https://www.kaggle.com/datasets/eoinamoore/historical-nba-data-and-player-box-scores/data

https://calreynolds.medium.com/using-machine-learning-to-predict-nba-team-wins-5b71681e86f5

Project Proposal - NBA Game predictor

a. **Group Members:** Ying Catlin, Brock Bye
b. **Dataset:** NBA Games Dataset - Using TeamStatistics.csv and Games.csv
TeamStatistics.csv
    - 48 columns, a game has two rows, for each team played statistics
    - Gameid, team name, free throw attempts, free throws made, total rebounds, turnovers, home/away??
    - All games, each team played in a season, dating back to the 1947 season. We would use 2-5 previous seasons, prior to the 2024 season

The dataset consists of data from NBA games dating back to the 1947 season. The TeamStatistics.csv, specifically, contains 48 total columns and two rows per game that contain game statistics for each team. Examples of columns include the GameID, team name, free throw attempts, free throws made etc.

c. **Task:** Use previous season's statistics (rebounds, free throws, free throw attempts, and turnovers, home/away) to predict the winner of each game in the 2024 season.

Our goal is to use previous season's statistics to predict the winner of each game in the 2024 season. We will transform the data into the following: one game row, each stat will be denoted by home or away team, each stat will be a rolling 5 game average going into the current game. We will use seasons 2021, 2022, 2023 as our training set and 2024 as the test set. Our transformed data set will contain 9240 rows (77 games x 30 teams x 4 seasons) and between 5-15 columns of game stats.

d. **References to similar examples:**

A couple similar examples are:
Using Machine Machine Learning to Predict NBA Wins - Predicting total wins for a team's season using logistic regression and a neural network, they had good success in that 70% of their results were within 3 wins of each team.
Predictive Analysis leveraging ML - Using previous team and player statistics, use various models (SVM, logistic regression, random forest, and LSTM) to predict NBA games and determine specific feature importance in predicting the winner, all of their models were about 74% predicted accuracy.

e. **Our proposed models:**

Our two proposed models are a binary classifier using logistic regression and binary classifier using a multi-layer neural network. For logistic regression, we will use a train test split, normalize the data, define a dataset class, and define our logistic regression formula using sigmoid loss function and SGD optimizer. Hyperparameters to tune include the learning rate and regularization. For the multi-layer neural network, we will similarly use train test split, normalize, and create a dataset class. Hyperparameters to tune are the number of layers, learning rate, regularization, batch size, and optimizer.

Binary Classifier using logistic regression -

Split train test, Normalize data using standard scalar on training set, define Dataset class, define our logistic regression formula using sigmoid loss function and SGD optimizer.

Hyperparameters: learning rate, regularization

Binary Classifier using Neural Network -

Split train test, define Normalize training data using standard scalar, Dataset class, create dataset from training and testing data, define our neural network, create optimizer and loss function

Hyperparameters to tune: number of layers, learning rate, optimizer, regularization, batch size

f. **Evaluation Metrics:** Accuracy, Precision, Recall, F1, ROC curve

The evaluation metrics relevant to our task include the accuracy, precision, recall, F1, and ROC curve score. These metrics will allow us to see how accurate our models perform and easily compare the output of the two.

g. **Graphs and Visualizations:** Loss rate line chart, ROC curves, diagram of neural network architecture

The graphs and visualizations we plan to include are the loss rate graphs, ROC curves, and the diagram of our neural network architecture. The loss rate graphs will show us if our model generalizes well and can also help tune our learning rate. The ROC curves are a great visualizer for how well our models can distinguish between our two classes for different thresholds.

h. **Meeting schedule, timeline, individual tasks:**

Tasks:

Ying - Start on Logistic regression model

Brock - Start on Neural network model

Timeline:

12/1 - Dataset established, normalized/preprocessed, train test split, define dataset class

12/5 - Evaluation, tuning

12/9 - Charts and analysis

12/10 - Poster, write up

I plan to have the dataset transformed into what was described in the task section by Nov 27. I'll clean the dataset, only include the regular season, and implement rolling 5 game averages. I plan to make the code for the neural network model by Dec 1. I plan to have evaluation and tuning completed by Dec 5, and provide my charts and analysis by Dec 9. We plan to meet twice a week to share progress, review code, tune models, and make decisions collaboratively.

i. **Backup plan:**

If our chosen features appear to not have high correlation with the results, we can replace them or add more features. We could expand the rolling window, use interaction features (assist to turnover ratio, offensive defensive rating), or use game context metrics (rest days) to try to improve the models.