable by groups, separating by country, gender, and whether the losing team had more wins than losses. The first group shows us which countries had more wins than losses over all their teams. One interesting statistic from this grouping is that of the 22 Canadian teams that won at least once, they all had more wins than losses. When looking at gender, we can see that men's winning teams are more likely to have more wins than losses than the women's winning teams. The last group is interesting because it shows that it is more likely for the winning team to have more wins than losses if the loser also has more wins than losses. However there are a greater number of losing teams with more losses than wins when compared to the losers with more wins than losses.

```
## # A tibble: 14 x 3
##    w_p1_country     count  mean
##    <fct>            <int> <dbl>
##  1 ""                   1 0
##  2 "Belarus"            2 1
##  3 "Brazil"             3 0.667
##  4 "Canada"            22 1
##  5 "England"            5 0.8
##  6 "France"             4 1
##  7 "Germany"            3 0.667
##  8 "Guatemala"          5 1
##  9 "New Zealand"        1 0
## 10 "Poland"            31 0.387
## 11 "Puerto Rico"        5 0
## 12 "Spain"              1 0
## 13 "United States"   1805 0.711
## 14 "Virgin Islands"     7 0.429
```

```
## # A tibble: 2 x 3
##   gender count  mean
##   <fct>  <int> <dbl>
## 1 M       1010 0.732
## 2 W        885 0.679
```

```
## # A tibble: 2 x 3
##   winning_record_l count  mean
##              <dbl> <int> <dbl>
## 1                0  1180 0.614
## 2                1   715 0.860
```

This analysis allowed us to get an overall idea about the main predictors and outcome variables that we will be using in our model building process. The main predictors we used were the average age of the winning and losing teams, gender, countries of the winning and losing teams, and the binary variable relating to whether the loser had more wins or losses.

Methods

Our data will be split into training and test sets using the tournament year. We will use 2016 data as our training set and 2017 data as our test set. We did this in order to keep the number of wins and losses seperated by year and to prevent overlap with teams. From this data, we will have 975 training observations and 920 test observations. The dimensions of each set are shown below.

```
## [1] 975  10
```

```
## [1] 920  10
```

The first method we plan to explore is kNN analysis with cross validation. To prepare the data for knn analysis, we will create a subset of the data that includes only numeric predictors. We will use kNN to determine the training and test error when k=2, 10, and 100. Then we will use cross-validation in order to find the best k. We will then use this k to compute the test error for our kNN model.

Next we will perform linear discriminant analysis and find the test error. This process looks for linear combinations of variables which best explain the outcome. During this process we will convert the gender variable in the clean dataset to binary, with 0 representing male and 1 representing female. Gender is a predictor that could be significant, and making it binary makes it easier to use in future analysis. We will calculate the test error of the lda model to compare with the other models we are exploring.

Continuing our analysis, we will perform logistic regression on the trained data. The logistic regression will give us information on the most influential variables that would best fit–yet do not overfit–our data from the significant p-values it obtained for our predictor variables. We will then compute the test error for comparison.

Next we will explore a decision tree model. Decision trees will allow us to investigate how our predictor variables influence whether the team has more wins than losses or not.The test error for the optimally-pruned single tree model will be calculated after we create and prune the decision tree. Next, we will apply bagging to the data by creating a random forest with m=p and once again calculate the test error. Lastly, we will create a random forest with m=p/3 to further refine our test error. To conclude, we will compare the test error values from each model in order to decide the best model to predict whether a team will have more wins than losses.

Model Building:

This analysis will employ supervised machine learning to build models through knn, linear discriminant analysis, logistic regression and decision trees to help us train our classifiers. We began by creating a model employing kNN. In our knn analysis, we are arbitrarily using k = 2, 10, 100 and viewing their MSE's to get an idea for the test and training error with varying k. The training error rates were computed to be 0.08205128, 0.1169231, 0.2133333 for k=2, 10, and 100 respectively.

```
## [1] 0.08205128
```

```
## [1] 0.1169231
```

```
## [1] 0.2133333
```

The test error rates were computed to be 0.2467391, 0.173913, 0.1619565 for k=2, 10, and 100 respectively.

```
## [1] 0.2467391
```

```
## [1] 0.176087
```

```
## [1] 0.1673913
```

Next, we used cross validation to find the most optimal k to use (instead of arbitrarily choosing k). With k ranging from 1 to 100, we found through cross validation that using k=3 performs the best. To get the test error, we have to train the kNN classifier on the training set and predicted winning_record_w on the test set. Using k=3, we computed the test error to be 0.176087.

```
## [1] 0.176087
```

Next, we used lda to estimate the classification boundary. Prior to our analysis we changed gender into a binary variable to add it as a predictor. We used lda to fit our predictors (average player age per winning and losing team, whether the losing team had more wins than losses, and gender) against our response variable which is binary (winning_record_l). We chose these predictors because they had no crossover with the response variable, unlike the number of wins and losses of teams and players. We only use lda because qda requires more data (in terms of observations and variables) that our dataset simply just doesn't have. The test error of the LDA model was computed to be 0.2586957, which is higher than the error from the kNN model.

```
## [1] 0.2586957
```

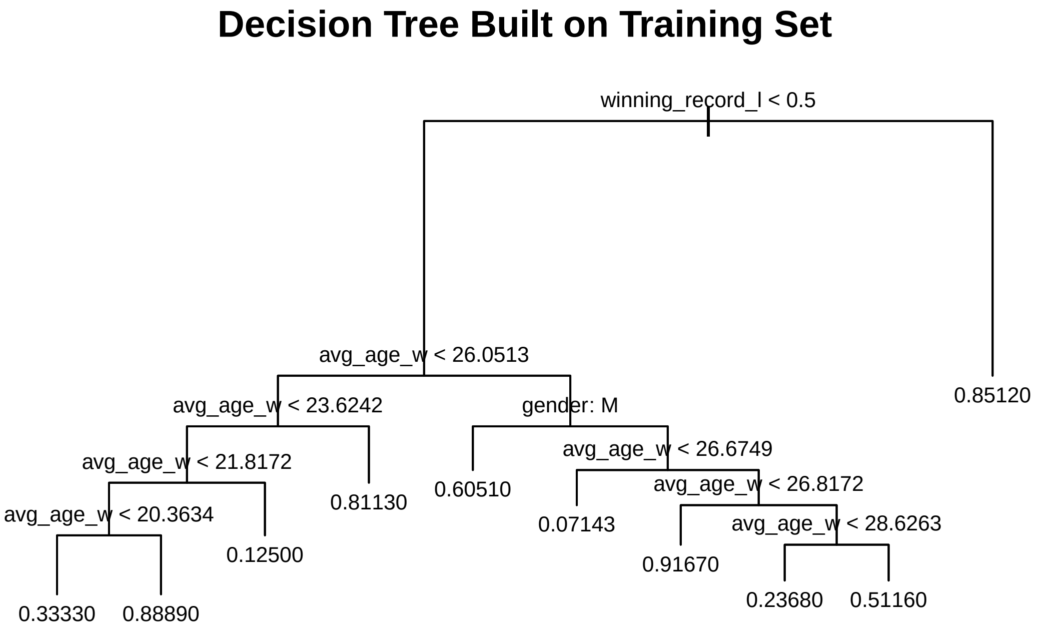
Next, we built a logistic regression model to compare to our previous models. The logistic regression function we built carries the same variables as our lda function earlier in this exploration. We purposely wanted to examine the differences between logistic regression and lda by carrying over the same variables throughout. In this application, whether the losing team had more wins than losses, average player age (of winners), and gender (of winners) seem to be significant variables. Our calculated test error for this logistic regression model is 0.3353846, which is higher than our lda model.

```
##
## Call:
## glm(formula = winning_record_w ~ avg_age_w + avg_age_l + gender +
##     winning_record_l, family = binomial, data = train_clean)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.1548  -1.2554   0.5751   0.9894   1.2617
```

```
##
## Coefficients:
##                    Estimate Std. Error z value Pr(>|z|)
## (Intercept)       1.404e+00  6.444e-01   2.179   0.0293 *
## avg_age_w        -2.957e-02  1.523e-02  -1.941   0.0523 .
## avg_age_l        -7.656e-05  1.441e-02  -0.005   0.9958
## genderW          -3.951e-01  1.561e-01  -2.531   0.0114 *
## winning_record_l  1.441e+00  1.757e-01   8.200 2.41e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1228.4  on 974  degrees of freedom
## Residual deviance: 1142.0  on 970  degrees of freedom
## AIC: 1152
##
## Number of Fisher Scoring iterations: 4
```

```
## [1] 0.3353846
```

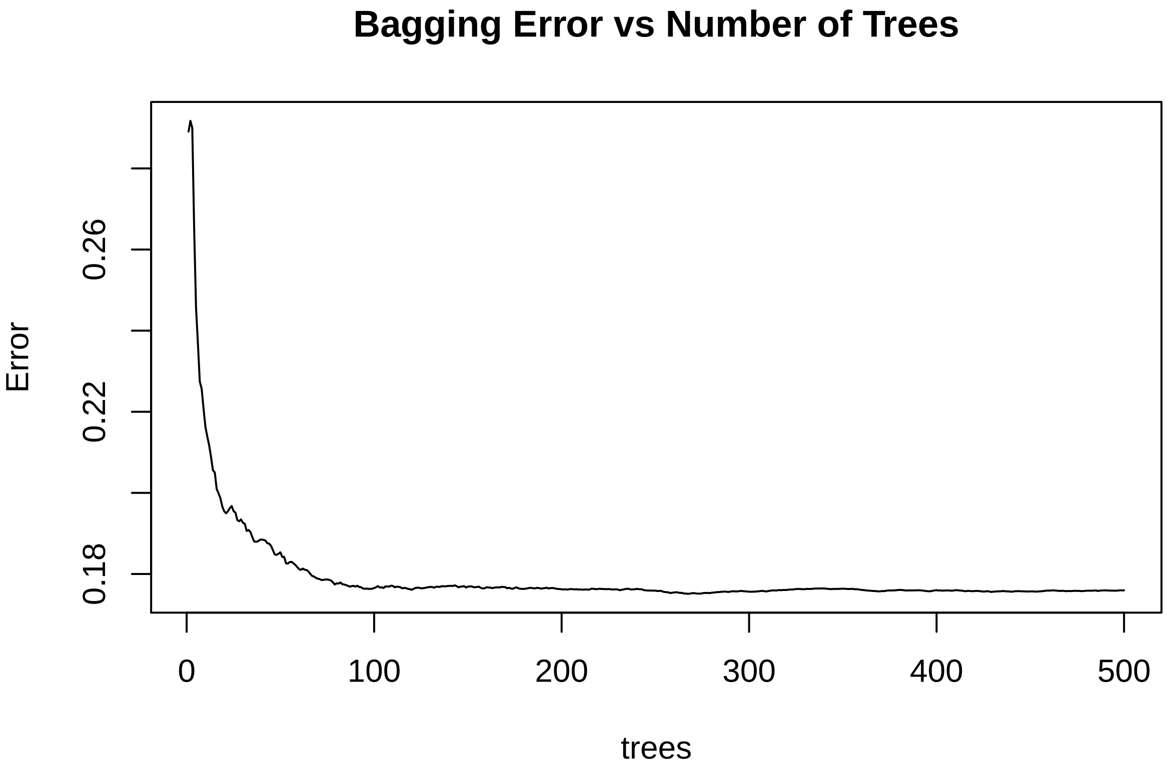
Next, we evaluated a decision tree model. Once again we used our cleaned training data to evaluate the relationship between the binary response relating to the win/loss ratio and the predictor variables average age of winning team, average age of losing team, gender, loser's winning record, and winning countries. The single tree decision tree is displayed below.



```
## [1] 0.2127222
```

Calculating the test error for this tree results in 0.2127222. Next we pruned the tree in order to determine the best size of the tree. From pruning we discovered that the ideal size of the tree is 10, which is the size of the original tree. Therefore the test error remains unchanged.

Next, we applied bagging to the AVP data to compare to our test MSE we obtained from other methods. This is done by creating a random forest with m=6. the graph below displays the changes in error with differing values

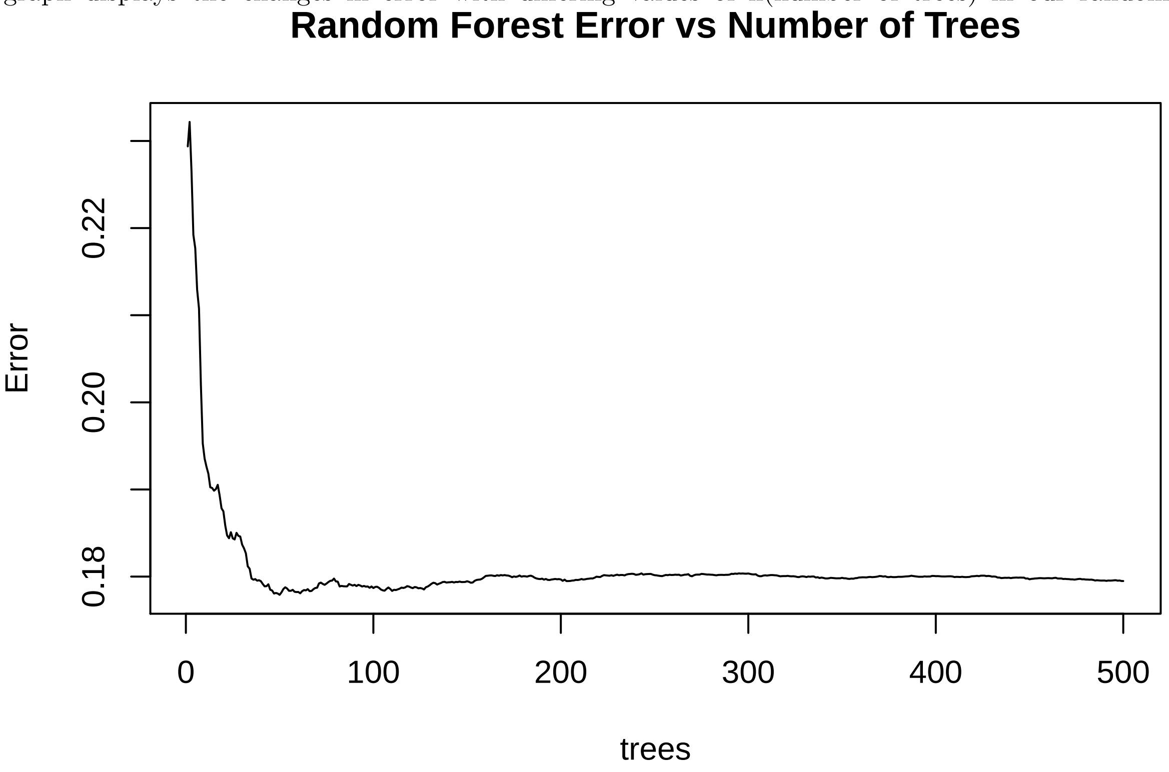


of n(number of trees).

The test set error rate associated with the bagged tree is 0.2331492, higher than that obtained using an optimally-pruned single tree(0.2127222).

```
## [1] 0.2331492
```

Lastly, we proceeded with creating a random forest to further refine our test error rate. The graph displays the changes in error with differing values of n(number of trees) in our random forest.



The

test set error rate associated with the random forest is 0.1897039, which is lower than the error from the bagged tree(0.2323647) and the optimally-pruned single tree(0.2127222).

```
## [1] 0.1897039
```

##Conclusions When it comes to model validation methods, the method with the lowest test error is considered to be the best model, all other factors aside. The test errors of the various models we explored is displayed in the table below.

```
##                             Test Error
## kNN                          0.1760870
## LDA                          0.2586957
## Logistic Regression          0.3353846
## Pruned Single Decison Tree   0.2127222
## Bagging                      0.2331492
## Random Forest                0.1897039
```

Overall, our analyses found that the logistic regression method resulted in the highest test error out of all the methods used. Therefore logistic regression would be the worst model when looking to predict whether a team had more wins than losses. Out of all of the classification models we have utilized throughout this report, the kNN model with k=3 yielded the lowest test error when compared to other models. Therefore, the kNN model would perform the best when looking to predict whether a team had more wins than losses. The model has a test error of 0.176087, which means the model accurately predicts approximately 82% of the test observations.

Some limitations from this project can be largely attributed to the dataset we chose. While we were able to accomplish several preprocessing tasks with our data, we were still limited with our variables due to missingness and lack of numeric data. For example, we need to remove player height data due to the high volume of missing data. Height is a valuable asset in volleyball, so it could have been a valuable predictor had the data been more complete. In the future, we can look to aggregating multiple datasets from multiple sources to employ better data manipulation and analyses for classification.