My final project for CSE 4095 involves predicting the next game stats for NBA players using linear regression and AI transformers. I'm superexcited to share this project because I believe it can be used for various applications like sports betting, fantasy sports, and even sports journalism. Anyway, let’s get right into my code.

First, we import necessary libraries such as pandas for data manipulation, scikit-learn for machine learning, and the nba_api for fetching NBA player data.

The 'fetch_player_game_logs' function takes a player_id and season as input and fetches the game logs for the specified player and season using the nba_api library.

Next, we have the 'prepare_data' function that takes a dataframe (df) and a number of lags (num_lags) as input. It converts the REB, AST, and PTS columns to numeric and creates lag features for each of these columns. This will help our model capture the relationship between a player's past performance and their future performance.

Now let's look at the 'predict_next_game_stats' function. It starts by finding the player_id using the player_name provided. If the player is not found, it prints an error message and returns. Otherwise, it fetches the game logs for the specified player and season and prepares the data by creating lag features.

For each of the stats (PTS, REB, AST), we preprocess the data by splitting it into input features (X) and target values (y). We then split these into training and testing sets using the train_test_split function from scikit-learn.

After that, we create a LinearRegression model. Linear regression is a technique used to model the relationship between an input variable (or multiple input variables) and an output variable. It tries to find the best-fitting straight line (or hyperplane in case of multiple input variables) that represents the relationship between input and output. In our case, we're using linear regression to model the relationship between a player's past performance and their next game stats.

We fit the LinearRegression model to the training data, which helps the model learn the relationship between the input features (past game stats) and the output (next game stats). Once our model is trained, we prepare the input for the next game prediction and use the model to predict the stats for the next game. This prediction is based on the learned relationship between past and future performance.

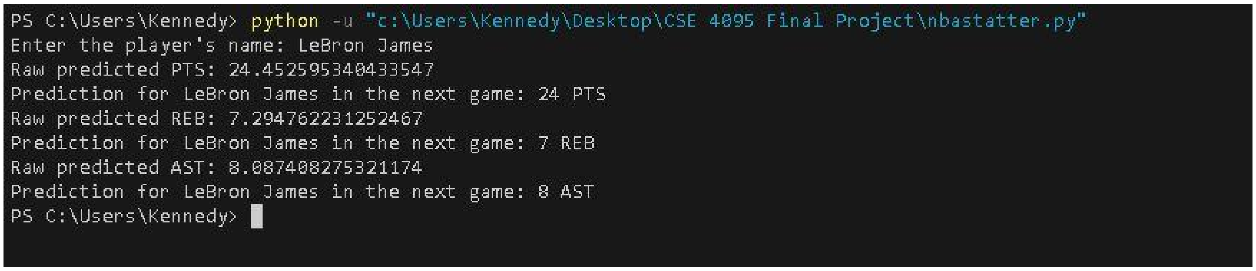
Finally, we round the predicted stats to the nearest whole numbers, as game stats are typically reported as whole numbers. We then print the final predictions for the player's next game stats.

The main function (main) simply takes the player's name as input and calls the 'predict_next_game_stats' function to predict their next game stats. Note that I chose the year

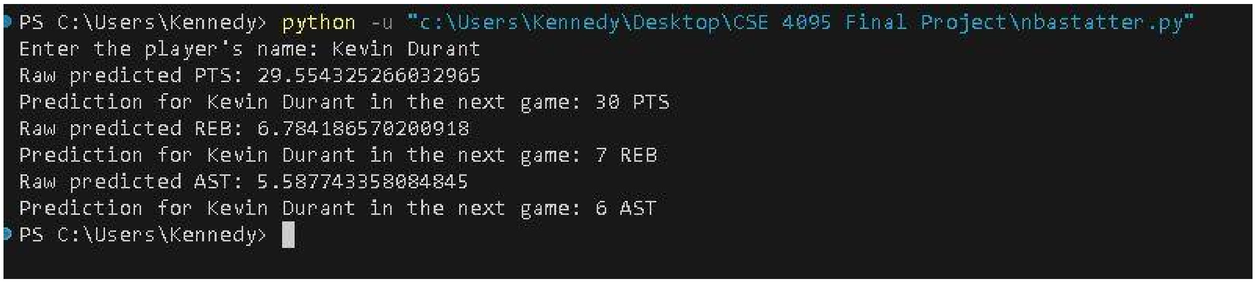
2020 because the program seems to work best from that year as opposed to like 2023 where data might be missing from the stat databases.

This project has the potential to change the world of sports analytics and betting by providing accurate predictions of player performance. With further improvements, it could help fans, bettors, and professionals make better decisions based on data-driven insights.

Screenshots:



```
PS C:\Users\Kennedy> python -u "c:\Users\Kennedy\Desktop\CSE 4095 Final Project\nbastatter.py"
Enter the player's name: LeBron James
Raw predicted PTS: 24.452595340433547
Prediction for LeBron James in the next game: 24 PTS
Raw predicted REB: 7.294762231252467
Prediction for LeBron James in the next game: 7 REB
Raw predicted AST: 8.087408275321174
Prediction for LeBron James in the next game: 8 AST
PS C:\Users\Kennedy>
```



```
PS C:\Users\Kennedy> python -u "c:\Users\Kennedy\Desktop\CSE 4095 Final Project\nbastatter.py"
Enter the player's name: Kevin Durant
Raw predicted PTS: 29.554325266032965
Prediction for Kevin Durant in the next game: 30 PTS
Raw predicted REB: 6.784186570200918
Prediction for Kevin Durant in the next game: 7 REB
Raw predicted AST: 5.587743358084845
Prediction for Kevin Durant in the next game: 6 AST
PS C:\Users\Kennedy>
```