Peter Atsaves

Data Analysis Final Project

1 Problem and Data Description

Briefly describe the data mining problem and the data

The problem I have is to figure out the winner of each game of the NCAA March Madness tournament. This involves using numerous amounts of data for each game and determining which factors are the most important in order to determine the winner.

Among the data are 20 sets of data. One of the most important is the data of the actual games from the tournament. This includes the points, field goals made, and a lot of other stats from the winners and losers. Each game also has a winning and losing team id which I am able to link to the seeds table based on season to figure out the seed of the team. Another important one is the regular season data. This contains the games played of the teams during the regular season. This allows us to look at how their regular season performance can impact their tournament performance. We can also look at other pieces of data such as performance in conference tournaments, overall seeding, coaches, conferences, etc.

2 Data Preprocessing & Analysis

2.1 Handling Missing Values

2.2 Exploratory Data Analysis

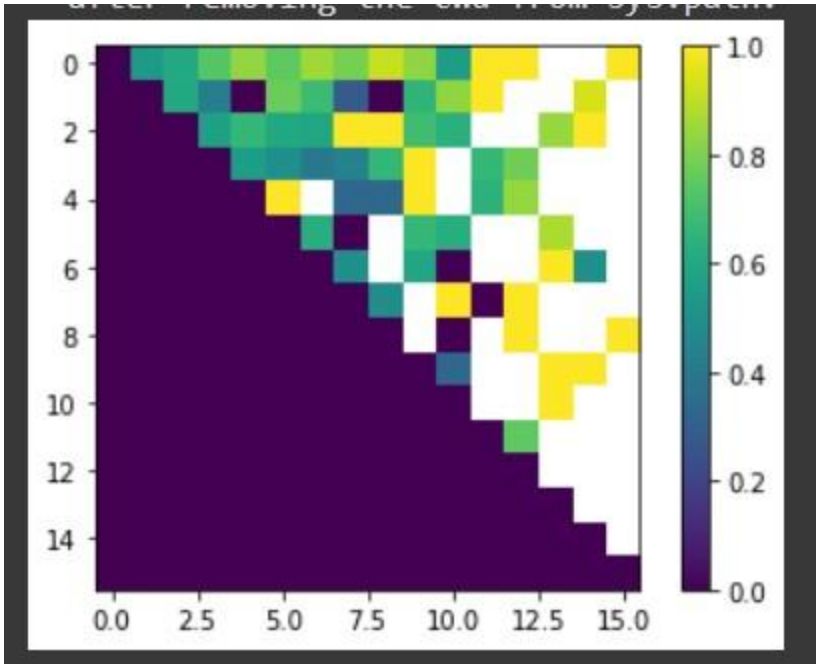
In terms of missing values, there weren't any in the datasets because each game has information about it. The only thing that would be missing would be certain categories from 1985 to 2002. There are files of compact and detailed results for data about the regular season and ncaa tournament. From 1985 to 2002, for each game there are only 8 columns of data, whereas

after 2002 there are 34 columns of data. Instead of trying to fill in this data by adding means of averages for these values, I decided not to touch the data. This is because the data is already functional and there are no missing values, just less columns. It just means that whenever I wanted to look at the detailed results, I only looked at the data from 2003 to present.

We learned a lot of things from looking at the data. One of them is the overall seeding of teams. This is different from the typical 1-16. Teams could be ranked from 1-353rd best in the nation, and using this data we were able to see how a number of teams who were above a certain ranking never won a single tournament game in the history of the tournament.

One correlation I was hoping to see was if teams that won more games later in the regular season performed better in the tournament. Unfortunately, this prediction was false, and there was only a slight difference. More specifically, I looked at 2 and 15 seeds and the difference between the 2 seeds that won and lost this matchup was minimal, and the same went for 15 seeds.

I also looked at how certain seeds play against each other. We were able to notice a lot of things from this. Whenever a 12 and 13 seed matchup, the 12 seed wins that matchup a majority of the time, despite there only being one ranking difference between them. Another surprising one was that 1 seeds don't perform very well against 11 seeds.



We get to see those win percentages here and not for 1 vs 16 it is completely yellow since 1 beats 16 nearly every single time. There are many other interesting ones.

As far as how far teams make it in the tournament, there are always a couple low ranked teams that make it far, but the vast majority are 1,2, and 3 seeds. We also noticed that no 13 seed or higher made it to the final four. So if we ever predict a 13 seed to win in the first two rounds, we would expect them to lose in the elite eight.

When looking at regular season stats vs tournament stats, we found that the average winning and losing scores were extremely similar. I was hoping to find a difference to see if for example teams that played better defense won more games, but I didn't find this to be the case.

I also looked at the teams that have been to the tournament the most. North Carolina, Duke, and Kansas among the leaders. If we are ever in doubt of a pick, we may choose one of these schools to advance based on the fact that going to the tournament the most could mean these programs have more experience and better coaching than their opponents, even if they have a low seed in the tournament.

The NCAA tournament comes down to luck a lot of the time, but maybe that luck is because of factors we haven't considered. I hope to come up with a solution to this problem in phase ii to determine the winners of each game for the next tournament.

We can see that no seed ranked above 280 has ever won a tournament game, and only 9 teams ranked above 200 have ever won a tournament game. If a team above 200 is in the tournament, it is very unlikely they will win a game. From teams 70-199, only 41 of them have won a tournament game. The rankings we are using are from the system "BIH" and are from day number 133 which is the last day that rankings are made.

3 Algorithm and Methodology

3.1 Logistic Regression

I thought this was the best choice to use from the start since this is a classification problem and because logistic regression does really well with binary classification problems. I arranged the data so that the final column would be 1 or 2, which represents whether team 1 or 2 wins, instead of having the models guess the amount of points for each team and then choosing the winner.

3.2 Random Forest Classifier

I also used random forest classifier to make decisions on the data. I thought it would be beneficial in determining things like a 1 seed should almost always beat a 16 seed, no matter the records of the teams or their average points per game.

3.3 K-Means

Even though K-means is unsupervised, I wanted to try it out to see if there were other patterns I wasn’t considering or looking into.

3.4 Naive Bayes

I tried to use naive bayes also because it is also a good classifier. The only problem is that a lot of the data points are not independent from each other which affects the algorithm's performance. For example, points per game and seeding might seem unrelated, but a better ranked team is more likely to score more points per game.

4 Experiments and Results

Before I selected any features and started data manipulation, I started the dataframe with just the season and the id of the winning and losing team. I then created what would be the results column which consists of the winning team, and this would be a value of either 1 or 2. Since the winning team is already known, I changed the columns to team 1 and team 2. Then I randomly had around 50% of the teams switch places and assigned the winning team to the results column. This way, the computer won't already know the winning team.

I also tested linear regression just to see what our initial accuracy would be, and it turned out to be 50%, which is as good as a coin flip. This makes sense because you can't make a good prediction without relevant data.

```
0.5150862068965517
```

For each team, I got the ranking (1-16) and added that to the dataframe for both teams. The ranking is a key factor in determining who wins. Next, I wanted to find the all time winning percentage of a seed. For example, for a 1 seed, this would be the percent of times it wins against every team it has faced. This would also be really useful in cases where seeds that don't usually match up, play against each other such as a 1 and a 12 seed.

At this point, I just ran the linear regression with only these few features selected, and got an accuracy of 70%. This is actually pretty good for only adding a few features, although they were important ones.

```
0.7047413793103449
```

I wanted to explore the regular season more. I was able to look at day number 133, which is the last day before tournament play. I got the ranking of each team playing in the tournament (this is the overall ranking, different from 1-64 since there are many teams outside this that make the tournament). After adding this, there is hardly much improvement in the model, so we need to find more features.

```
0.7068965517241379
```

Looking more into the regular season, I got the average points scored per game of each team and added this to the dataframe. I also added the all time win percentages for the seeds. For example, if team1 is seed 3 and team2 is seed 7, the win percentage of 3 seeds vs 7 seeds would be added to the dataframe. Average scoring margin was also added for each team. Some teams can have similar records, but some can be much more dominant than others during the regular season.

After just using linear regression, I tested out 3 other classification models. Random Forest Classifier was comparable with logistic regression, but k means and naive bayes did not do well at all.

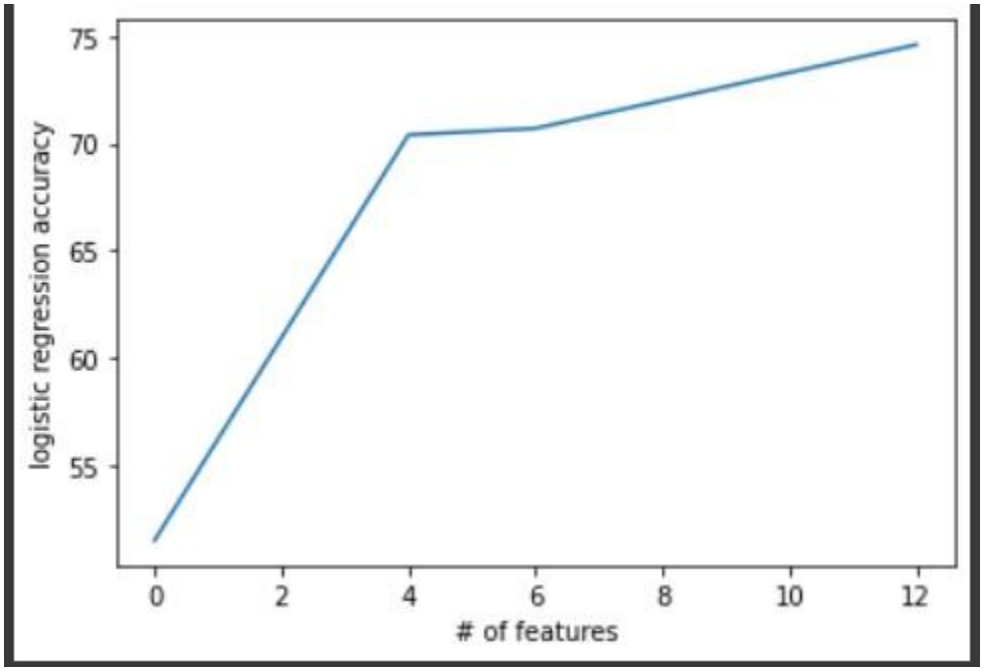
Logistic Regression: `0.7456896551724138`

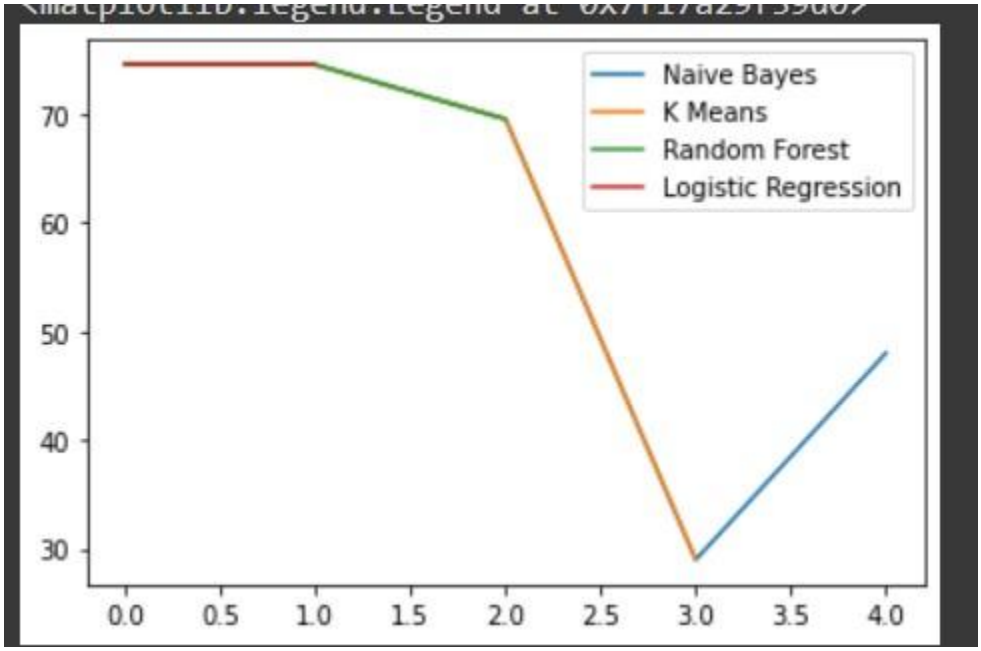
Random Forest: `0.6896551724137931`

K Means: `0.28657747086750107`

Naive Bayes: `0.47844827586206895`

It’s not surprising that K Means and Naive Bayes did poorly as a lot of the variables have some dependence on each other, but I was happy to see that logistic regression improved to 74.5%, which is a healthy increase. I decided to stick with this for the rest of the testing.





One feature that appeared to be very important was the average win margin during the regular season. This gives a good explanation of how dominant a team was over the course of the regular season which is a large amount of games. Teams that are more dominant may be better than teams ranked above them who only won games due to small margins. The opposite is true for teams that barely lost many games and don't have the record or ranking to show for it. Another key feature is that all time seed win percentages. There seems to be patterns when certain seeds match up. We hardly see a 7 or 6 seed matchup or even beat a 1 seed, but there always seems to be a couple 8 seeds that beat a 9 seed and then a 1 seed the next round. Adding these features lead to a big improvement in the model.

5 Summary and Conclusions

Overall, I found logistic regression to be the best performing classifier, which is what I expected coming into this. It had the best accuracy consistently and was able to reach a high of guessing 75% of the games correctly. The testing was done on the final 20% of the dataset. 75% is not a great accuracy, but it is a lot better than 50% which would be the odds of randomly picking a team to win for each game.

From the data preprocessing I did, I was able to realize what factors contributed to certain teams winning. There is a lot of luck in March Madness, but there were a lot of patterns and things I found surprising when I looked at the data. I was able to execute on this using logistic regression. I used factors that would be impacted over a long period of time as opposed to sometime more short term. For example, I noticed more correlation in winning when comparing to regular season stats (spanning over 30 games) and actual ranking (1 to 300+) as opposed to what teams are "hot" at the end of the season. I also used features consisting of all time performance of teams and seeds. Each year, the teams are always different, but the results end up similar so I knew there had to be a lot of that consistency from the seeding and underrated strength of teams.

This project went well. Ideally, I would've been able to guess 80+% of the games correctly, but there are so many factors that come into play that I can't account for. I am happy with guessing 3 out of every 4 games correctly, as that is much better than I typically do.