Brief overall summary of what was done by the game-by-game team:

1. Designed model: takes in a difference vector as an input, outputs either a win/loss (0/1) or a point spread based on the home team. Used linear regression, logistic regression, neural network, decision tree, and random forest
2. Data querying and parsing:
    a. Queried the NBA API to get all NBA games for all NBA teams, and stored this data in a dictionary that mapped team_id to a games history dataframe
    b. Cleaned the data such that game history dataframes would only contain 82 regular season games (66 for NBA lockout)
3. Created training set
    a. For each season from 2009 to 2018 inclusive, we iterated through each team's corresponding games history dataframe.
    b. For each game, we got the statistical team averages for both teams from the past n games, normalized them, and then subtracted the two. We would also get the corresponding label (0 if the home team lost, 1 if the home team won), or the point differential between the home and away team.
    c. We had roughly 82 games x 9 seasons data points, where the features were the normalized differences between each team stat, and the labels were either a win/loss or the point differential
4. Results (directly predicting win/loss)
    a. 67% for logistic regression and the neural network. 55% and 57% respectively for decision tree and random forest

b. 67% also for predicting win/loss using linear regression point differentials