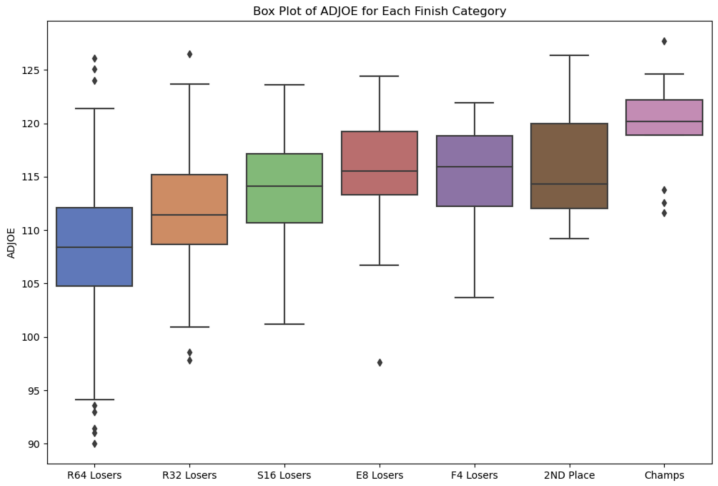
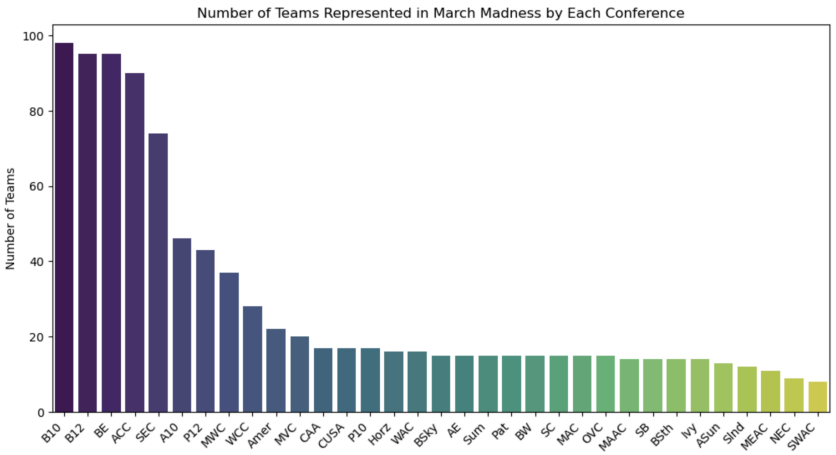
Overview and Initial EDA: This project aims to understand what type of teams, based on their regular season performance, had the best chance of advancing past the first round of the NCAA Men's College Basketball March Madness tournament. More specifically, we wanted to create a model that predicts first round success more accurately than a team's seed, a ranking given by an NCAA committee that uses a team's "resume." However, the statistics that translate to a team getting a higher seed in the tournament do not necessarily correlate with winning tournament games. Our question began by trying to predict the champions; however, we quickly realized that as the tournament progressed, we had fewer data points to find correlations for winning and had insufficient data to build an accurate champion-predicting model. As we see in the graph, 4 out of the 15 champions are outliers in adjusted offensive efficiency. This demonstrates that the data for Adjusted Offensive Efficiency (ADJOE), an estimate of the offensive efficiency (points scored per 100 possessions) a team would have against the average Division I defense, would not be predictive enough to determine the March Madness champion. This same issue was relevant for most statistics in the dataset. For this reason, we decided to try to predict whether or not a given team would win their round of 64 game, for which we had a plethora of data.



Data Cleaning: Originally, we found a clean, extensive college basketball dataset on Kaggle that scraped data from barttorvik.com, a reputable college basketball statistics website. The dataset includes season long statistics for each NCAA Division 1 Men's basketball team from the past 10 seasons (excluding 2020, since there was no tournament that season). However, as we began to explore it further, we realized that the dataset included data from games in the March Madness tournament, so we could not use this dataset because the independent variables of our model would have already been influenced by our dependent variables. If we used this dataset, our model would likely fit the training data very well but struggle to predict new data.

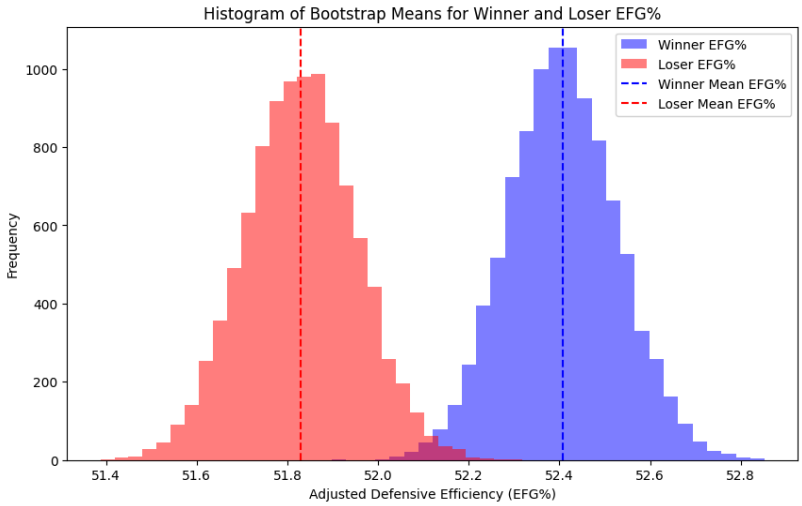
To fix this problem, we went to the source of the Kaggle dataset, barttorvik.com, which allows users to filter the data by time period. Using this feature, we were able exclude tournament games from our data, and expanded our sample size by five seasons as well, since the website has data for every season since 2008. However, based on the way the data was formatted on the website, strange characters appeared in some of the columns in the process of transporting our data from the website to a csv file. Every team also had two rows of data instead of one, with the second row containing their seed in the tournament and what round they lost in — very important information for our purposes — so we needed to extract both pieces of information from the second row. We then filtered out our dataframe so that it only included teams that qualified for the tournament, and added a column whose value was zero if a team lost in the Round of 64 and one if they won. This would become our dependent variable, what we were trying to predict. Once we had coded to add that information to the row above it, deleted the second row and removed all strange characters from the data, we had a clean dataset.

EDA and Analysis: Additionally, we hypothesized that it could be advantageous to use a given team's conference to help predict how well that team would perform. However, conferences realign on a year-by-year basis, we once more ran into sample size issues: Conferences change on a year by year basis, and many of the conferences and many conferences have only been represented by a few teams over the past 15 years. Some conferences have so few teams in the tournament that we do not have enough data from them to use conferences as a metric for predicting success. As demonstrated in the graph, 5-7 conferences represent the majority of the teams in the tournament while many of the smaller conferences average just above one team in March Madness per year. For this reason, it is challenging to use



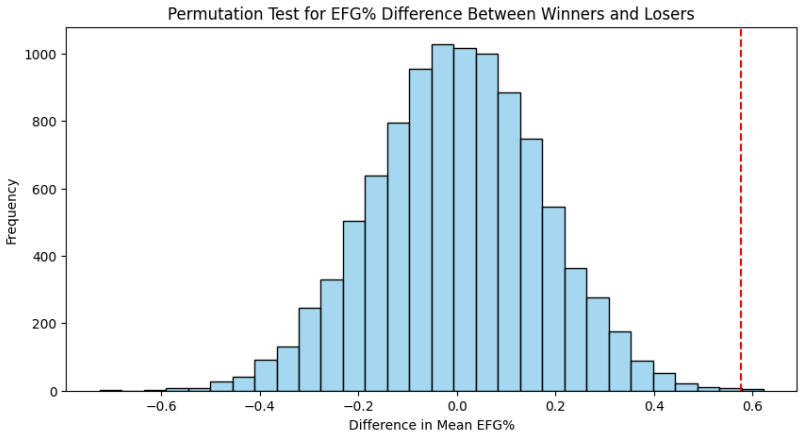
conferences to predict team success because many of the teams in the tournament come from conferences that do not have much data surrounding them, while other teams have an abundance of data surrounding them. As seen in the graph, conferences such as the SWAC and the NEC have almost no representation in the tournament, and there are many conferences that are similar to them. On the other hand, conferences such as the ACC and SEC have much more representation in the tournament which leads to more data. As a result of these issues, we decided not to use conferences in our model.

We decided to use two different types of models to predict the teams that win their Round of 64 game: logistic regression models and LASSO models. To start, we decided to use the 2023 season as our testing data, setting every other year (still excluding 2020) to be our training data. We wanted to observe how our models would perform in a one-off tournament using data from all of the previous years, since that is what we'll use our models for when we predict future tournaments.Logistic regression models worked well for our purposes, as they are good for binary classification. They take in multiple explanatory variables (season-long team statistics, in our case) and outputs a predicted probability of the classification. To contextualize our models, the output was the probability that each row (team) won a *generic* Round of 64 game. One of the limitations of our data was that it did not allow us to make predictions based on a team's matchup; we are only making predictions based on the team's stats and do not consider their opponents, although a team's seed gives an estimate to how good their opposition will be in the Round of 64. We made logistic regression models for two different combinations of features. Initially, we hypothesized that all of the features that are adjusted for strength of their opponents, might form the most predictive model because some teams have a very competitive schedule while some teams in smaller conferences do not face very difficult opponents. The adjusted metrics counteract that difference such that they, in theory, negate the effect of conferences. After we found the first logistic regression model, we decided to bootstrap every statistic within our dataset to create confidence intervals to better illustrate the differences between winners and losers. We then proceeded to do a permutation test on every statistic to understand which statistics had significant differences.



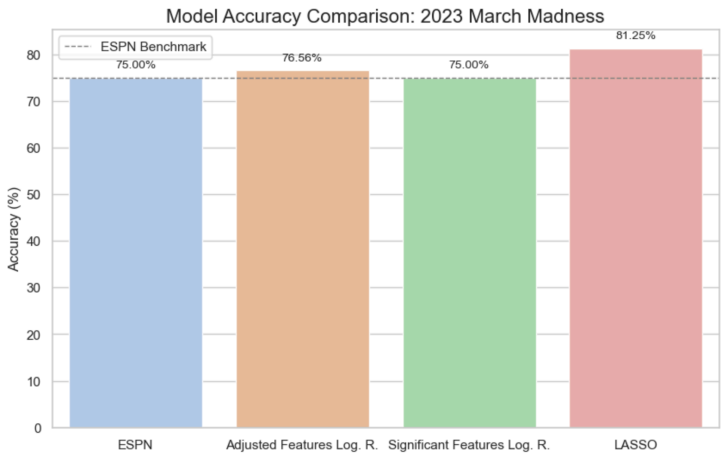
To give an example, when looking at EFG%, a statistic that adjusts regular field goal percentage to account for the extra value of 3-point shots, calculated as (FGM+0.5×3PM)/FGA, where FGM is field goals made, 3PM is 3-point field goals made, and FGA is field goal attempts, the losers in the dataset had a 95% confidence interval of (51.59, 52.07) while the winners had a 95% confidence interval of (52.18, 52.65). The graph to the right illustrates the distribution of means for EFG% for the winners and losers used to calculate the confidence intervals. The one-tailed null hypothesis (H0) for this permutation test was that there is no difference in the EFG% between winners and losers in the dataset, or more specifically, that the EFG% for winners is less than or equal to the EFG% for losers. The alternative hypothesis (Ha) was that the EFG% for winners is significantly higher than that for losers.

The graph is of said permutation test, illustrating the distribution of random mean differences in EFG% in the sample compared to the difference in EFG% between the winners and losers of the first round. Given the observed sample difference in mean EFG% of 0.58 and a p-value of 0.0005, we confidently rejected the null hypothesis. This indicated that there is a statistically significant difference in EFG% favoring the winners, with a very low probability (less than 0.05%) that such a difference could occur by random chance under the null hypothesis. Using the statistics that we found to be statistically significant, we made another logistic regression model.



One thing that logistic regression models do not adjust for, though, is multicollinearity. That is where the LASSO model comes into play: given access to all of the (standardized) numerical features, the LASSO model uses its penalty term, $\alpha \sum_i |\beta_i|$, to remove less relevant features from the model by setting their coefficients to zero. Due to the features that our LASSO model did *not* set to zero are **Adjusted Offensive Efficiency**, **Adjusted Defensive Efficiency** (an estimate of the defensive efficiency (points allowed per 100 possessions) a team would have against the average Division I offense), **Turnover Rate**, **Steal Rate**, **Offensive Rebound Rate**, **Wins Above Bubble** (The bubble refers to the cut off between making the NCAA March Madness Tournament and not making it based off of each team's schedule), and **Seed**. What is interesting about these features is that they were all measured to be statistically significant by our earlier analysis, but not all significant features were used. This can be explained by the way the LASSO model accounts for multicollinearity: it can identify the correlated features and drive the coefficients of the less impactful of the correlated features down to zero. Also notable is the fact that four out of the five available adjusted metrics the LASSO model identified as most important, supporting our hypothesis that the adjusted metrics are useful for predictions due to their adjustment to strength of schedule.

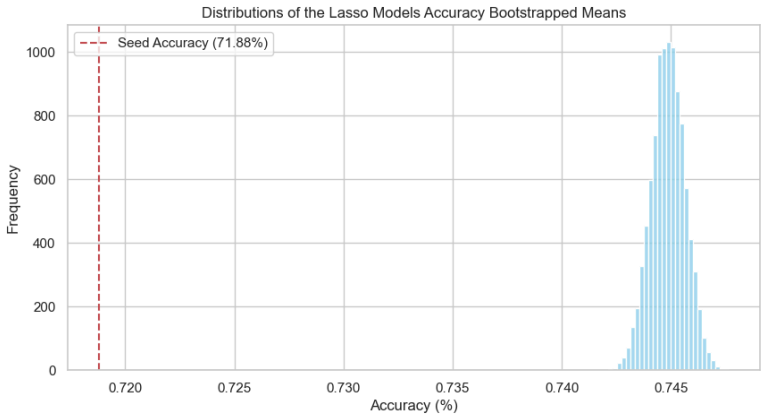
All three of our models performed well on the 2023 testing data, as all of them performed at least as well as ESPN's predictive model, which gives a percent chance each team has to win their Round of 64 game. One notable difference between the two is that their model accounts for matchups, while ours does not. Assuming that any team with over a 50% chance of winning was projected to win by ESPN, they correctly categorized 48 teams out of 64 (accuracy of 75%). Our significant features logistic regression model also got 48 correct, while our adjusted features logistic regression model got 49 correct. Our LASSO model correctly predicted the result of 52 out of the 64 teams.



Conclusion: The main limitation of our model is the obvious fact that match-ups are not taken into account. At the most basic level, this causes models like our LASSO model to, in the case of the 2023 testing data, over predict the amount of teams that can win. In the 2023 testing data, of the 12 teams the LASSO model got wrong, 10 of them matched up against one another and all of which were upsets based on Seed. Match-up data could potentially explain why these upsets occurred and help improve the quality of our model in future work. It's also interesting to note, while our model only is built to predict first round success, that of the 5 teams that performed the upset in these matchups, 4 of the teams

lost in the next round and Princeton, who won their round of 32 game, lost in the sweet sixteen (the next round). The other two teams the LASSO model got wrong matched up against other teams that the model also predicted would win. This led to the model predicting 53.13% of teams to win in the first round (which is impossible for obvious reasons). It's interesting to think how match-up data could have helped us differentiate which team would win these match-ups. Another limitation within the dataset is the inability to consistently control or adjust for conferences in the stats we used in the dataset. Since in college basketball, most team's games are played within conference play, weaker conferences can lead to teams putting up inflated statistics against weaker competitions. Strength of schedule is heavily considered when the NCAA committee chooses Seed. While adjusted stats attempt to account for this, the logistic regression model using significant features and the LASSO model both had stats in them that weren't adjusted. In future work, it might be interesting to have a dataset fully composed of adjusted metrics and see which are significant and which the LASSO model chooses as well as if each of them perform better. One interesting thing we learned from the data was that some statistics that are often used by analysts to predict upsets, like three point shooting, were determined both by our hypothesis tests and LASSO models to not be relevant in terms of predicting tournament wins.

A limitation of using just the 2023 tournament as testing data is its relatively small sample size. While using as many past tournaments as possible is important for accurately training our models, it can be difficult to confidently validate our models using only 64 teams. So, we used a second set of training/testing data to fit and validate our models. We randomly selected 192 of the 960 available teams to be our testing data, and the other 768 teams were used to train the model. This gave an 80:20 training:testing data split, such that we had sufficient data to train our model but also enough to validate it.



To get a better sense of the accuracy of our model, we decided to run 1000 simulations of randomly sampling 80% of our data to train the model and the other 20% to test it. For these models, we made the assumption that time has no effect on our models. For each model, this gave us an average accuracy across the 192 teams in the testing data. For reference, the higher seed won in 345 out of 480 possible games, meaning that 71.88%, categorizing solely based on seed would have resulted in an accurate categorization. Using this as a baseline, our models once again performed well: across 1000 simulations, our logistic regression model using statistically significant features had an average accuracy of 74.86%, our logistic regression model using adjusted features had an average accuracy of 74.78%, and our LASSO model had an average accuracy of 74.49%. Using these 1000 samples, we performed a one tailed sample t-test for each model to understand if there was a statistically significant difference between the accuracy of each model compared to using Seed as predictor. The graph shows the bootstrapped means from the LASSO sample compared to the accuracy based on Seed. In all three cases, we rejected the null hypothesis that the models' accuracies were equal to or less than the accuracy based on seed. We found p-values of 1.63×10^{-182}, 5.37×10^{-181}, 4.08×10^{-166} for the logistic regression model using statistically significant features, the logistic regression model using adjusted features, and the LASSO model, respectively. This results brings confidence to the fact that while the 2023 tournament sample was small, all our models are statistically more accurate predictors of first round performance than seed; however, something to note is that the LASSO model's accuracy may have been overestimated using 2023 as the testing data. When looking at the 1000 samples from the model, only 0.6% of the sample was greater than or equal to the 81.25% accuracy we found when testing our model on the 2023 tournament. It would be interesting, in future work, to investigate which years the LASSO model is very strong and very weak to find ways to better improve the model.

Group Contribution: I'd like to note that before writing the group contributions, it's pretty hard to clearly define each individual's contributions just because of the nature of how we worked through the project. A lot of the work was done together at the A level of the Reg or on facetimes together. Mike focused on the exploratory analysis that led to us steering away from trying to predict the champion of march madness and us not using conferences. Ethan helped with that initial exploration, helping steer us towards the question we ended at which was trying to understand the first round of the tournament. Ethan also did the hypothesis testing done to help create the logistic regression model using significant features and the exploration done in the conclusion. Liam focused on the initial data cleaning for the dataset and working on the actual models themselves, including a lot of the work not only in the analysis but the conclusion as well. Oh ya, and Ethan (that's me writing this) came up with the name "Understanding the Madness of March."