**Report Work On FIFA19 dataset.:-**

Before going to report work on each sections I will give you some idea about dataset.

We have downloaded the dataset and named it as 2020T2Data.csv.

Our dataset is redistributed dataset of Fifa19 released byKaggle (a data science community and competition platform). The dataset contains more than 18K rows and 64 columns (Redistributed dataset).

We first got this dataset by calling wget on given link (listed in code). We loaded it as a spark dataframe by reading it.

To do above we must have or installed wget, pyspark, jdk8 etc.

* 1. A.

1. Which are the (min, mean and max) for Age:-

In this part we got the results by applying SQL functions min(), max() and mean() on ‘Age’ column of our dataset.

The results are as :

“Minimum age is 16, mean of age is 25.122205.., and maximum age is 45”.

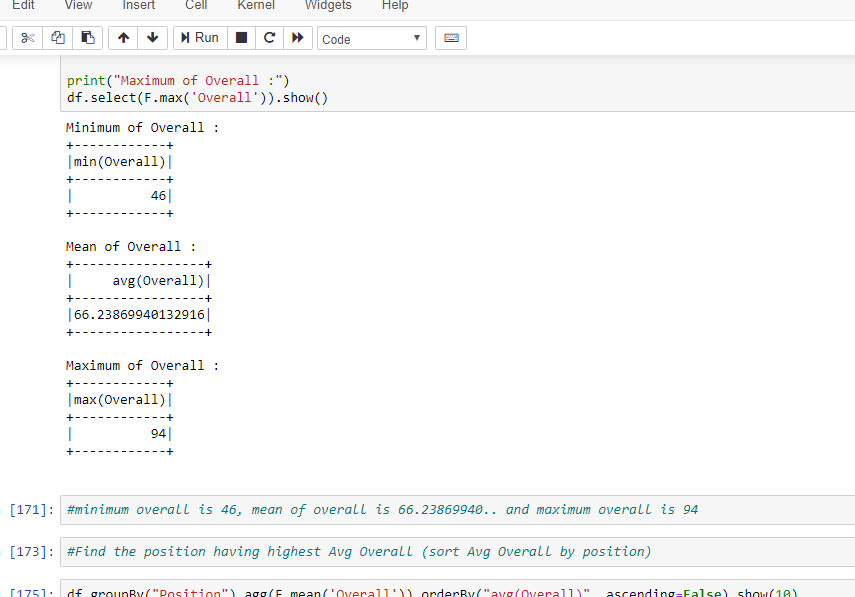


1. Which are the (min, mean and max) for Overall

In this part we got the results by applying SQL functions min(), max() and mean() on ‘Overall’ column of our dataset.

The results are as :

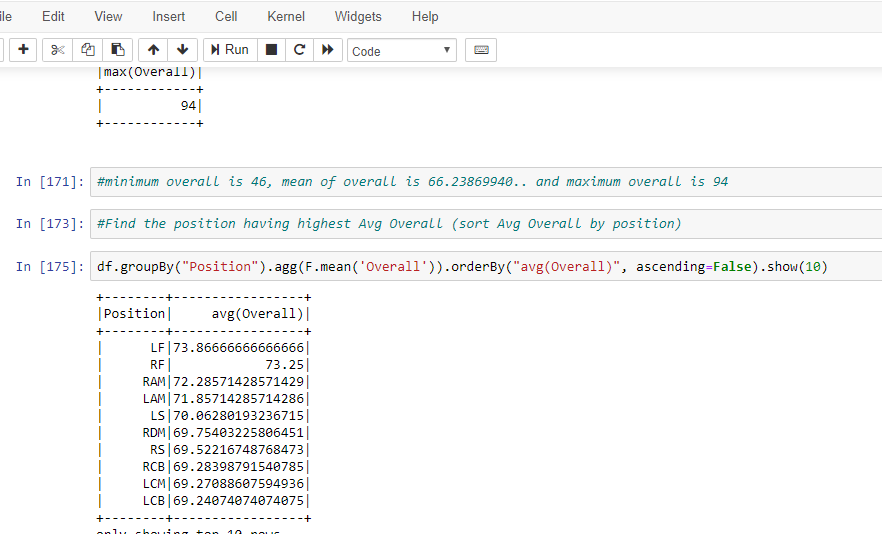
“Minimum overall is 46, mean of overall is 66.23869940.. And maximum overall is 94”



1. Find the position having highest Avg Overall (sort Avg Overall by position)

In this part I applied SQL function mean() on column ‘Overall’ with column ‘Position’ and got a dataframe which have ‘Position’ and ‘avg(Overall)’ columns. And got the result as:

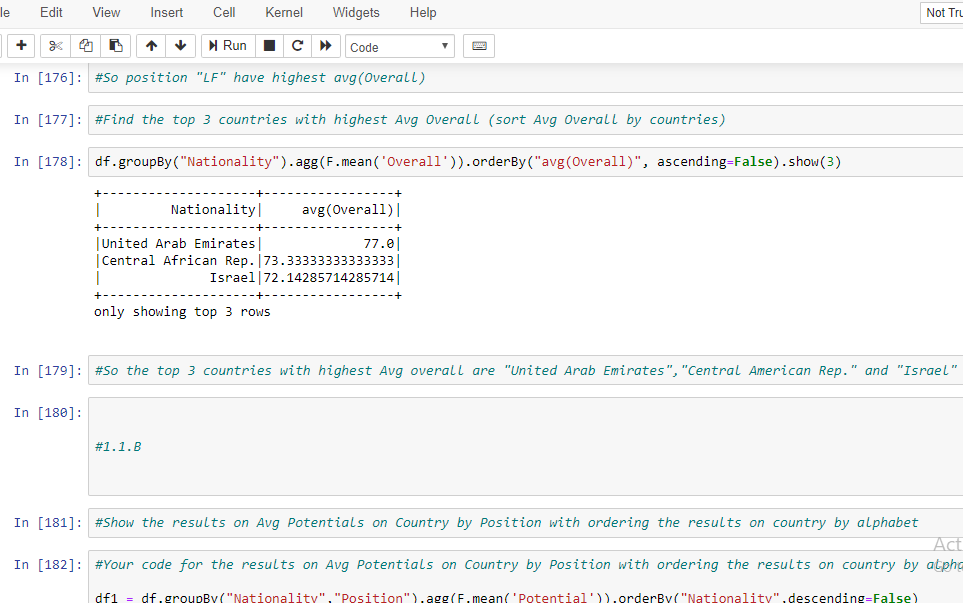
“Position "LF" have highest avg(Overall)”



1. Find the top 3 countries with highest Avg Overall (sort Avg Overall by countries)

In this part I applied SQL function mean() on column ‘Overall’ with column ‘Nationality’ and got a dataframe which have ‘Position’ and ‘avg(Overall)’ columns. And got the result as:

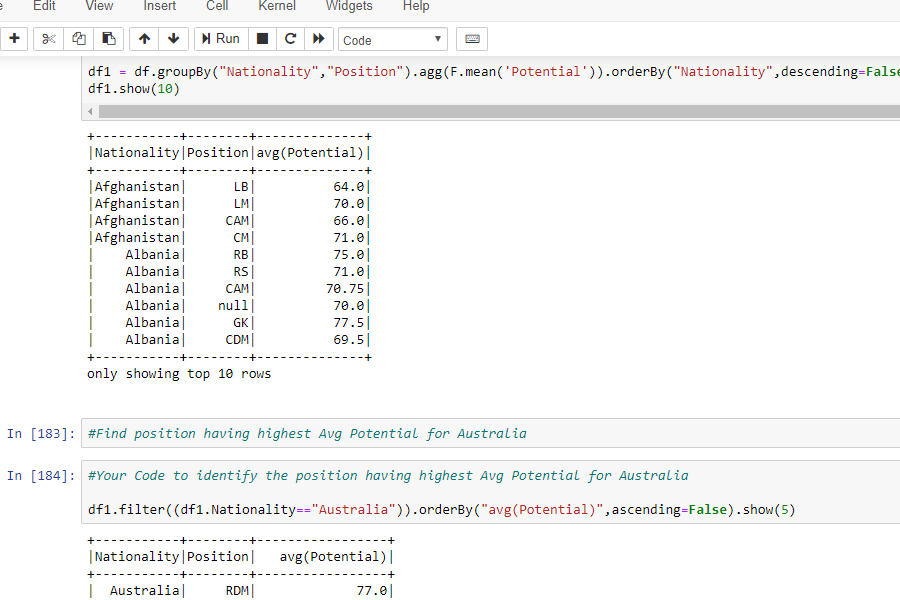
“The top 3 countries with highest Avg overall are "United Arab Emirates”, “Central American Rep." and "Israel"”



* 1. B.

1. Show the results on Avg Potentials on Country by Position with ordering the results on country by alphabet

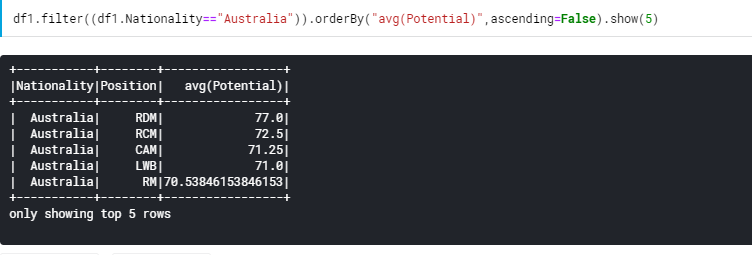
In this part I applied SQL function mean() on column ‘Potential’ with column ‘Nationality’ and ‘Position’ and got a dataframe which contains columns ‘Nationality’, ‘Position’ and ‘avg(Potential)’ in decreasing order.



1. Find position having highest Avg Potential for Australia

In this part I filter the previously got dataframe in order to achieve position having highest Avg Potential for Australia and got the result as:

“Position with Highest average potential for Australia is "RDM"”

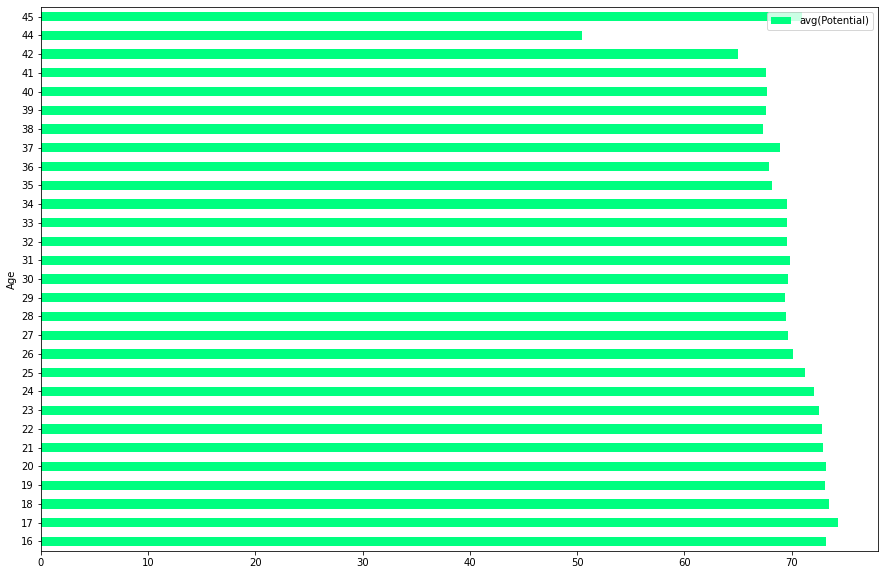


* 1. C.

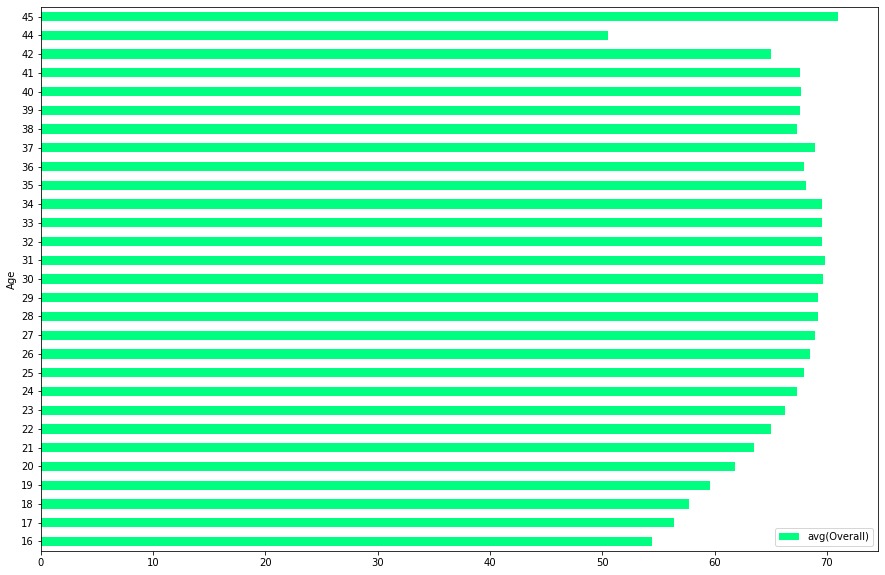
1. Plot the finding visually by having Age as the x-axis; the avg potential on age and the avg overall on age at y-axis.

In this part I have plotted a ‘barh’ plot on ‘Age’ at X-axis and ‘avg(Potentail)’ and ‘avg(Overall) at Y-axis. And got a graphs as:

For avg(Potential):



For avg(Overall):



1. Find the age when the players are fully released their potential in general?

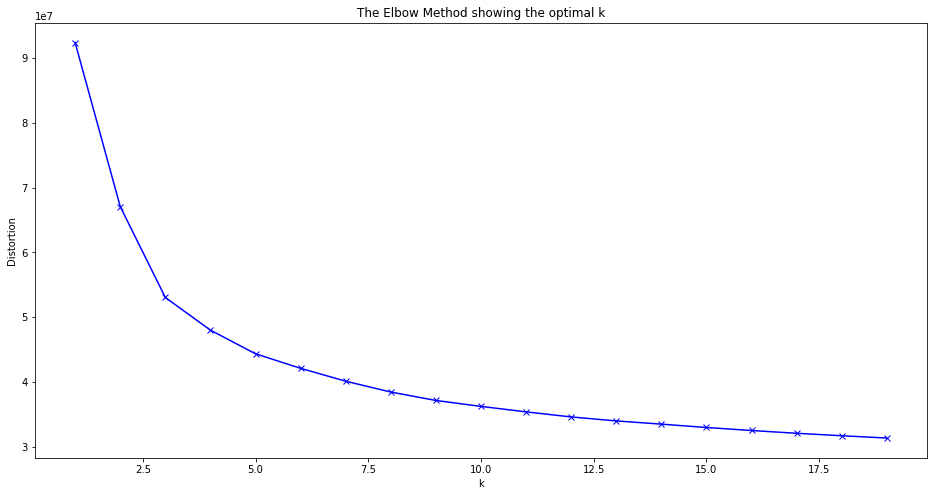
In this part I created a DataFrame of column ‘Age’ and difference between avg(Potential) and avg(Overall) and got the results as:

“At age of 31, 32, 33....42, 44, 45 players fully released their potential”

**Part 2 - Unsupervised Learning: Kmeans**

### 2.1 Elbow Plot:

In this part I created Elbow plot for KMeans.



### 2.2 K-Means

1. What you could find from those clusters (by cluster central point)?

In this section I applied KMeans Clustering for K = 8 (No. of Clusters). After applying it and plotting the plot I find that the values are clustered with their Centroids in a cluster.

1. Is there relationship between Position\_Group and a particular Cluster?

In my answers I got no relationship between Position\_Group and particular Cluster.

**3.2 Training Test Evaluation**

In this part first I prepared data as getting required columns and vectorising all the columns and got an additional column named as ‘features’.

After it I got a dataframe which have only two columns one is named as “Position’ and another is named as ‘features’ by select() function.

After it I applied StandardScaler() feature on features column to standardize it. Then created a new column named as ‘Target’ by labeling positions.

After it split the dataset in train (70%) and test (30%) dataset.

Then applied LogisticRegression on dataset.

1. **The confusion matrix**

In this part we got confusion matrix by calling confusion\_matrix function of metrics on ‘Target’ and ‘Prediction (Output from Logistic Regression)’

And got the results as.

Confusion matrix :

array([[ 821, 4, 207],

[ 0, 1432, 283],

[ 202, 270, 1617]])

1. **The Precision, Recall and F1 score**

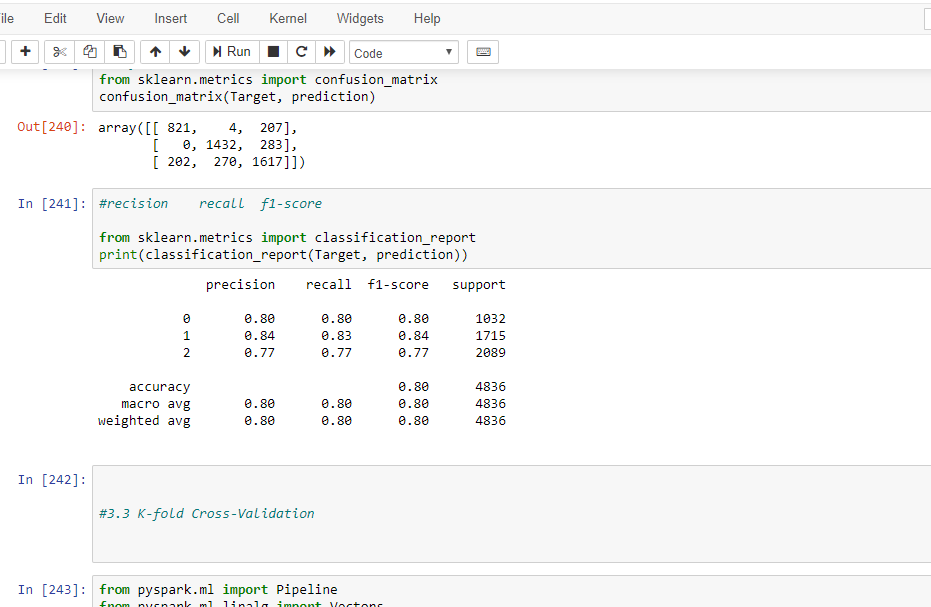
In this part applied classification\_report function of metrics on ‘Target’ and ‘Prediction’ and got the results as:

precision recall f1-score

0 0.80 0.80 0.80

1 0.84 0.83 0.84

2 0.77 0.77 0.77

****

**3.3 K-fold Cross-Validation**

To start this part of coding I have imported all the required libraries of pyspark like pipeline, vector, regession evaluator, linear regression, cross validator, param grid builder. Then applied 3- Fold cross Validator on the dataset and got the training summary.

Next we will apply Random Forest Regressor to classify our model. To do so first we import Random Forest Regressor. Then apply this random forest function onto labelCol and featureCol, after it evaluating the results.

After it we again imported required library for this part like Logistic Regression, One Vs Rest and Multi Class Classification Evaluator then we took a data frame and split it into train and test at random split of 0.8 (80% train) and 0.2 (20% test).

Then initiated both the classifier (Logistic Regression and OneVsRest).

After it we trained the multiclass model by fitting train data into OneVsRest classifier and scored the model on test data by transforming onto test data. Then evaluated accuracy of the model.

To apply above model on our dataset there were some difficulties I got. Some of them were like number of K Folds, model selection etc.

There are lots of many tasks we can still perform on our dataset like:

1. WE can find correlation between dataset’s features.
2. We can get release clause as high, medium and low or we can rank acoording to it in range of 1 to 10 or 1 to 5. Etc..

**Section - 4.**

In Manchester United De Gea has highest value 72M and with Wage of 260K and is raising star for his team.

Country with most players is England and has total 1662 players.

FC Barcelona is highest reputed club internationally.

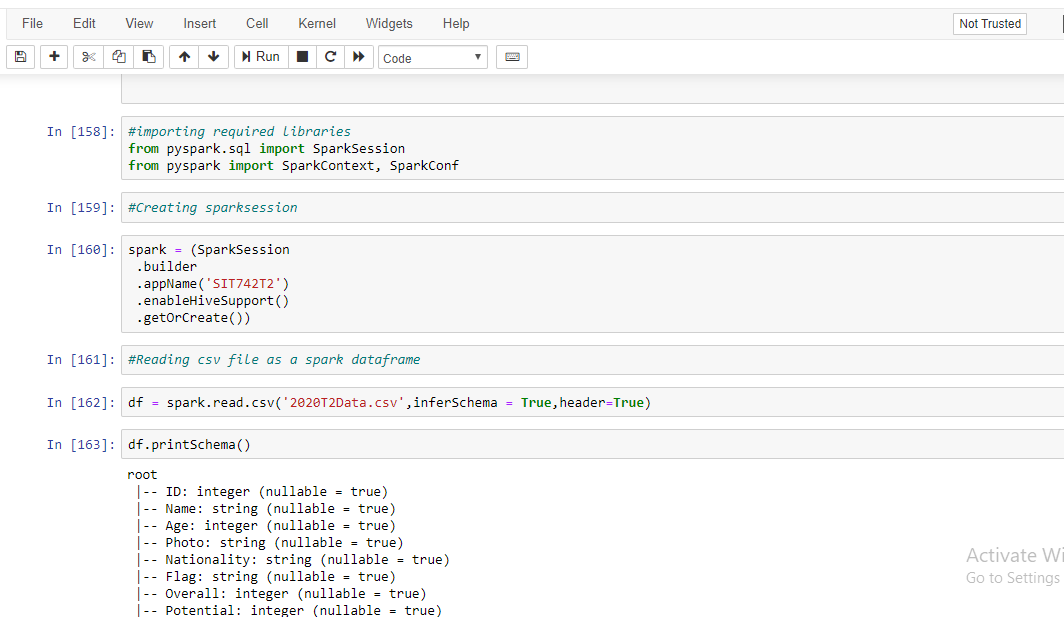
The eldest player is O. Pérez (45, Club - Pachuca, Country - Mexico).

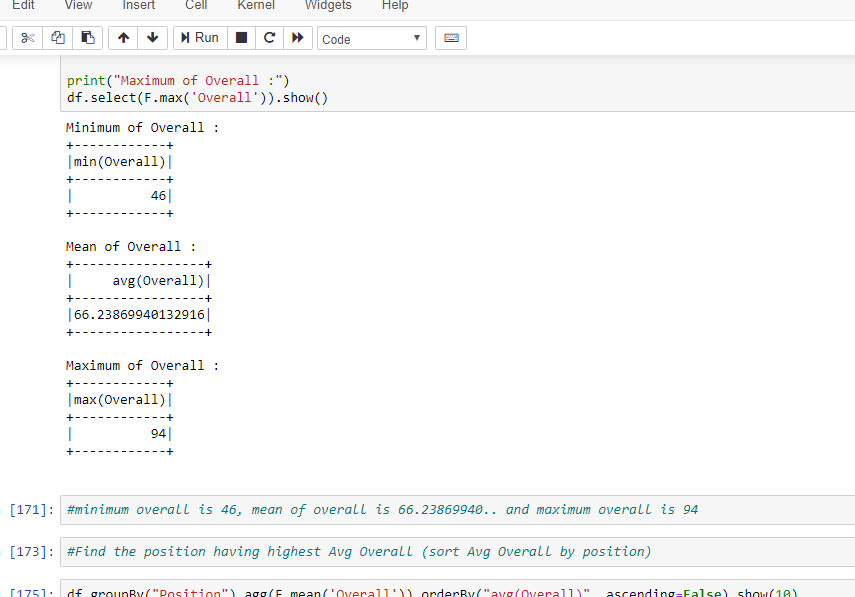
**Section - 5**

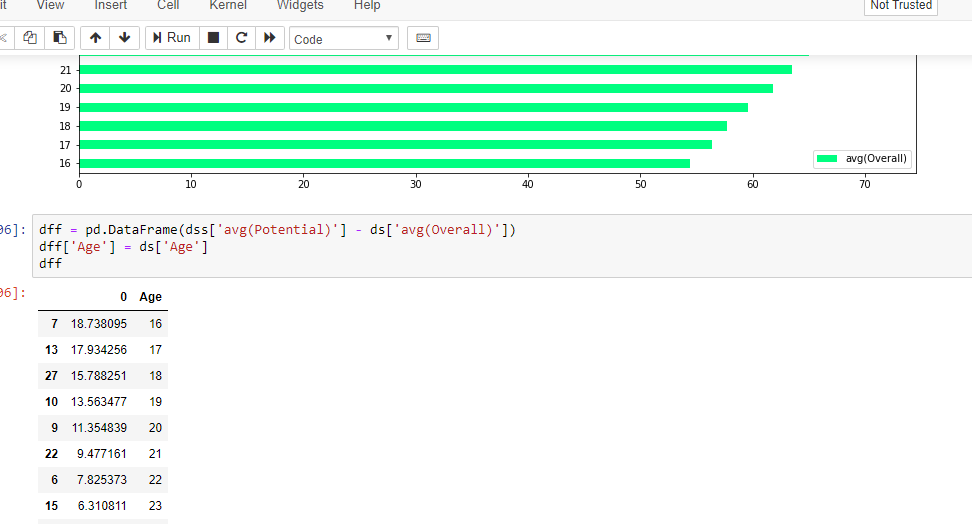
**…**

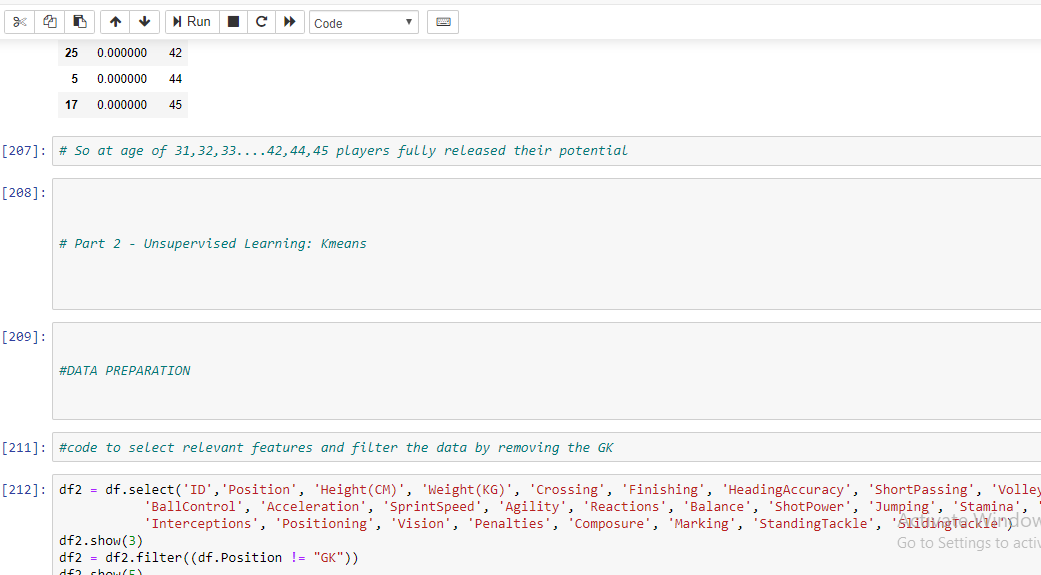
**Screenshots:**

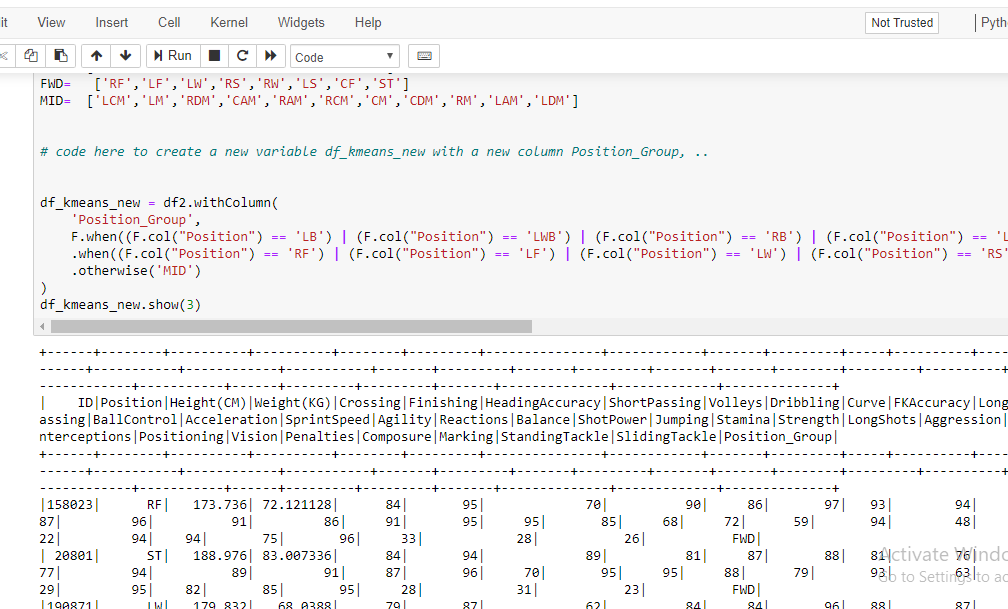
Some of screenshots of our code given below:

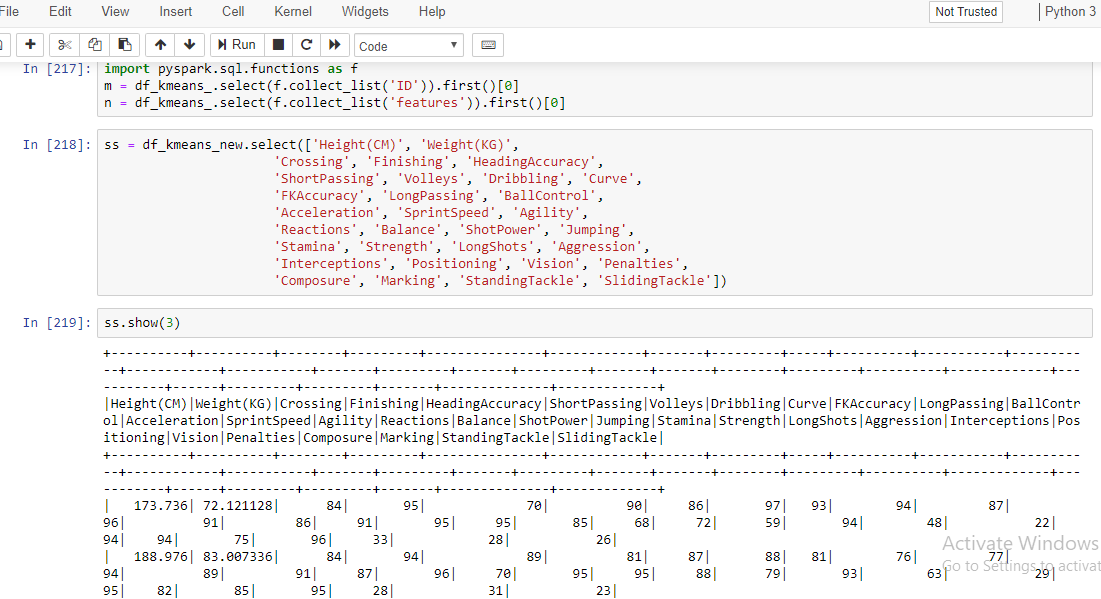


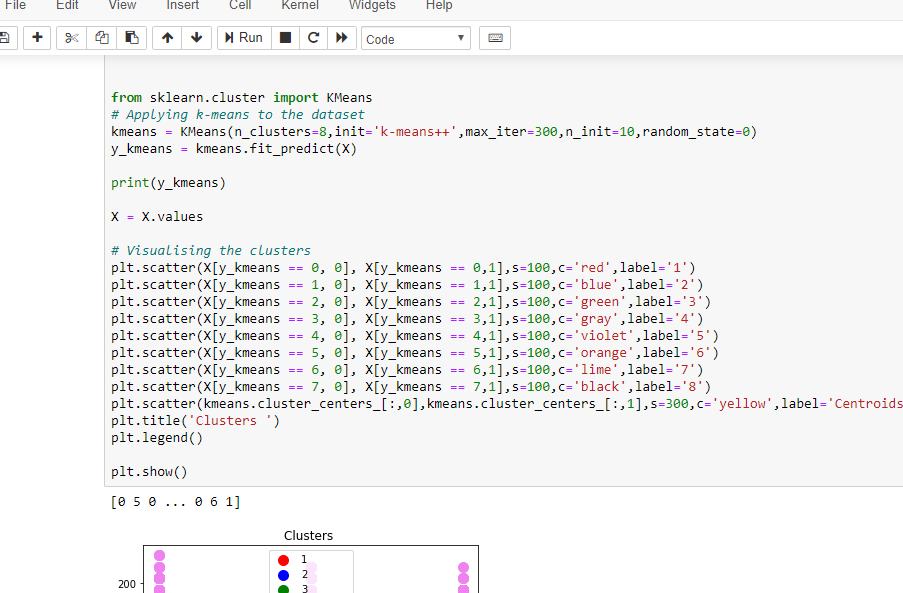


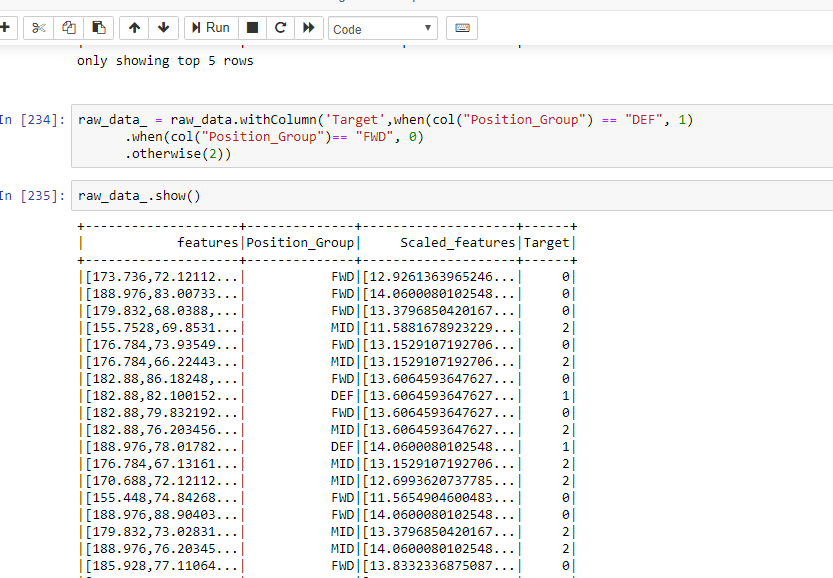












**References:**

1. <https://scikit-learn.org/stable/modules/generated/sklearn.metrics.confusion_matrix.html>
2. <https://medium.com/@mandava807/cross-validation-and-hyperparameter-tuning-in-python-65cfb80ee485>
3. <https://stackoverflow.com/questions/39540952/pyspark-creating-a-k-means-clustering-model-using-spark-ml-with-spark-data-fram>
4. <https://stackoverflow.com/questions/41540751/sklearn-kmeans-equivalent-of-elbow-method>
5. <http://spark.apache.org/docs/latest/api/python/pyspark.sql.html?highlight=filter>
6. <https://github.com/tulip-lab/sit742/blob/master/Assessment/2020/SIT742Task2.ipynb>
7. <https://intellipaat.com/community/7521/spark-dataframe-groupby-and-sort-in-the-descending-order-pyspark>
8. <https://stackoverflow.com/questions/36707344/split-rdd-for-k-fold-validation-pyspark>
9. <https://stackoverflow.com/questions/52740966/how-to-perform-cross-validation-without-using-paramgrid-builder-in-pyspark>
10. <https://stackoverflow.com/questions/37919250/spark-k-fold-cross-validation>
11. <https://spark.apache.org/docs/latest/ml-classification-regression.html>