NOTEBOOK

EA Sports

FIFA Player Position Classifier

T,
Learn

Knowledge Discovery through Brand Stories

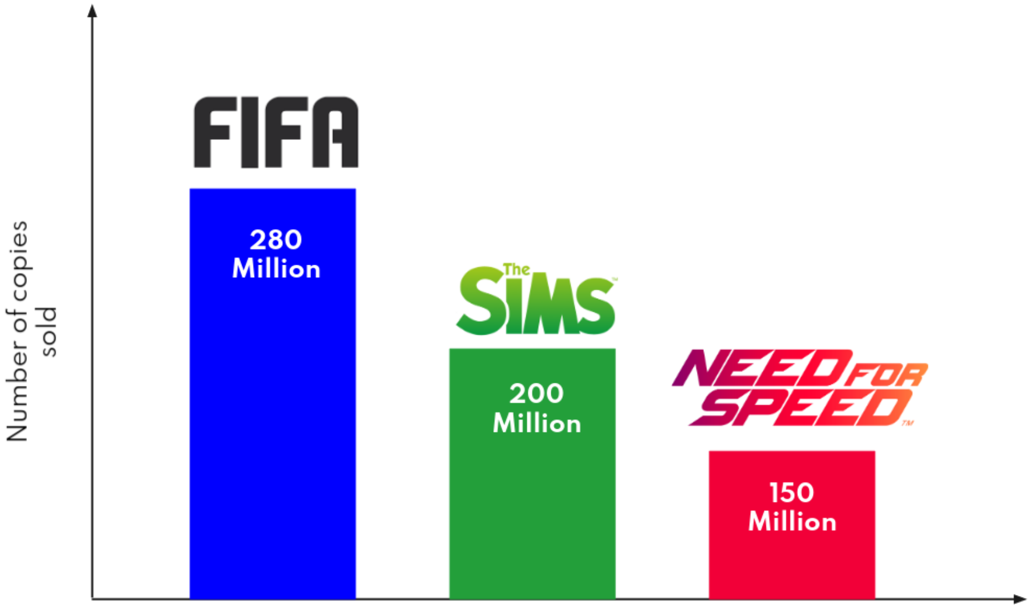
#1

Introduction

EA Sports is a division of Electronic Arts that develops and publishes sports video games. Formerly a marketing gimmick of Electronic Arts, in which they tried to imitate real-life sports networks by calling themselves the "EA Sports Network" (EASN)

FIFA is a series of association football video games developed and released annually by Electronic Arts under the EA Sports label. As of 2011, the FIFA franchise has been localized into 18 languages and available in 51 countries. Listed in Guinness World Records as the best-selling sports video game franchise in the world, the FIFA series has sold over 325 million copies as of 2021.

FIFA is best selling game franchise followed by SIMS and Need for Speed.

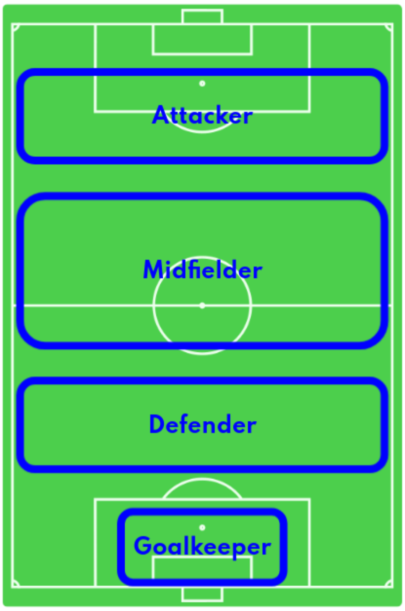


#1

Introduction

Starting from FIFA 20, EA Sports has given the option to gamers to create their own players. Pick any player – the race and gender you pick don't matter here. In the customization section of FIFA 20, you're able to fully design your own player. You can choose what clothes your player wears, but the tabs at the top also allow you to customize their name, sex, race, and appearance.

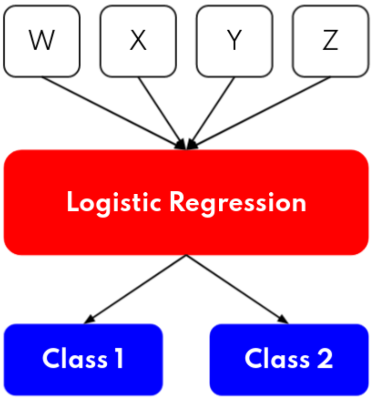
Apart from choosing skin, you can also change player attributes like stamina, dribbling, goalkeeping reflexes, ball control. These attributes will change player behavior on the soccer field. Also, the player's best position will also change. We will be using machine learning to predict the best position for this newly created player. We will train a model on current players' attributes and their current playing positions.

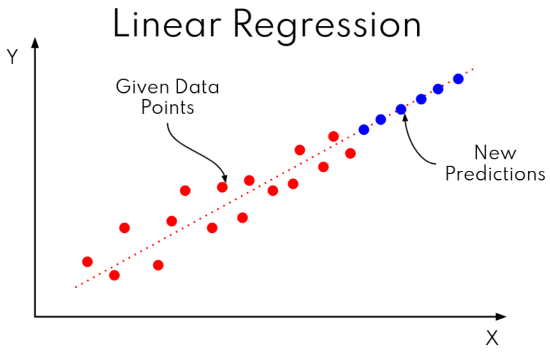


#2

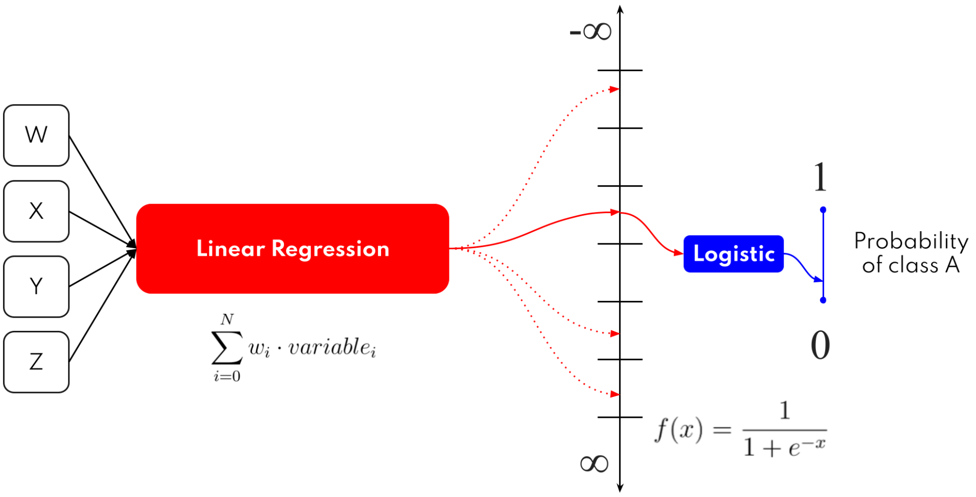
Logistic Regression

Logistic Regression is a classification machine learning algorithm that can only classify data points into two classes, that is it's a binary classification algorithm. It consists of two parts, we apply linear regression followed by a logistic function.





In linear regression, we try to fit a linear line with our training data points. After training, we will get output in range (-∞, ∞), but we wanted the output to be in the range [0,1], where 0 means class 1, and 1 means class 2. To change our output range we pass it through a special function called logistic regression

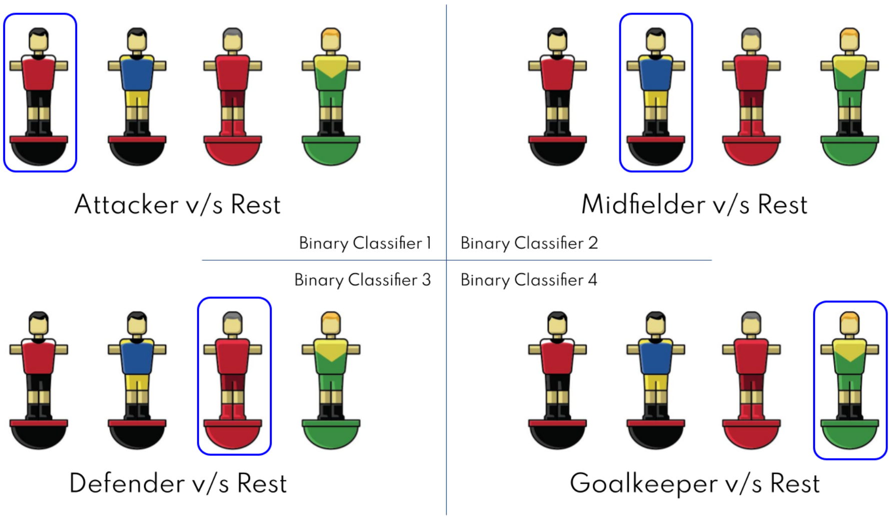

W
X
Y
Z
Linear Regression
$\sum_{i=0}^{N} w_i \cdot variable_i$
-∞
∞
Logistic
1
0
Probability of class A
$f(x) = \frac{1}{1+e^{-x}}$

#3

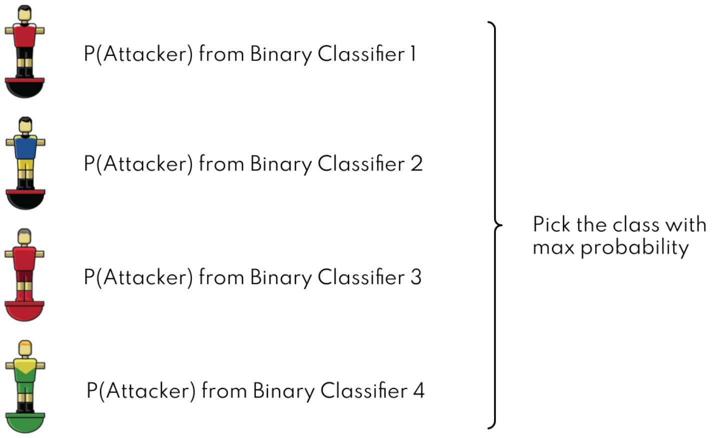
One v/s Rest

As you read on the previous page, logistic regression is a binary classification algorithm but our problem is a multiclass classification problem, where we have to classify players into four categories attacker, midfielder, defender, and goalkeeper. So we should use a different classification algorithm.

Instead of studying a complete new algorithm, we will use 4 logistic regression models. First model will check wether the player is an attacker or not, similarly other models will check other positions.



This is known as One v/s Rest, using we can use any binary classifier in multiclass classification problem.

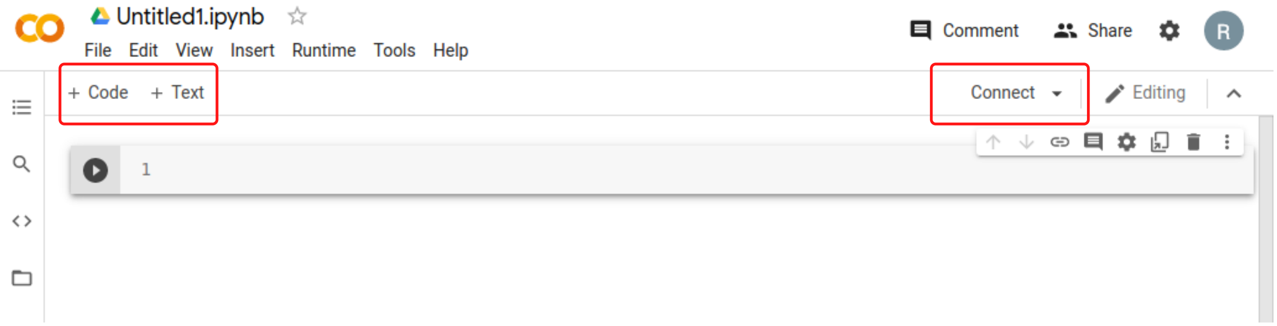

P(Attacker) from Binary Classifier 1
P(Attacker) from Binary Classifier 2
P(Attacker) from Binary Classifier 3
P(Attacker) from Binary Classifier 4
Pick the class with max probability

#Extra

Introduction to Colab

Nearly all machine learning and deep learning algorithms require good hardware. What if ... you don't have good hardware? Should you drop your dream to be a data scientist? No, there's an alternative Let us introduce you to Colab.

Colab is a service provided by Google which lets you access a virtual machine hosted on google servers. These virtual machines have dual-core Xeon processors, with 12GB of RAM. You can even use GPU for your neural networks. Colab is an interactive python notebook (ipynb), which means that with writing python code, you can also write normal text, include images.
To create a new colab notebook, just go to "https://colab.research.google.com", and create a new notebook. You will get something like below



You can connect to runtime by clicking the "Connect" button in the right corner. You can add a new Code cell, or text cell using respective buttons in the toolbar.

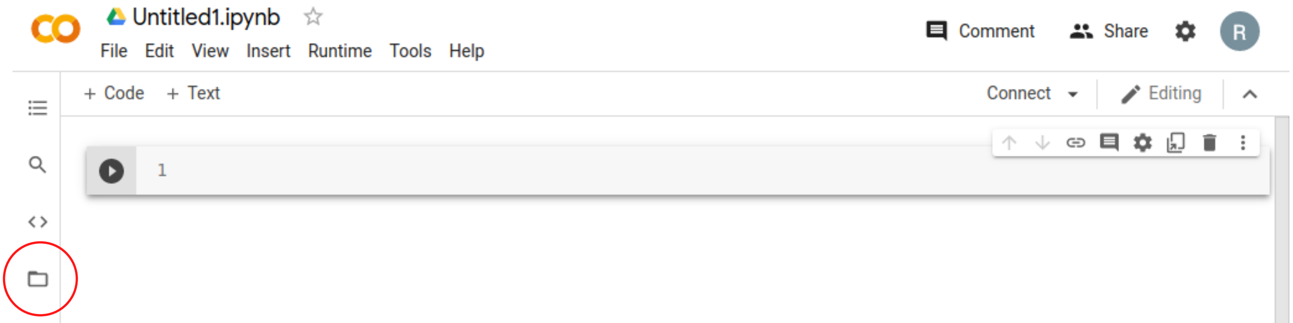
#Extra

Introduction to Colab

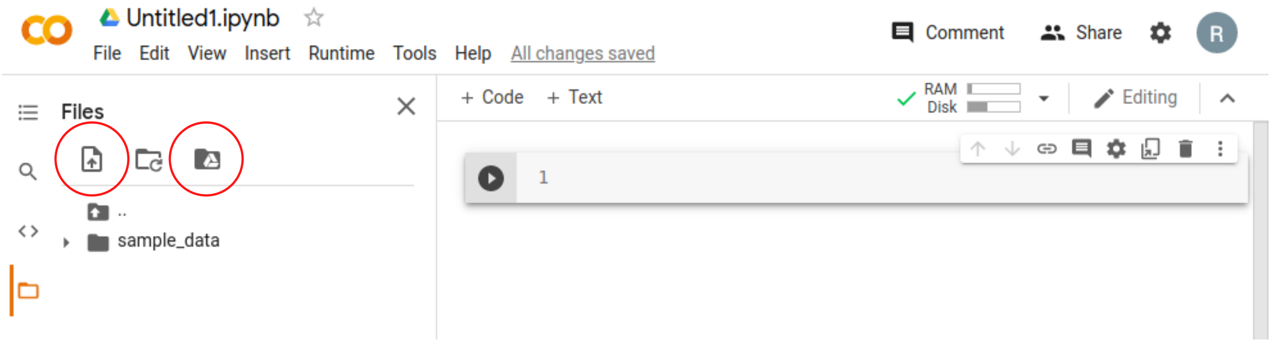
Uploading files

Sometimes you need to use a file from your PC. For that, you can upload the required files to google colab. Colab provides 100 GB in a colab session. If you want to access some files from your drive, you can even do so by connecting the drive to colab.

To upload something, open file pane from left toolbar.



Now, just upload the file using first button, and you can also mount google drive using last button



#4

Code

Importing libraries

Libraries to be used:

1. NumPy: Internally used by pandas to store dataframes
2. Pandas: A library to view dataframes and modify them
3. Matplotlib: A library for plotting data points
4. Seaborn: A library built over matplotlib for plots

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

Loading Dataset

Load the shared dataset "players_20.csv". This contains attributes and player playing positions.

```
data = pd.read_csv('players_20.csv')
data
```

Data Analysis

Let's find out median attributes of each playing positions

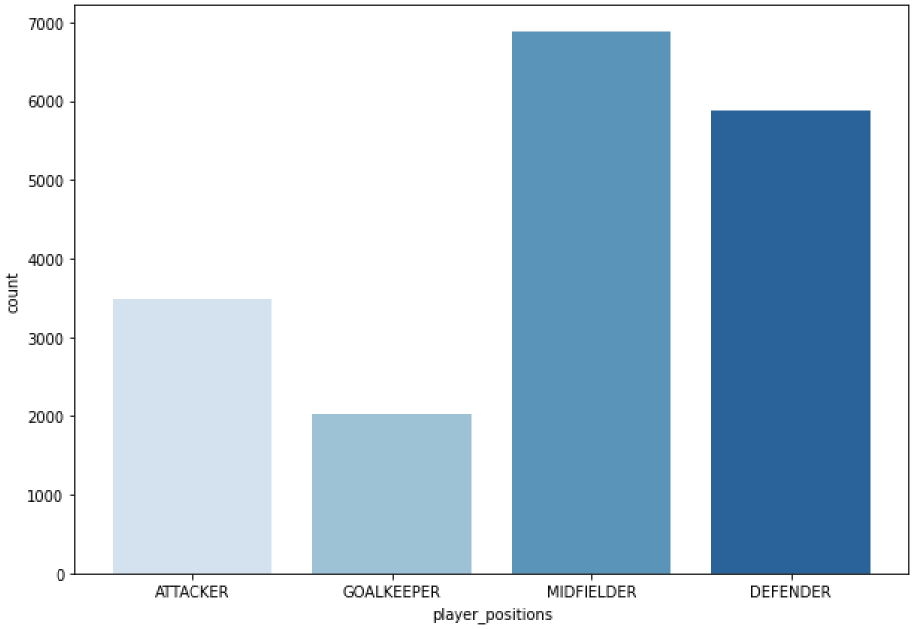
```
data.groupby(by='player_positions').median()
```

#4

Code

Let's see how many players are attackers, midfielder, defender, and goalkeeper.

```
plt.figure(figsize=(10,7))
sns.countplot(data=data, x='player_positions', palette='Blues')
plt.plot()
```



Maximum players in our dataset are midfielders, and defenders.

#4

Code

Data Preprocessing

Our player positions are in strings, and algorithms cannot understand text, so we need to convert these text to number classes.

We will replace player positions as follow:
Attacker -> 0
Midfielder -> 1
Defender -> 2
Goalkeeper -> 3

```
data['player_positions'] =
data['player_positions'].map({'ATTACKER':0,
                               'GOALKEEPER':1,
                               'MIDFIELDER':2,
                               'DEFENDER':3})
```

As we require only player attributes like stamina, we need to drop the columns that we don't require like short_names.

```
data.drop(['short_name','long_name','overall','value_eur','wage_eur']
, axis=1, inplace=True)
```

Now we have to create X, and y, where X will be player attributes, and y will be player position. We will also be splitting out dataset into training and testing sets.

```
X = data.drop('player_positions', axis=1)
y = data[['player_positions']]
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test =
train_test_split(X,y,test_size=0.2,random_state=42)
```

#4

Code

Model Training

Now we will import our logistic regression model, and train it with our player dataset. As we are dealing with a multiclass problem, we need to pass 'ovr' to the multi_class parameter.

```
from sklearn.linear_model import LogisticRegression
model = LogisticRegression(multi_class='ovr', max_iter=1000)

model.fit(X_train, y_train)
```

Training and Testing accuracy of trained model

```
score_train = model.score(X_train, y_train)
score_test = model.score(X_test, y_test)

print('Model accuracy on training set:
{:.2f}%'.format(score_train*100))
print('Model accuracy on training set:
{:.2f}%'.format(score_test*100))
```

We can also calculate f score

```
from sklearn.metrics import f1_score
y_pred = model.predict(X_test)

f1_score(y_test, y_pred, average='weighted')
```

T,

Learn

Knowledge Discovery through Brand Stories

Visit Us at :
techlearn.live

Email : support@techlearn.live
Phone : +91-9154796743