Our project topic is football squad analysis.in our project we are using multiple random forest. Here we use many modules in our project such as:

**import pandas as pd [**using for read csv data file]

**import numpy as np [**using for number python]

**import matplotlib.pyplot as plt [**using for graphical representation]

**from sklearn.preprocessing import StandardScaler [**using for standardize features by removing the mean and scaling to unit variance]

[scikit-learn:

Simple and efficient tools for predictive data analysis · Accessible to everybody, and reusable in various contexts ·]

**from sklearn.linear\_model import LinearRegression, LogisticRegression**

[Linear Regression is a machine learning algorithm based on supervised learning. ... Regression models a target prediction value based on independent variables. It is mostly used for finding out the relationship between variables and forecasting.]

[Logistic regression is a statistical analysis method used to predict a data value based on prior observations of a data set.]

**from sklearn.base import BaseEstimator [**BaseEstimator[source] Base class for all estimators in scikit-learn. Notes.]

**from sklearn.utils.estimator\_checks import check\_estimator [**check based estimator]

**from sklearn.utils.validation import check\_X\_y, check\_array, check\_is\_fitted**

**[sklearn.utils.validation :**Input validation on an array, list, sparse matrix or similar. By default, the input is checked to be a non-empty 2D array containing only finite values.]

[ check\_X\_y, check\_array, check\_is\_fitted **: checking fit or not}]**]

**from sklearn.ensemble import RandomForestRegressor**

**[**A random forest regressor. A random forest is a meta estimator that fits a number of classifying decision trees on various sub-samples of the dataset and uses]

**from sklearn.metrics import r2\_score** [Best possible score is 1.0 and it can be negative (because the model can be arbitrarily worse)]

[R^2 (coefficient of determination) regression score function. ... A constant model that always predicts the expected value of y, disregarding the input features, would get a R^2 score of 0.0. Parameters: y\_true : array-like of shape = (n\_samples) or (n\_samples, n\_outputs) Ground truth (correct) target values.]

**from typing import Optional, Union [**A Python optional argument is a type of argument with a default value.]

The CSV dataset is collected from internet. There are 4 datasets: players, team, data, element type. Read these csv datasets by using pandas and print these players, team, data, element type datasets.

**train\_data = data[data.event\_id < 38]**

**test\_data = data[data.event\_id == 38]**

**print("Training data entries:", train\_data.shape[0])**

**print("Test data entries:", test\_data.shape[0])**

**We** are removing these coloumn ['timestamp', 'fixture\_code', 'kickoff\_time', 'opposition', 'event\_id']