**Guide to running the Shiny FPL Lineup Optimization Tool**



This tool is meant to predict points output gameweek by gameweek using live stats from the FPL API, paired with a regression model of historic player performance data.

Historic gameweek data was pulled from the following Github page: <https://github.com/vaastav/Fantasy-Premier-League/tree/master>

**OLS Regression Model**

OLS regression is a simple linear regression technique meant to evaluate the effect of inputs (independent variables) on outcomes (dependent variables). The script ‘fpl\_fetch\_new.R’ uses a basic OLS model that considers the following independent variables and their impact before performing a prediction:

* Total points for the gameweek, the dependent variable
* Player, classified by name as a categorical independent variable
* Team, classified by name as a categorical independent variable
* Opponent team, classified by name as a categorical independent variable
* Probability of playing by historical player trends, historical team matchups and aggregate during past seasons and the current season
* Probability of starting by historical player trends, historical team matchups and aggregate during past seasons and the current season
* Goals scored per 90 minutes by historical player trends, historical team matchups and aggregate during past seasons and the current season
* Goals conceded per 90 minutes by historical player trends, historical team matchups and aggregate during past seasons and the current season
* Assists per 90 minutes by historical player trends, historical team matchups and aggregate during past seasons and the current season
* Clean sheets per 90 minutes by historical player trends, historical team matchups and aggregate during past seasons and the current season
* Saves per 90 minutes by historical player trends, historical team matchups and aggregate during past seasons and the current season
* Yellow cards per 90 minutes by historical player trends, historical team matchups and aggregate during past seasons and the current season
* Red cards per 90 minutes by historical player trends, historical team matchups and aggregate during past seasons and the current season
* Penalties missed per 90 minutes by historical player trends, historical team matchups and aggregate during past seasons and the current season
* Penalties saved per 90 minutes by historical player trends, historical team matchups and aggregate during past seasons and the current season

For prediction, the model replaces previous season statistics with those for the 2024 season.

The script also pulls live form, injury, suspension, and transfer values and news from the FPL API to update player availability and predicted points.

* Form is added to player points through the following formula:
* If the FPL API indicates there is a 75% chance a player will play, predicted points are given a weight of 0.85.
* If the FPL API indicates there is a 50% chance a player will play, predicted points are given a weight of 0.5.
* If the FPL API indicates there is a 25% chance a player will play, predicted points are reduced to 0.
* All other injury statuses and suspensions are given a predicted points value of 0.
* Players who have been transferred or loaned are removed from the dataset.

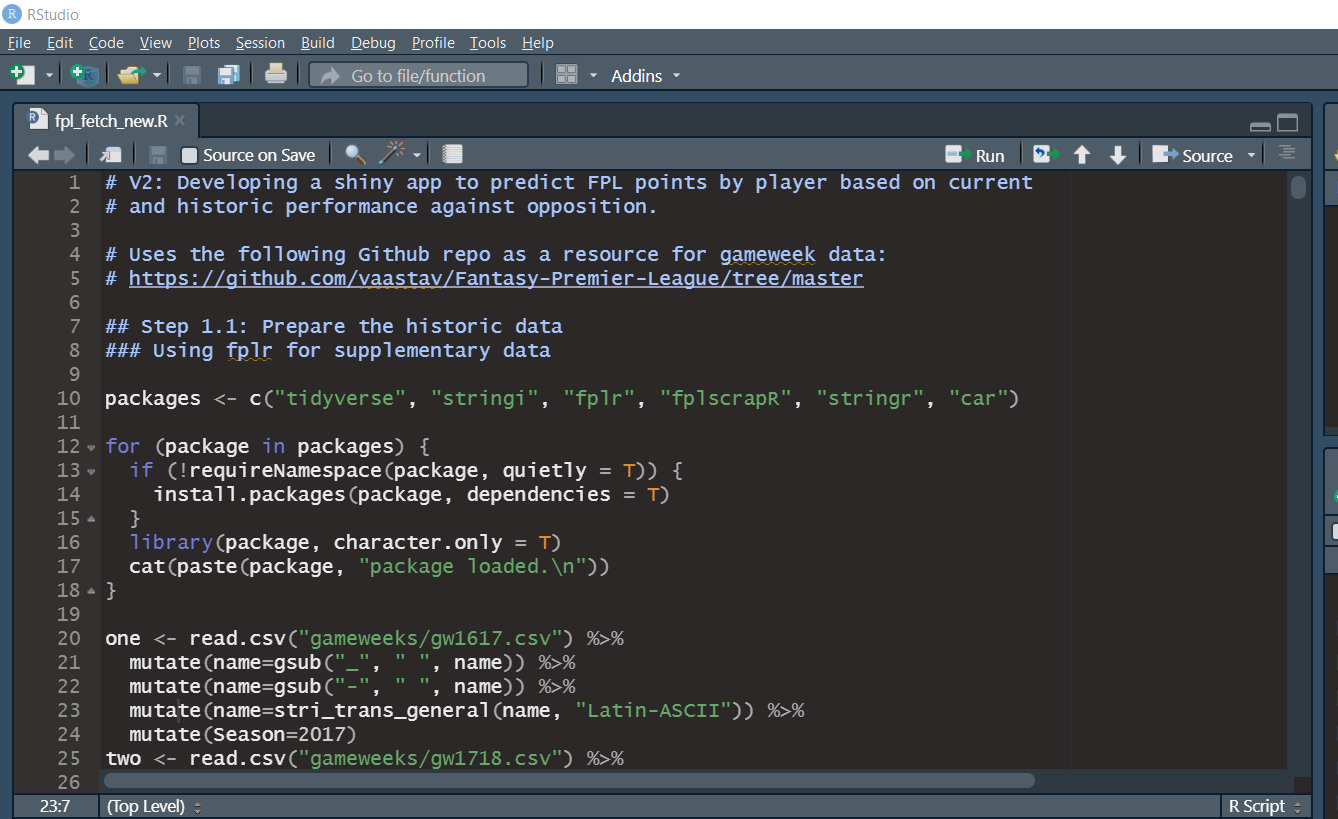
**Required Softwares**

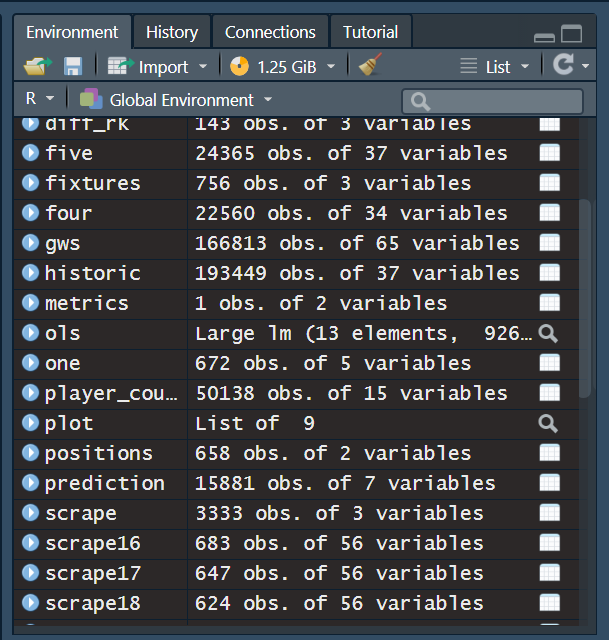
* Install R version 4.1.2: <https://cran.r-project.org/bin/windows/base/old/4.1.2/>
  + **NOTE: SOME OF THE PACKAGES REQUIRED WILL NOT RUN FOR THE MOST RECENT VERSIONS OF R.** It is advised to use version 4.1.2 until the packages are updated.
* Download and install R Studio: <https://posit.co/download/rstudio-desktop/>

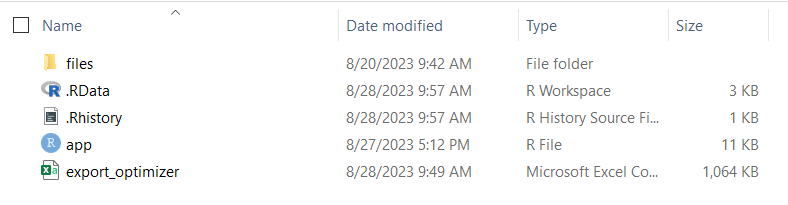
**Run the Model**

To utilize the Shiny tool, the user first needs to run the OLS model. To run the model, follow these steps:

1. Download the directory from either the Google Drive link (<https://drive.google.com/drive/folders/1MczPsv4Kffw2A_HayIDiYyFrxeHz97Cv?usp=sharing>) or clone the directory from Github (<https://github.com/ncfisher411/Shiny-lineup-optimizer/tree/main>). **Altering the directory structure is not recommended unless you are editing the tool.**
2. Open the R script “fpl\_fetch\_new.R”. Click “Ctrl+A” and “Ctrl+Enter” to run the script. This should update the file “export\_optimizer.csv”. This script must be successfully run before the user can run the shiny tool. The full run will take approximately 10-15 minutes. **NOTE:** Running the script may take longer on your first use, as R will potentially need to install several packages used in this script.

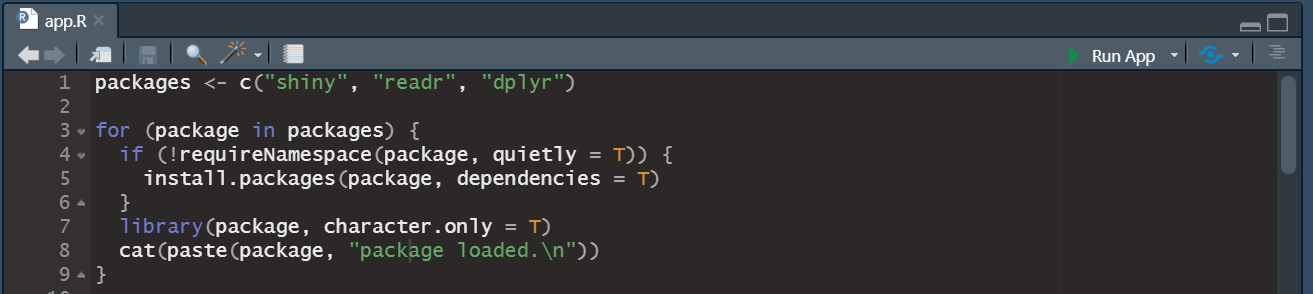


To ensure the script has run successfully, check the R environment for the object “prediction,” and check that the file “export\_optimizer.csv” has updated by checking the date and time modified. The CSV file can be found in the folder “FPL\_Lineup\_Optimizer.”

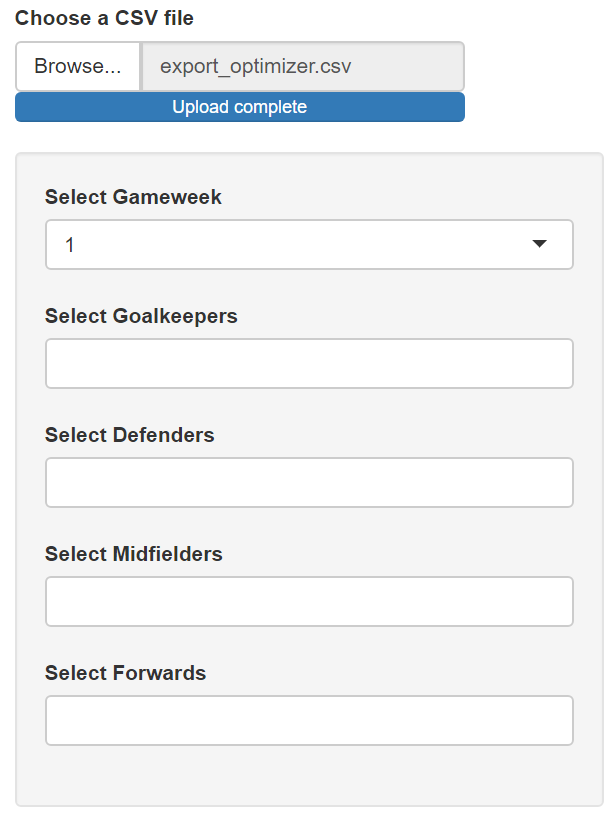


Once the script has finished running, close the script in the top left corner of the console, and then close RStudio.

1. Open the folder “FPL\_Lineup\_Optimizer” and open the script “app.R” using RStudio. This will open the script for the shiny app in R Studio. Click “Run App” in the top right corner of the console and the app will open in a new window. Maximize the window to have a better view of the app.



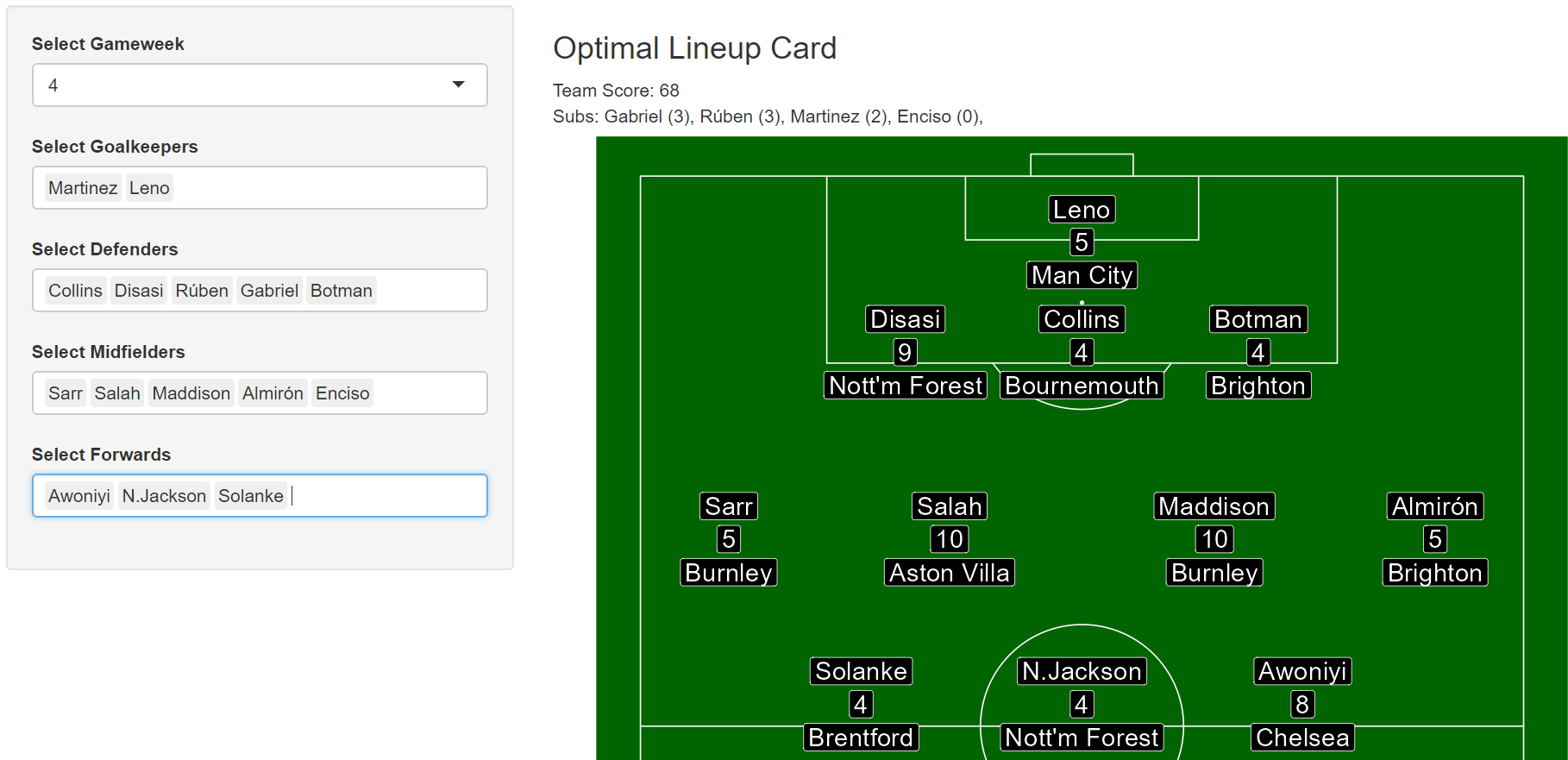
1. Under the header “Choose a CSV file,” click Browse and select the results from the model script, “export\_optimizer.csv.” These should be one folder higher than the “app.R” in the directory. This will prompt new drop down menus to load:



* “Select Gameweek” allows the user to filter the results by gameweek. A gameweek must be selected in this version of the app.
* “Select Goalkeepers” allows the user to select up to 2 goalkeepers from a dropdown menu. The user can also type the names to filter through the options.
* “Select Defenders” allows the user to select up to 5 defenders from a dropdown menu. The user can also type the names to filter through the options.
* “Select Midfielders” allows the user to select up to 5 midfielders from a dropdown menu. The user can also type the names to filter through the options.
* “Select Forwards” allows the user to select up to 3 forwards from a dropdown menu. The user can also type the names to filter through the options.

1. Selecting the maximum number of player names will show the full selection of players in a lineup card plot, filtered from the model results, for indicated gameweek. This plot shows the following information:

* In the main body of the plot: The player name, position, predicted points for the gameweek, and their opponent.
* In the header above the plot, the list of subs organized by predicted points



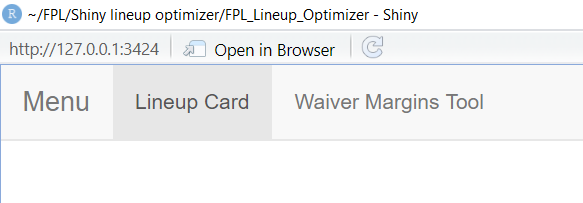
**Waiver Margins Tool**

This app also contains a tool to directly compare players predicted outputs. The tool is located on the second menu of the navigation bar, and uses the uploaded CSV file as well. This tool is intended to compare players for adding and dropping on waivers or free agency by comparing z-scores both overall and by position where:

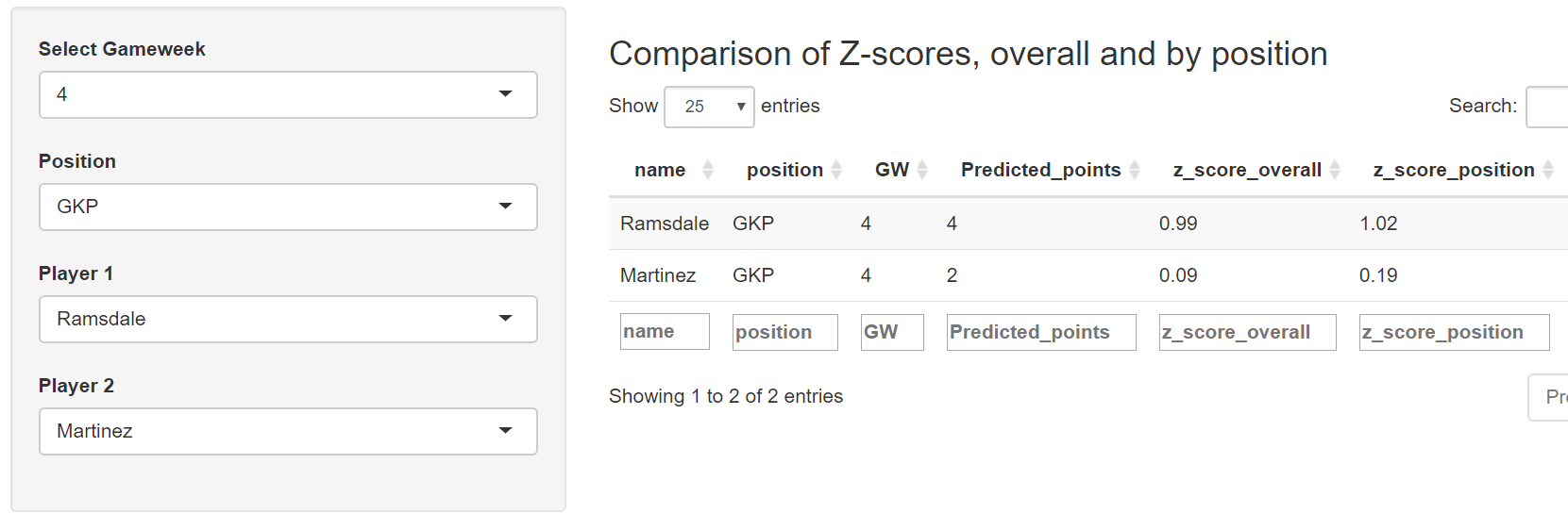
* A negative z-score indicates below average predicted points
* A positive z-score indicates average predicted points
* A zero z-score indicates exactly average predicted points

To use this tool:

1. Upload the CSV file and navigate to the second window on the navigation bar, titled “Waiver Margins Tool.”



1. Select the gameweek, position, and two players to compare. This will load a data table of the two players with the desired statistics.



**Caveats for user consideration**

This model will predict very high or low points for players who are new to the league with no prior Premier League experience between the 2016/17 season and the current date due to a lack of data. The current solution is replacing player data with an average of current season statistics and team statistics. This may be fixed once the new player’s team has played against each team at least once. This may need to be rectified in later model versions, as new players such as Moussa Diaby who are extremely valuable in FPL should be more accurately modeled going forward. Consideration for transferred players could include adjustments to the per 90 statistics based on their experience in previous teams or leagues.This would require marrying FPL data with those from other open data sources such as [FBref](https://fbref.com/en/) or [Transfermarkt](https://www.transfermarkt.us/).

**Future Shiny app additions**

Future options include expanding the lineup optimizer to apply to public leagues as well, which will compare all players regardless of selection. This will need to include functionality such as optimization for captains and bonuses.