**FINAL PROJECT REPORT**

Group Members: Shashank Gampa, Sai Srinath Putta

**PROBLEM STATEMENT**

Scraping the data from a website and creating a database. Using a regression model on our dataset for a relevant prediction.

**WEB SCRAPING**

Web scraping is the process of collecting and parsing raw data from the web or the web page.

It will acquire text-based data from page sources, store it into a file and sort the output according to the set parameters.

We have scraped the data using Python. Notably, there are several types of Python web scraping libraries from which we can scrap the data.

- Requests

- Beautiful Soup

- Pandas

- Selenium

**WEBSITE**

The website we used to scrap our data was FBref.com, which is a football statistics website that contains information about all the football leagues and competitions.

Link: <https://fbref.com/en/comps/9/2021-2022/2021-2022-Premier-League-Stats>

**PROCEDURE**

SCRAPING THE DATA :

* We used Jupyter Notebook (Python) to scrap the data we need.
* We used libraries Pandas, Request, and Beautiful Soup libraries to extract the contents of our dataset.
* We first used pandas to scrap the datasets that are small and direct such as players shooting data, teams data, squad stats, etc.
* We then used Request and Beautiful Soup to extract our biggest dataset which consisted of the data of every match including every statistic from every single match.
* We used this ultimate dataset in our bonus regression model.
* These are the code snippets for our scraping:

Graphical user interface, text, application

Description automatically generated Graphical user interface, text, application, email

Description automatically generated

CREATING A DATABASE

* We first exported the dataframes into a CSV file.
* We then manually cleaned the data and made sure that the CSV files are in the right format so they can be converted into a table in the database.
* We used MySQL workbench to create our database.
* We imported the formatted CSV files into the schema we created in our database.
* The overview of our database is as below:

Graphical user interface, text, application

Description automatically generated

* Above towards the left, we can see the scheme we created and the tables it contains.
* We can also see the overview of the data in our database.

**QUERY EXECUTION**

* Finding out the top 10 matches with the highest possession by a team in a match.
* Query: **select \* from league\_matches where poss > 70 order by poss desc;**

Graphical user interface, text, application, email

Description automatically generated

* Finding out the top goal scorers of the season.
* Query: **select \* from player\_goals order by goals desc;**

Graphical user interface, text, application, email

Description automatically generated

* Finding out the highest goal scorers from the winning team.
* Query: **select \* from player\_goals where squad = 'Manchester City’ order by goals desc;**

Graphical user interface, text

Description automatically generated with medium confidence

* Finding out the highest XG in a game.
* Query: **select comp, round, team, gf, ga, opponents, xg, xga, poss, sh, sot from league\_matches order by XG desc limit 10;**

Graphical user interface, text, application

Description automatically generated

* Finding out the highest assister of the season.
* Query: **select \* from player\_assists order by ast desc;**

Graphical user interface, text, application, email

Description automatically generated

* We cross-checked our query data with the actual data and found there was no data loss.
* 100% of data from scraping was imported into the database.

**BONUS**

* We used the league\_matches dataset to predict the match winners.
* The Regression model we used for our data was the ensemble RandomForestClassifier.
* Using the sklearn ensemble, RandomForestClassifier makes the prediction using regression. Ultimately, Random Forest uses Linear Regression here.
* We used the Result attribute as the response variable here to predict the winners.
* Using the statistics present in the same dataset, we calculated the probability of each match winner and predicted the outcome.
* “Football” is a sport where the outcomes are solely not based on stats but also on luck.
* Despite this we got an accuracy score of 52, which in regard to the dataset present and being sports result prediction is a good one.
* Overall, we got more than half our predictions right.
* Accuracy :

Background pattern

Description automatically generated with medium confidence

* Code snippet of our Regression Model:

A screenshot of a computer

Description automatically generated