Background:

Since 2018, I have competed in Kaggle's March Madness competition and developed my codebase for the project to new highs after each year. The main goal is to build a machine learning model that predicts the probability of any one team beating another in the college basketball tournament. Once I have predictions for every possible matchup, I run 1000 Monte Carlo simulations on the tournament bracket that estimates the likelihood of every team advancing to the next round. You can use this output to fill out a bracket strictly based off the model predictions which is similar to the output from Nate Silver's FiveThirtyEight predictions.

Code:

main.R:

The main file that runs and creates the output. In this file, I read in all the data and spend lots of time manipulating the data to create a clean data set from which to build models. Some of the data include past game-by-game regular season, conference tournament, and march madness tournament team statistics. This code aggregates game-by-game statistics into season averages per team for each season and feature engineers more advanced basketball metrics. For example, I create offensive rating, defensive rating, net rating, effective field goal percentage, assist/turnover ratio, rebounding percentage, three-point field goal percentage, strength of schedule, distance traveled to games, etc. Each observation is a game between two teams, so I differenced the variables for both teams and used the differenced variables as inputs to the models.

I created the main training data set to perform cross validation for the models to predict the outcome of the next tournament. Furthermore, I created another data set that is similar, but acts as a "repository" of all possible matchups that could theoretically happen per season. This is done to have an observation for any possible matchup for the Monte Carlo simulations. In terms of modeling for the 2021 tournament, I trained four machine learning models [glmnet (lasso/ridge/elastic net), random forest, gradient boosted model, and xgboost] using 3-fold cross validation. The training set consists of all past tournament games from 2003 through 2019 (1,115 games). I compared the average performance of each model on the cross validated hold out set and chose the model that minimized log loss error. Lastly, I used that model and ran 1,000 bracket simulations to come up with a table of each team's chances of advancing to every round. More precisely, I calculated the proportion of times each team advanced to the next round out of 1,000 tournaments and used those proportions as an estimate to the chance or likelihood that a team will advance to each round.

functions.R:

Stores all functions created that are used in the main.R file. This includes functions to run each model, visualize data through scatterplots and correlation matrices, find the specific tournament round for any two teams matched up, and create "repository" data set.

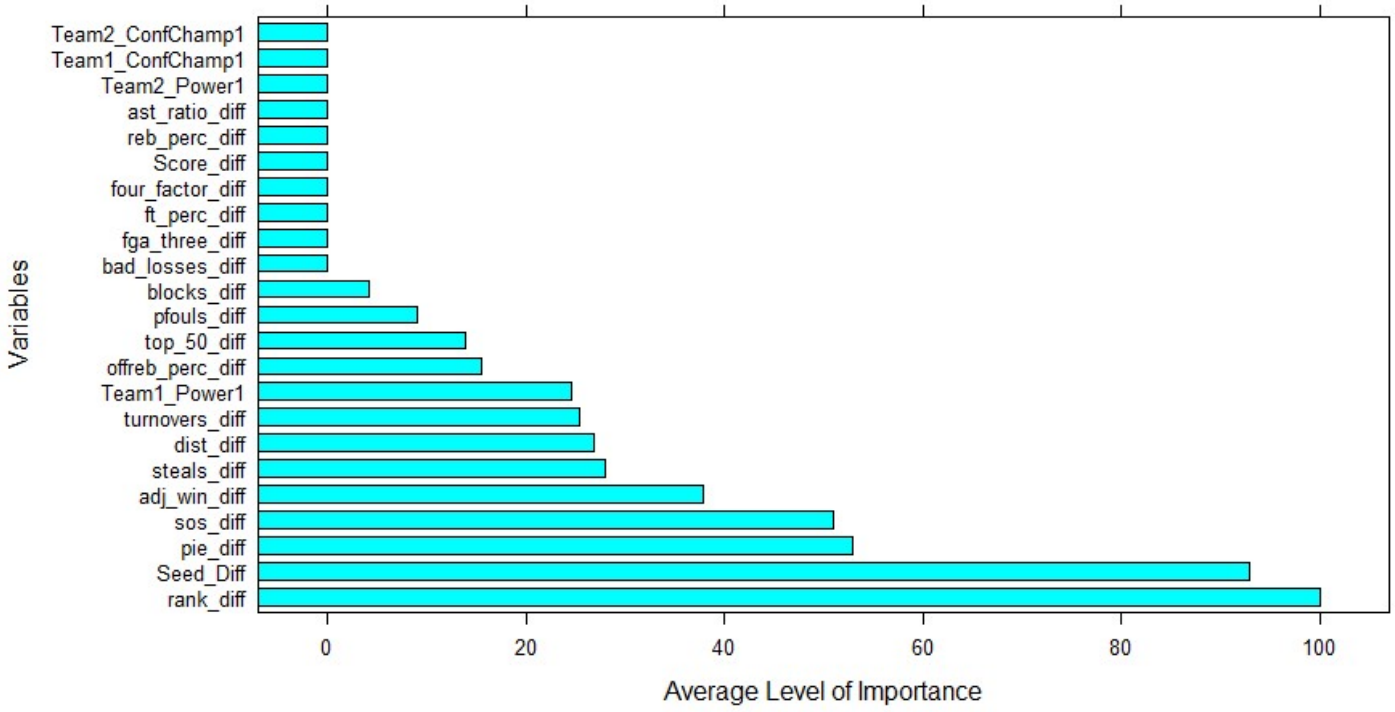
bracket_sim_functions.R:

Large functions to run the Monte Carlo bracket simulations for each of the models.

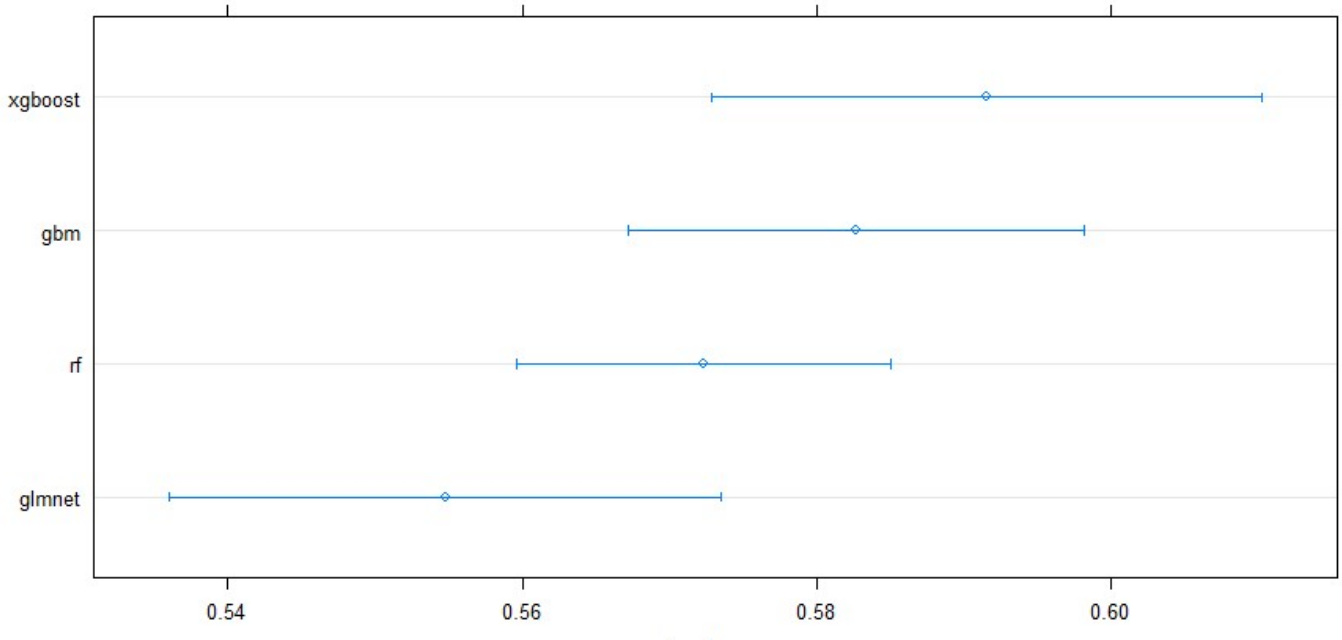
Output:

1) 2019 Monte Carlo Simulation Results table using best model (glmnet). Use link below. The table is formatted in the specific order one would normally see on a bracket.
    a. (htmlpreview.github.io/?https://github.com/rdavtian/March-Madness-Predictions/blob/master/Preds/Preds19/GLMNET/Bracket_Sim_Results.html)
    b. Notice I predicted Virginia as the most likely team to win the tournament, which did happen.
2) Glmnet variable importance chart
    a. Chart can be found on next page.
3) Model Comparison Chart plotting average cross validated test set error with confidence interval band.
    a. Chart can be found on next page.

GLMNet Variable Importance



3-Fold CV Error Caret Model Comparison



Confidence Level: 0.75