Dustin Casey

Original Case Study

DSC 550

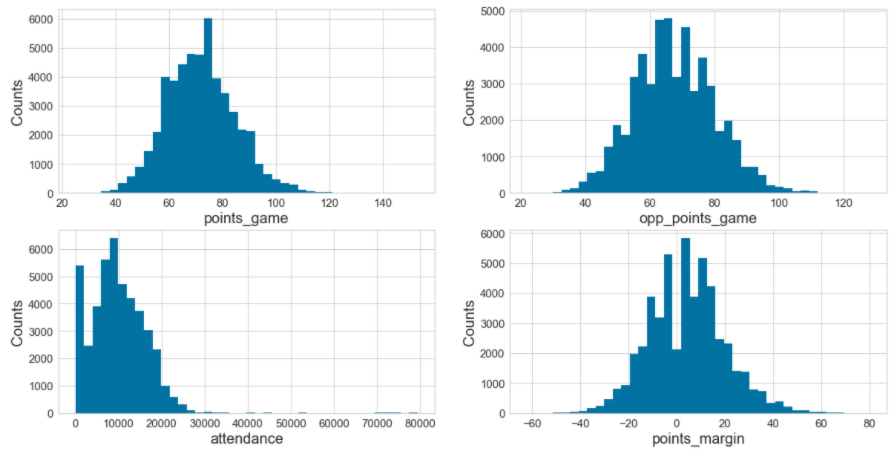
**NCAA Men’s Basketball Game Outcomes Case Study**

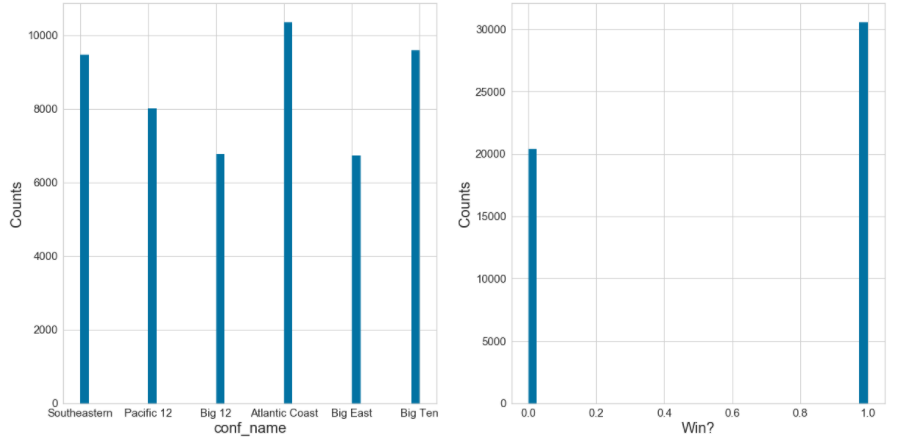
The case study I decided to do is an analysis of NCAA Men's Basketball results to try and find what makes some teams successful and others unsuccessful. Success in this analysis is measured by trying to understand the relationships the variables have with winning any particular game. I've trimmed a dataset of NCAA men's basketball results from 1996-2017 down to include only teams in the Big10, Big12, Big East, ACC, SEC, and PAC12 conferences and their opponents to make the dataset easier to work with as well as make it significant (these are by far the largest basketball conferences at the Division 1 level). My goal is to try and predict whether a game will result in a win or a loss for the team based on the data.

The dataset is 23 columns by ~51K rows and consists of fields such as season (year), date of game, attendance, home team, away team, division, win (Boolean), home conference, away conference, and the margin of victory.

My hope is to have some takeaways and some significant indicators on whether or not a team will win any given game depending on the variables we know.

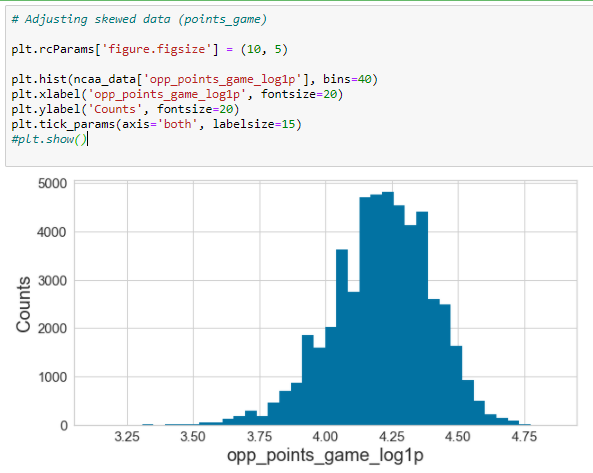
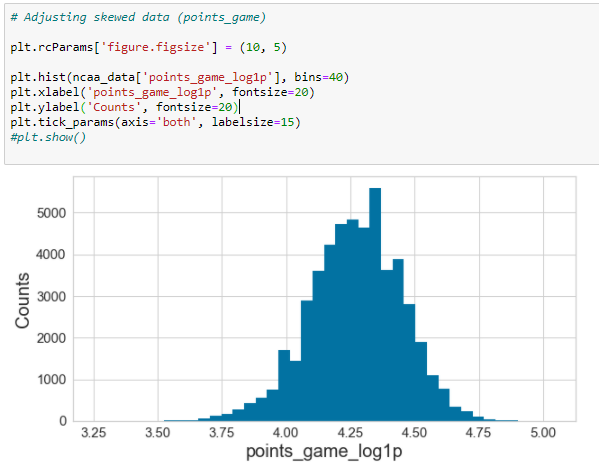
**Summarizing Data and Exploratory Analysis**





**Data Cleanup and Transformation**

I adjusted and updated some of my fields to make them more useful, such replacing zero and null values for Attendance where those are missing information in the data. I updated these values to reflect the average value across all of the values we have in our data. I also updated missing values for division to be the most common division, as most teams in this dataset are Division 1 teams. I also created dummy data for categorical variable conference. Lastly, I created two new fields that are log-transformations of points scored and points allowed. This is how the graphs turned out for those new fields.



Based on how few of columns I have in my dataset and the data already being in relatively good shape from a cleaning perspective, I didn't have a lot of changes to make but I tried to utilize some of the skills and code we learned this week and create additional fields as well as fix a few existing ones to make more sense. Overall, this step of cleaning and adding additional fields was important and useful for the project as a whole.

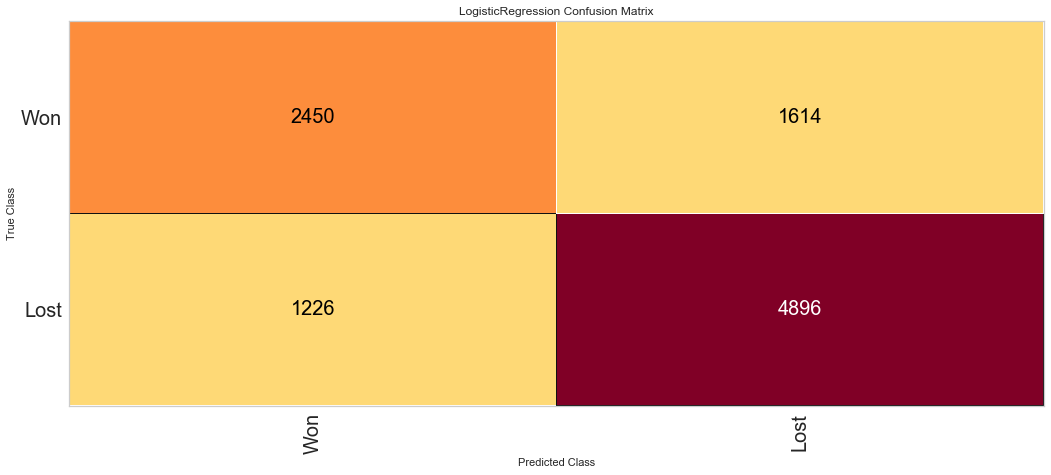
**Modeling and Predicting Outcomes**

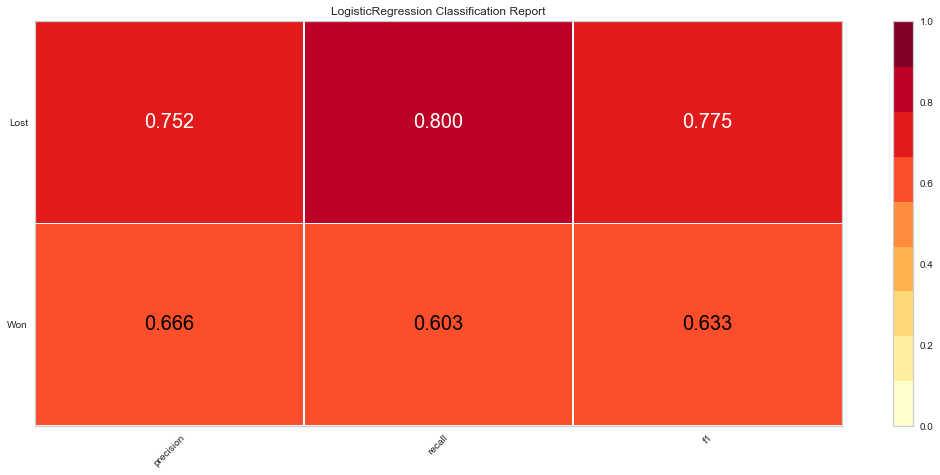
Now that I've begun to build out my models, I'm noticing that the number of columns I have for training the model is limiting my ability to increase the accuracy and ability to predict the correct outcomes (win or loss). One thing I noticed is that my idea of replacing missing attendances with the mean is actually causing that field/variable to have so many repeating means that it’s not actually very useful from a prediction standpoint.

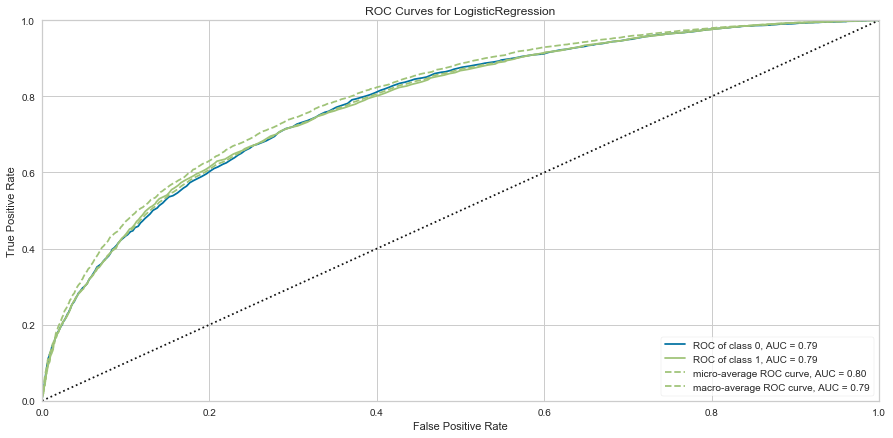
I've tried doing some different inputs, but the ones that looked the best or I got the best results from were using the number of points scored for the team as well as their conference, and how many points they allow and the team's conference. Here are the results/evaluation metrics for those two different models:

**Logic Regression**

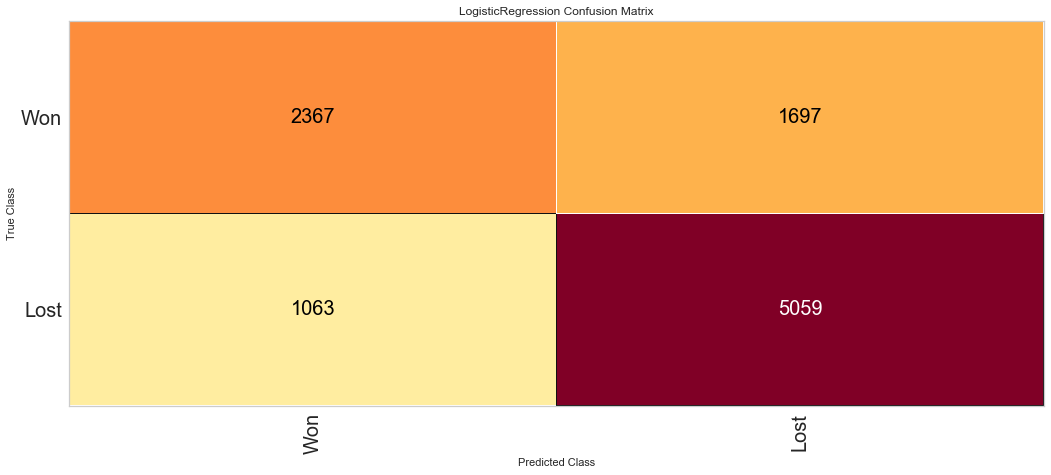
**Points Scored + Conference:**

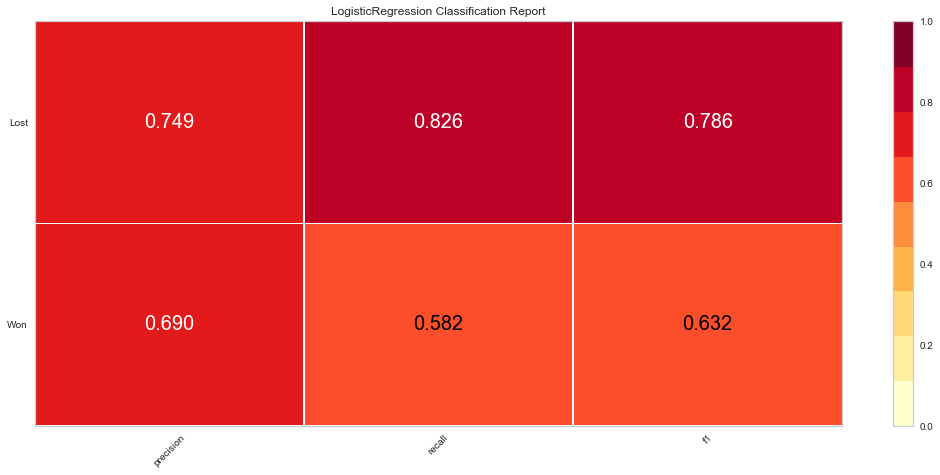


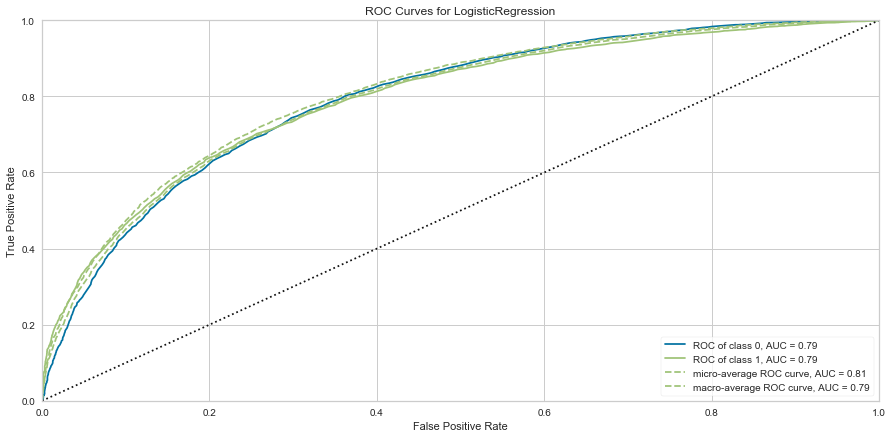




**Points Allowed + Conference:**





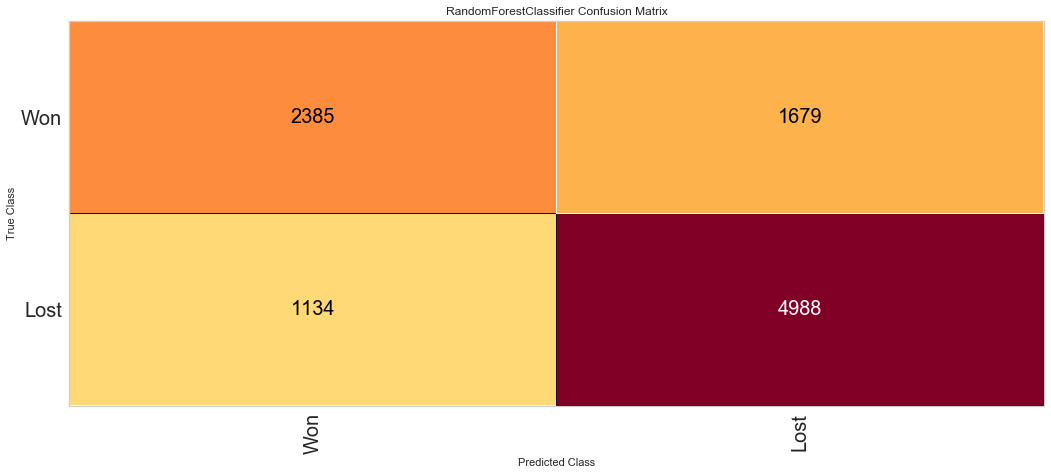


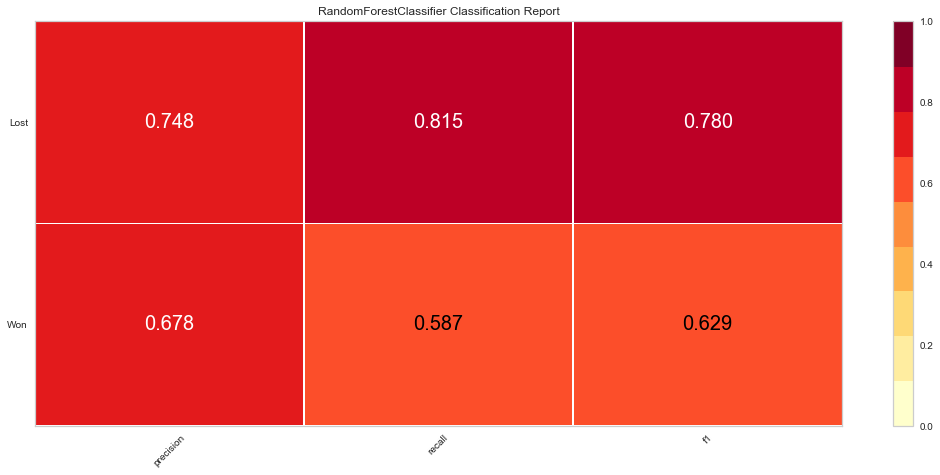
Between the two models, I think the points allow + conference is slightly more accurate when predicting wins and losses, but does have a tendency to have a false negative (predicting a loss when the actual result was a win) than the other model. Neither are perfect and I've kind of touched on the reasons for that in the beginning of this section. Hindsight is 20/20 and I think I learned a valuable lesson to start with more fields and data than you think you will need so you don't have to circle back later when you realize that you probably don't have enough after running your model.

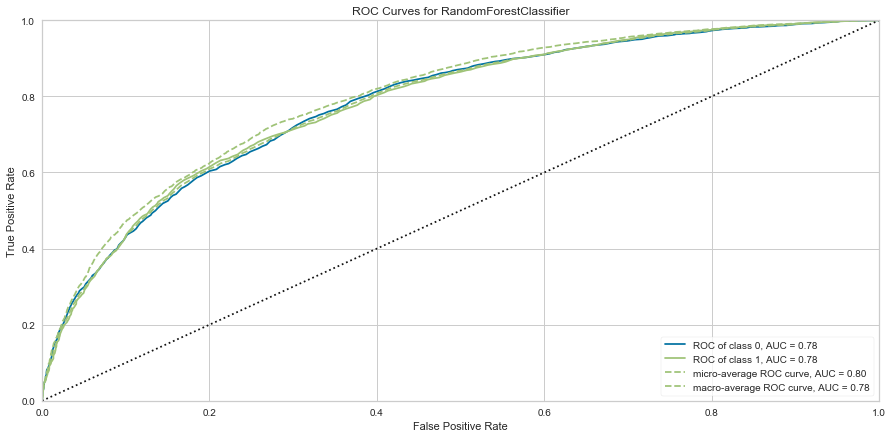
**Random Forest Regression**

After running logistic regressions, I wanted to test a different model and try to improve my accuracy and fit of the model to predict winning/losing outcomes. Because of this, I tried to implement a Random Forest Classifier model to my data (a non-lineral model) to see if my results would improve at all. The results of this model on predictions using Points Allowed and Conference as my features were as follows:

**Points Allowed + Conference:**







In the end, the RFC has improved the accuracy of the games lost but is also predicting more losses when a team actually won. Overall, the ROC and AUC decreased slightly, so it appears that RFC is about as good if not a little bit worse at predicting the outcomes of games using the same parameter/feature inputs.

**Conclusions and Summary**

The overall outcomes of this case study are that with a few significant variables such as points scored or points allowed can be highly influential indicators if a team will win or lose a game, but they are only part of the story. If a team allows a lot of points, typically they are likely to lose, but it’s not always the case if they still outscore everyone they play. There were many lessons I learned along the way while working on this case study project. One of which is, while it is important to be narrow in your object, it is crucial not to be too narrow in your data selection/feature selection. I eliminated features that I did not think were important and ended up with too few to test and try to model results against. This was a painful yet important lesson for me and I think it will be something that influences how I plan and go about similar projects in the future.