Machine learning on Cloud

Submitted by

Norbert Szilvasi

22073183

Contents

[Task 1 – Cloud feasibility study 3](#__RefHeading___Toc6459_9171070751)

[1.1 – Cloud computing 3](#__RefHeading___Toc6459_9171070752)

[1.2 – Machine learning-as-a service 3](#__RefHeading___Toc6459_91710707521)

[1.3 – Amazon SageMaker 3](#__RefHeading___Toc6459_91710707522)

[1.4 – Microsoft Azure Machine Learning Studio 3](#__RefHeading___Toc6459_917107075221)

[1.5 – Google Cloud AutoML 3](#__RefHeading___Toc6459_9171070752211)

[Task 2 – Data analysis and opportunity identification 6](#__RefHeading___Toc6467_917107075)

[2.1 – Features and data types 6](#__RefHeading___Toc6467_9171070754)

[2.1 – Literature review 11](#__RefHeading___Toc6467_9171070751)

[Task 3 – Data preprocessing 13](#__RefHeading___Toc6469_917107075)

[3.1 – Preprocessing steps 13](#__RefHeading___Toc6459_91710707524)

[3.2 – Feature engineering 19](#__RefHeading___Toc6467_9171070751211)

[Task 4 – Model selection and training 30](#__RefHeading___Toc6469_9171070751)

[4.1 – Scored and received goal models 31](#__RefHeading___Toc6467_91710707513)

[4.2 – Final shoot-out model 37](#__RefHeading___Toc6467_917107075131)

[4.3 – Monte carlo simulation 40](#__RefHeading___Toc6467_917107075132)

[Task 5 – Model evaluation and visualization 40](#__RefHeading___Toc6469_9171070752)

[Task 6 – Model deployment 42](#__RefHeading___Toc6469_9171070753)

[Appendix: Google Colab file attachment 44](#__RefHeading___Toc6499_917107075)

[References 45](#__RefHeading___Toc6501_917107075)

# **Task 1 – Cloud feasibility study**

# **1.1 – Cloud computing**

Cloud computing helps with the distribution of different services such storage, servers and databases throughout the internet. Reliability of accessing these services is delivered by many well-known service providers (Guptha, Murali and Subbulakshmi, 2021). Amazon Web Services, Microsoft Azure, Google Cloud Platform will be the main service providers examined in this paper. Each cloud service has a specific networking and safety plan with different operating costs and characteristics to protect against different types of cyber attacks. Development of the cloud service has decreased costs, provided an efficient model for a business organisation, and delivered a higher level of flexibility. Many businesses have introduced cloud services and profited from it. With the implementation of cloud service, businesses are strengthening cross-platform cooperation between experts, enabling them to create more innovative solutions on IT infrastructure, which in exchange allows them to increase their businesses and boost their revenue (Surbiryala and Rong, 2019).

# **1.2 – Machine learning-as-a service**

Machine learning-as-a service (MLaaS) cloud platforms basically offer machine learning (ML) as a web service to consumers who would like to train ML models. Customers normally perform ML exercises with the help of a web interface (Noshiri, Khorramfar and Halabi*,* 2021). MLaaS generally is a very efficient business strategy because we can either pre-train the model ourselves or buy an already pre-trained model for a cheaper price, which is more beneficial.

The most famous and well-known cloud service providers that have ML service are Google, Microsoft and Amazon.

# **1.3 – Amazon SageMaker**

Amazon SageMaker is a ML service provider that enables quick and easy building, training, and deployment of machine learning models in the Amazon Web Service (AWS) cloud at any scale. It offers tools and settings for different activities such as to wrangle data, store feature, detect bias, develop, train, tun and deploy ML models (Tesser, Marques and Borin, 2021). SageMaker enables all the main three types of machine learning. Built-in SageMaker algorithms enable data scientist to personalise, add and change own algorithms, datasets and execute algorithms using its deployment characteristics.

# **1.4 – Microsoft Azure Machine Learning Studio**

Many ML algorithms can be accessed by Azure Machine Learning Studio together with modules for data input, output, preprocessing, data visualisation. It is possible to create an advanced analytics project, iterate and work in it, and employ it for model training applying these elements (Pliuhin *et al.* 2021). Junior developer and data scientists can quickly build, train and deploy algorithms avoiding the complex structure of instance management in cloud, Python coding, and the use of Jupiter Notebook. This results in a quicker ML algorithm’s construction, development and execution.

# **1.5 – Google Cloud AutoML**

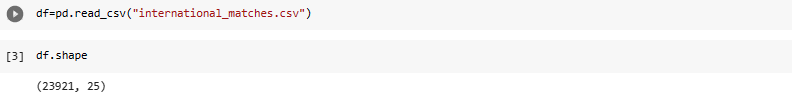
Cloud AutoML is a user-friendly ML platform to enable individuals with modest ML expertise to build industry-specific algorithms. This device allows users to reach the Google’ s research and tailor the outcomes in the line with their individual requirements. It has the ability to recommend built-in algorithms with the help of a collection of APIs. Google Cloud Platform is one of the most important and significantly increasing cloud APIs globally (Challita *et al.* 2018). APIs textual documentations. Out of the cloud APIs, Google Cloud Platform (GCP) is one of the most essential and rapidly increasing cloud market recenlty. It provides developers several products to build a range of programs from simple websites to complex worldwide distributed applications. GCP offers hosting services on the same supporting infrastructure that Google uses internally for end-user products like Google Search and YouTube (Noshiri, Khorramfar and Halabi*,* 2021).

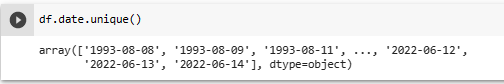


source: https://cloudwithease.com/aws-vs-azure-vs-gcp-the-3-big-cloud-providers

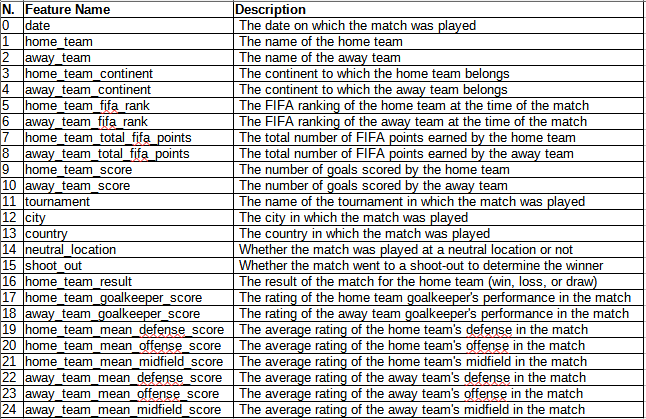
# Task 2 – Data analysis and opportunity identification

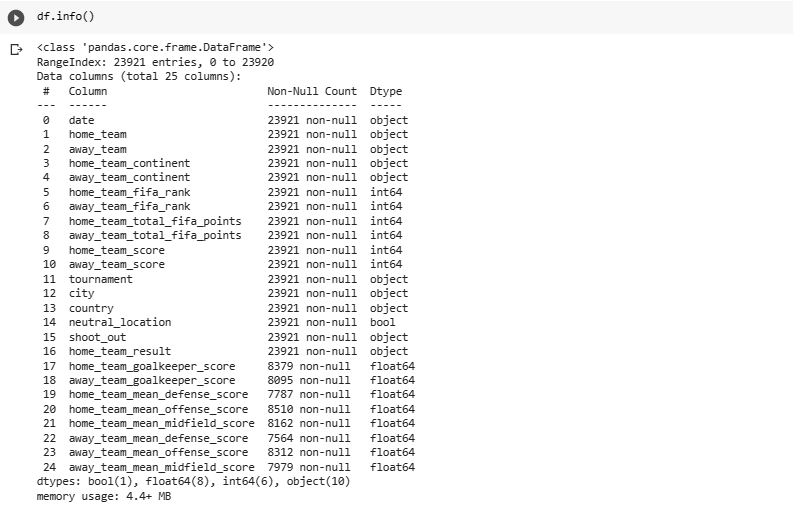
# **2.1 –** **Features and data types**

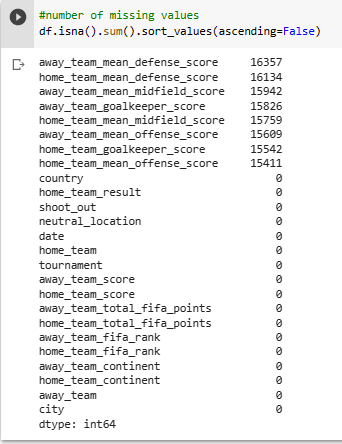
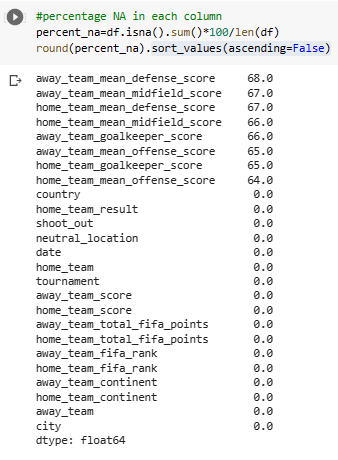
The dataset contains data between 1993 and 2022.



The summarised and detailed table of the dataset’s features.

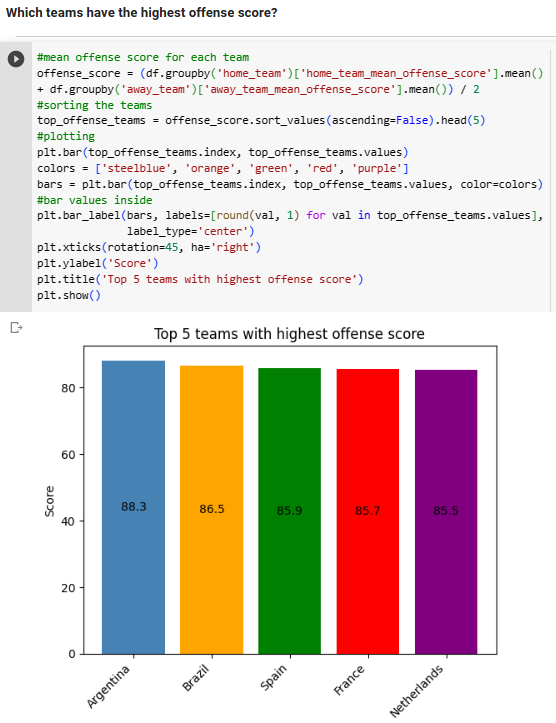
****

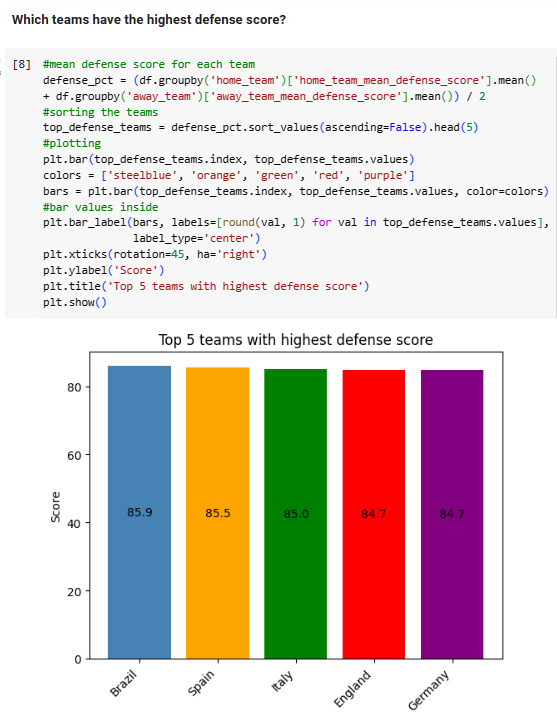
****

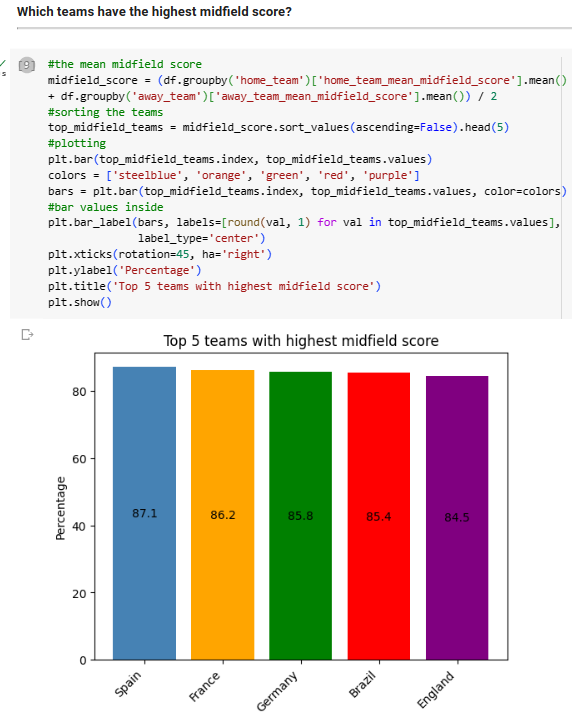
****

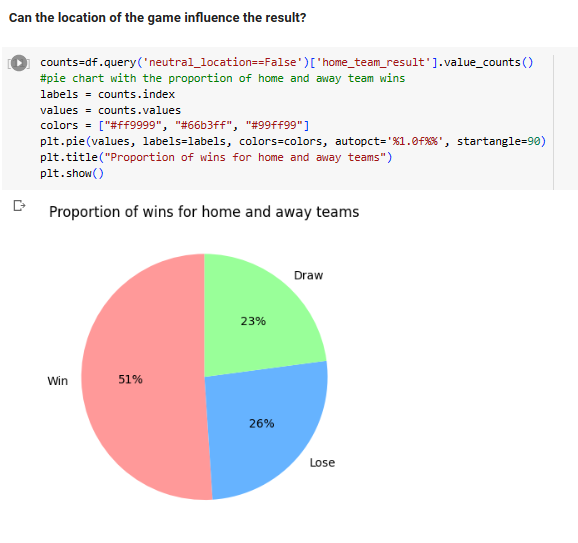
There are many missing values in the dataset.

****

****

****

****

****

# **2.1 – Literature review**

Football is one of the most popular sport activities in the globe (Constantinou, 2019). Machine learning (ML) techniques have been used in many studies to forecast the victor of a football game.

Zhang *et al.* (2019) examined 1,440games from Super League in China between 2013 season and 2016 season. They applied fourteen parameters and three main ML techniques (Support Machine Vector, Random Forest, and Bayesian network) to forecast the game with 76.9% accuracy.

A Deep Learning-based research was conducted by Rahman (2020) to forecast the result of games in 2018 of World Cup utilising freely accessible datasets. The writers recommended artificial neural network (ANN) and long short-term memory-based (LSTM) algorithm with an accuracy rate of 63.3%. Stübinger *et al.* (2020) suggested using various ML algorithms to estimate the result of football games based on player and game-specific characteristics as well. A model presented by Tiwari, Sardar and Jain (2020) suggest the usage of Recurrent Neural Networks (RNNs) and LSTM to forecast the outcome of soccer. To increase the accuracy rate of algorithm, the researchers analyse all the game-related activities and their impacts of the total score.

Danisik, Lacko and Farkas (2018) applied LSTM to estimate the result of games. The researchers created a dataset from a number of places and separated its parameters in the line with the players’statistics and game records. An accuracy rate of 52.4% was obtained using a dataset from many European Leagues between 2011 and 2016.

Kinalio˘glu and Coskun (2020) contrasted six predictive machine learning models and analyzed models with various classifier values applying different evaluation models. Beal et al. (2021) increased the predictive performance by 7% by taking specialists and personal viewpoints from media which is self-published and including climatic, players’ attitude, competitors, and external factors in the forecasts.

Data show that the Gaussian Naive Bayes Approach is able to forecast the outcome of a soccer game with 85.43% accuracy, which is slightly better than the decision tree classifier’s of 79.81 % (Athish *et al.* 2023). Lasek (2016) contrasted the results of ordinal regression evaluations and least squares scores, along with a distribution based on Possion, to make forecasts for the Euro 2016 matches and model the event many times applying Monte Carlo simulations.

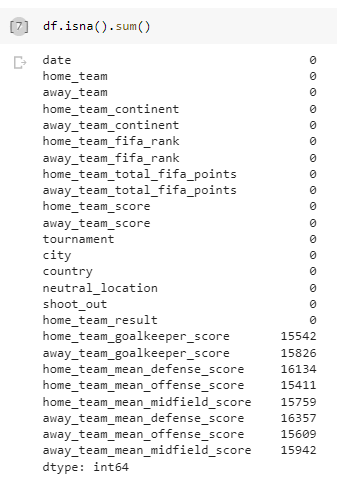
Adam (2016) obtained game forecasts and modelled the outcomes of a tournament applying a simple Generalised Linear Model that was developed applying gradient descent. Having only a small number of features, he was able to achieve decent outcomes, and he suggests using further attributes and applying an attribute selection technique.

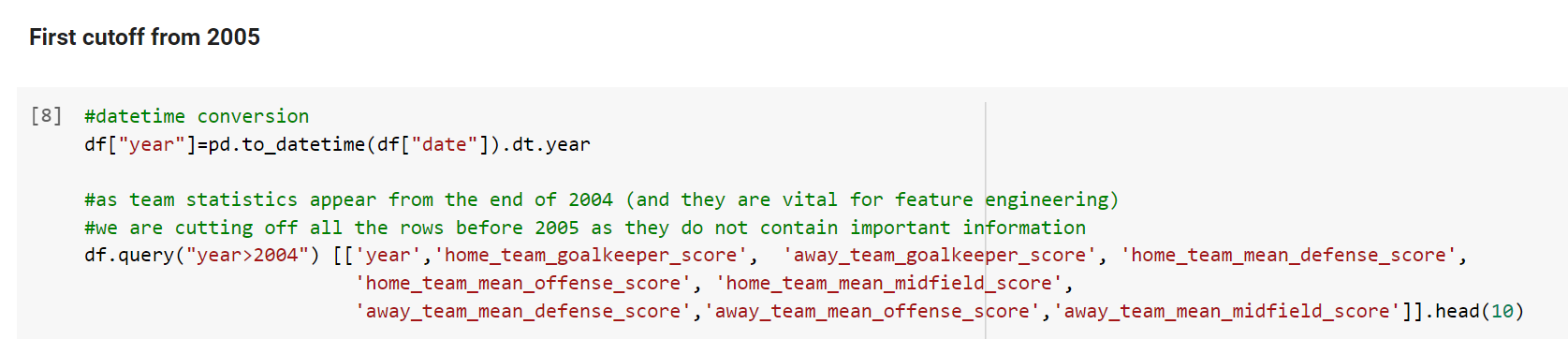
# Task 3 – Data preprocessing

Data preprocessing is a crucial step in the machine learning pipeline where raw data is transformed, cleaned, and pre-processed to prepare it for further analysis and modelling. The quality and accuracy of the results of a machine learning model heavily depend on the quality of the data used to train the model. Data preprocessing is method of converting raw data into a form that can be interpreted. Data gathered from the real world might occasionally be incorrect, inconsistent, redundant, and incomplete. Data preprocessing helps with turning raw data into a cleaned and understandable forms (Agarwal, 2015).

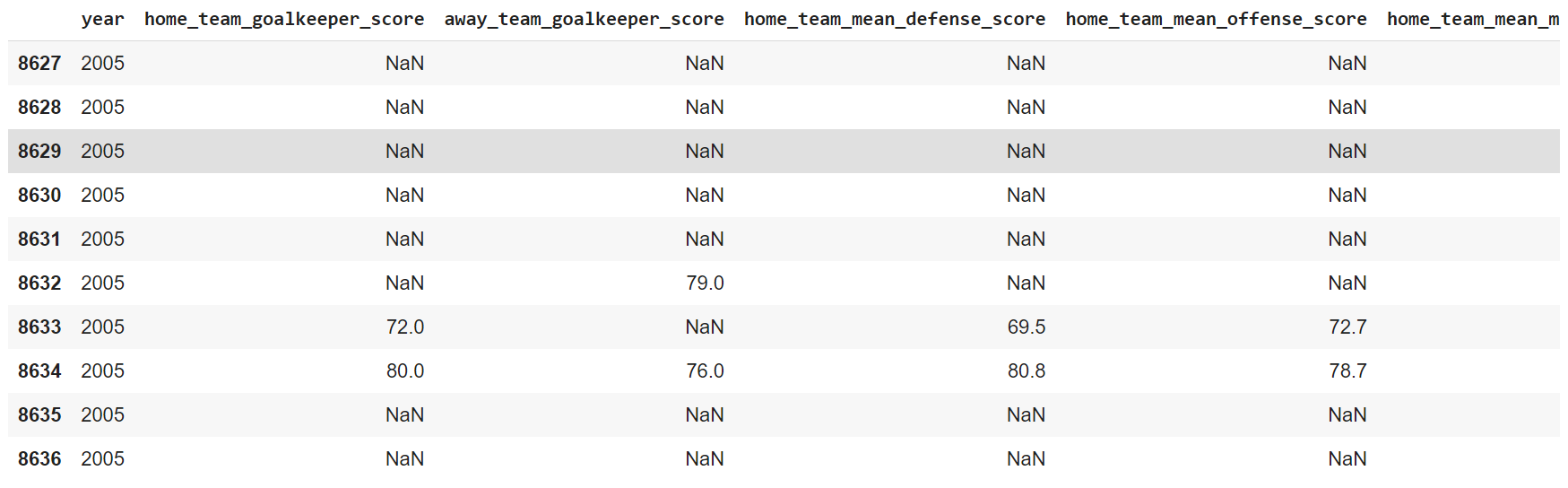
# **3.1 – Preprocessing steps**

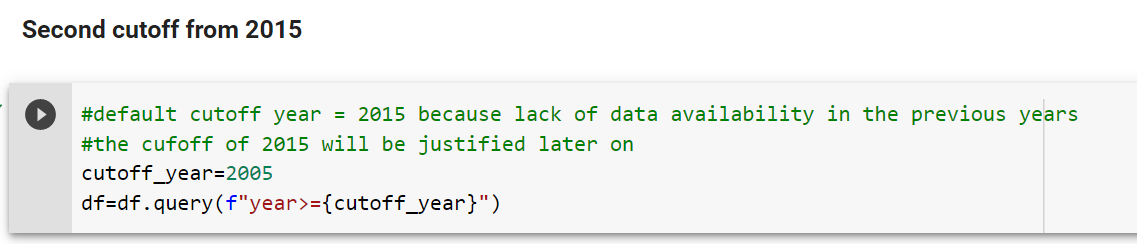
The dataset contains many missing values.

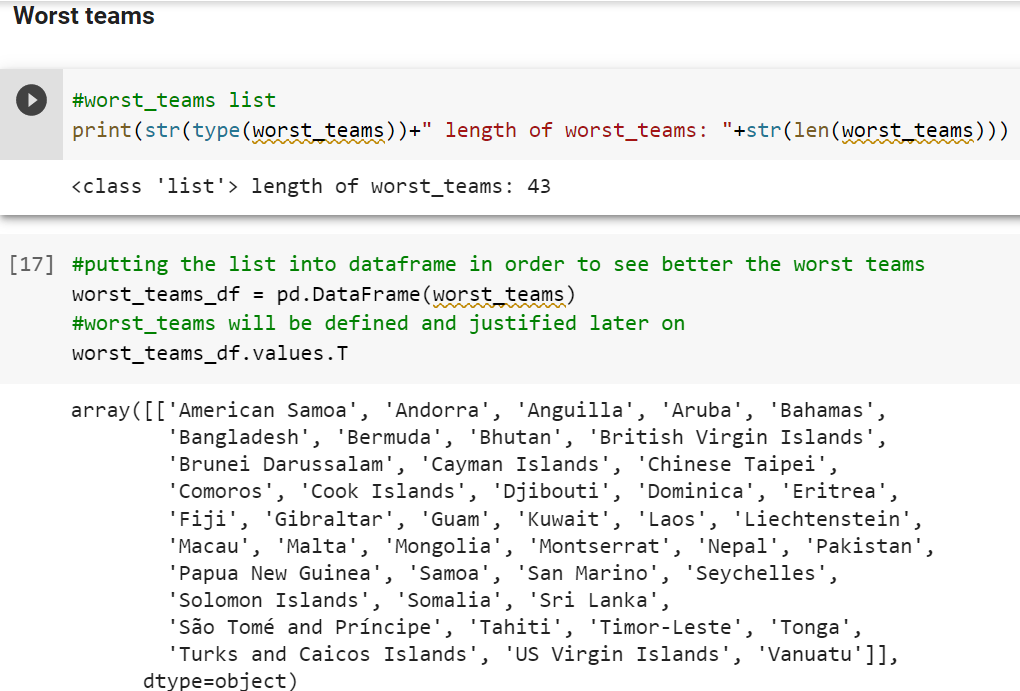




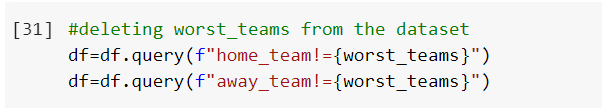
The first cutoff was used here from 2005. As team statistics are available only from 2005, therefore, the data must be cut off before 2005. This is very important for feature engineering at the later stage to have available data.

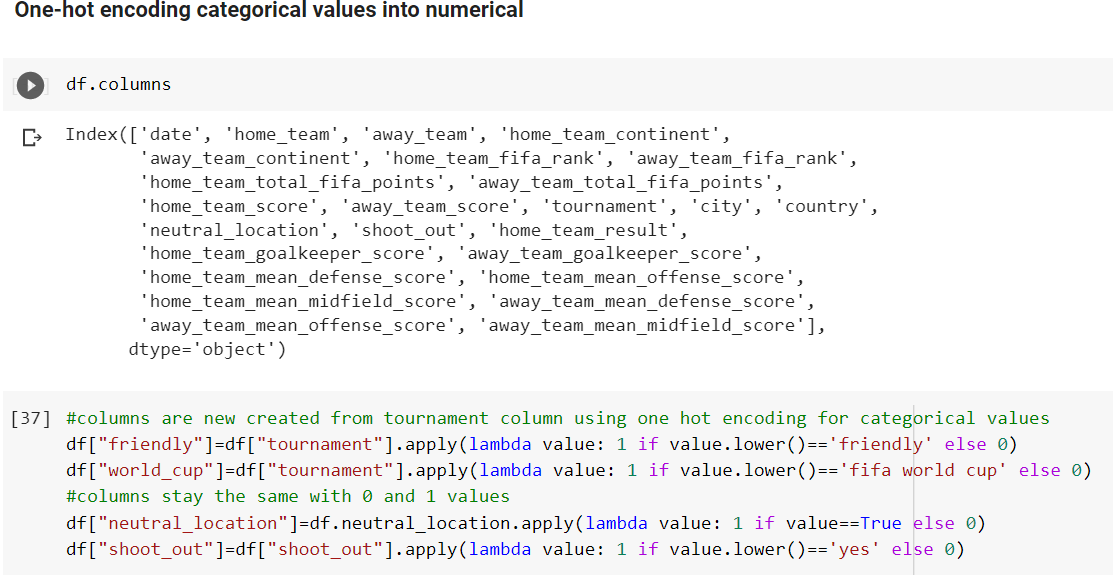
 Then another cutoff is used from 2015 which will be justified later on with data.

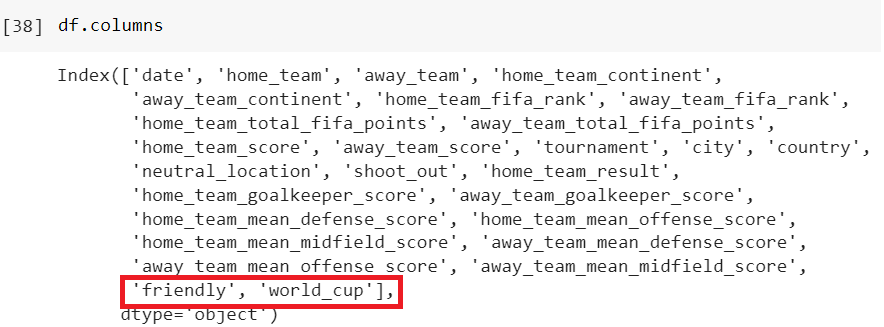
 Worst teams will be justified with data later on as well and will not be the part of the dataset based on analysis and supported numerically.

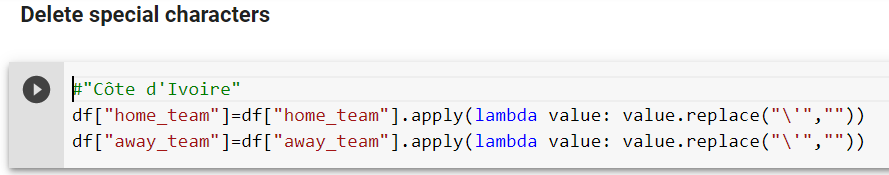


The worst\_teams are deleted from the dataset now in order to preprocess the data.

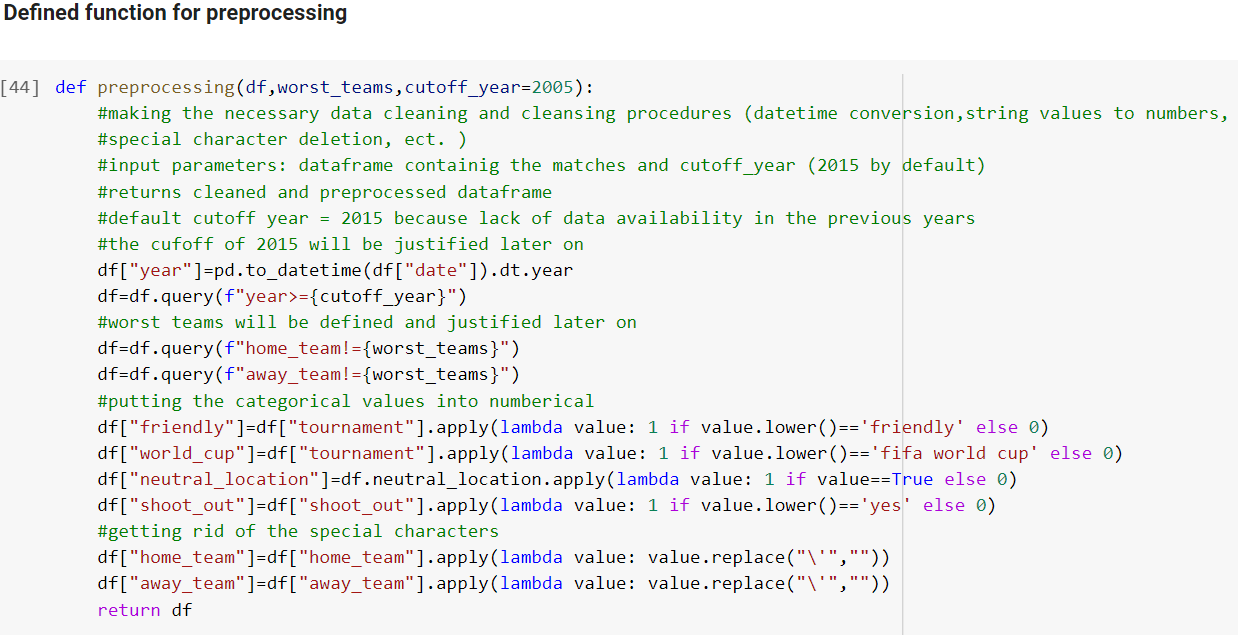


 Two new columns are added to the dataset. Those new features will be used later in the model.



The home and away teams contain special character which must be replaced in order to avoid any further confusion in the model later on.

All the preprocessing steps can be put and used in one function. Worst\_teams and cutoff year of 2015 will be justified later on. Worst\_teams are used here without justification. Cutoff year of 2005 is used here which will be changed to 2015 later on.



As during the FIFA World Cup related games the maximum scored and received goal is 7, but there are many games in the dataset, where the scored and received goals are more than 7. These can be considered as outliers because FIFA World Cup related games based on the data will not go over more than 7 goals because the teams most likely the be stronger than other teams in various competitions. In order to avoid these outliers, cap is used for the received and scored goals above 7 scores in a game to a fix 7 goals cap. Those data can might contain important information for the model, therefore, these games can be kept by using the cap of 7 goals in such cases.

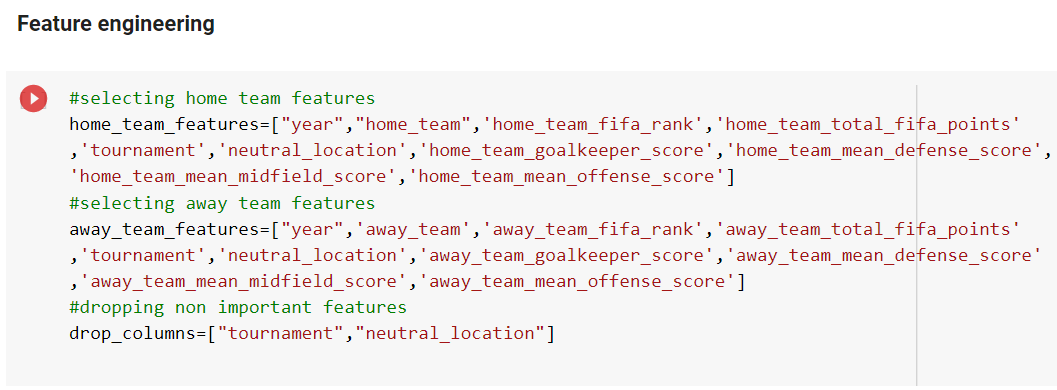


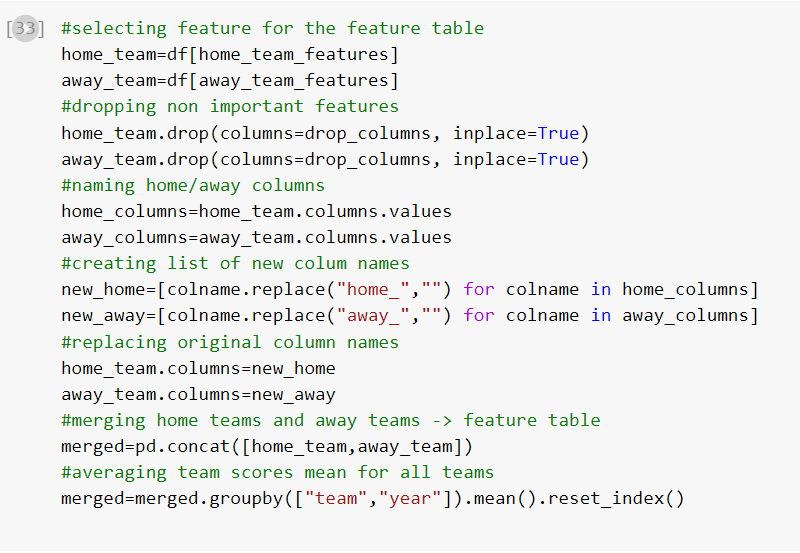
The previous preprocessing function will include these two codes to deal with such “outliers” in the dataset. This is an own subjective decision supported by data.

# **3.2 –** **Feature engineering**

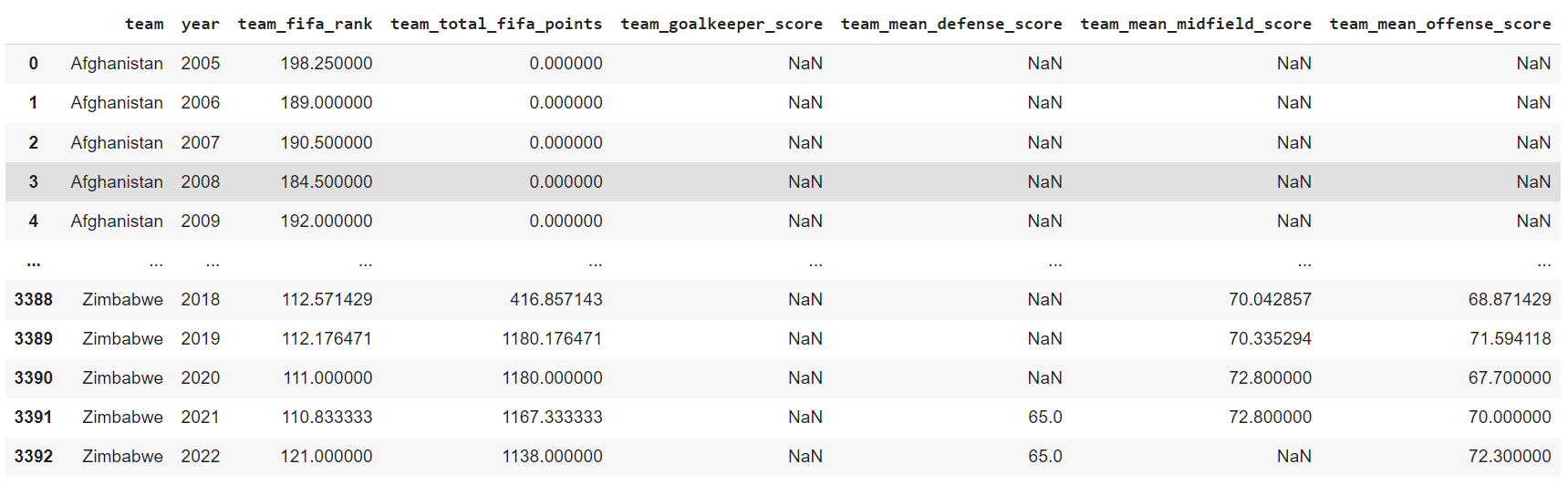
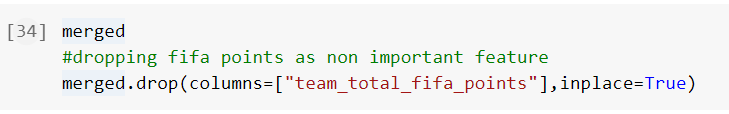
Creating effective attributes for machine learning algorithms, also know as a feature engineering, is one of the most crucial jobs in data analysis. Well-formed attributes that gather the specific features of the data, and enhances the model’s predictive power and its explain ability. Feature engineering is regarded as one of the most challenging stages of data analysis due to the fact that useful features typically reflect an in-depth knowledge of the data’s business fields and its analysis job (Heaton, 2016).

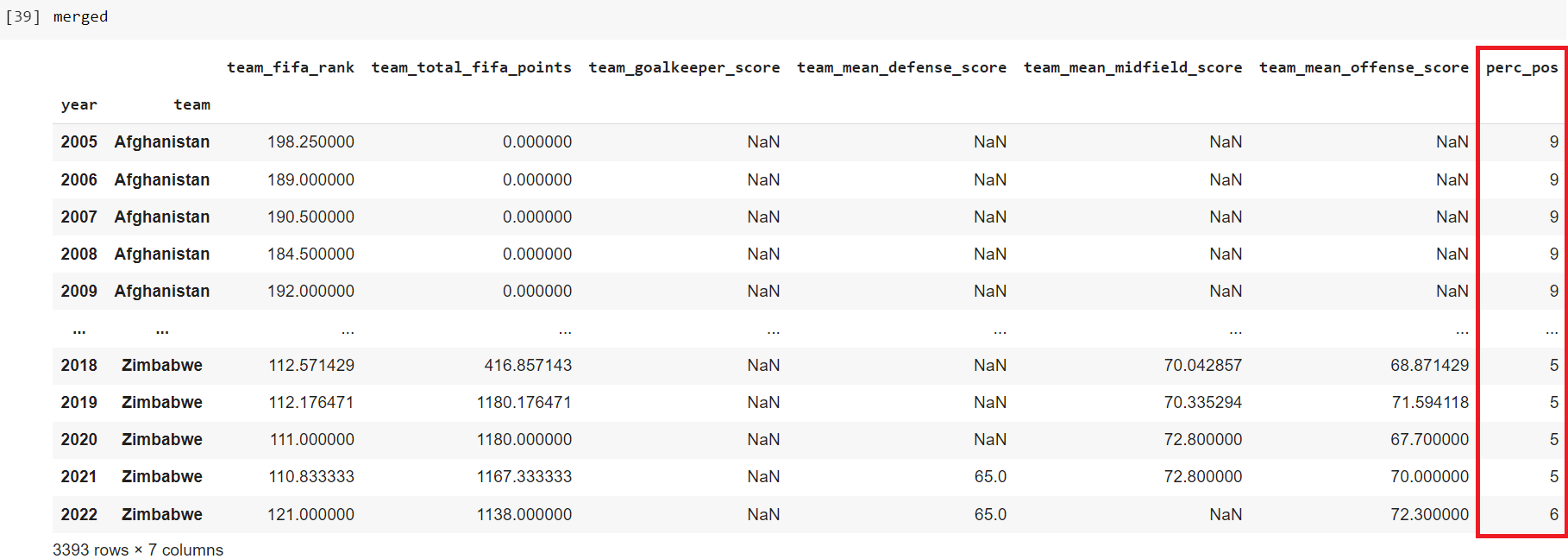
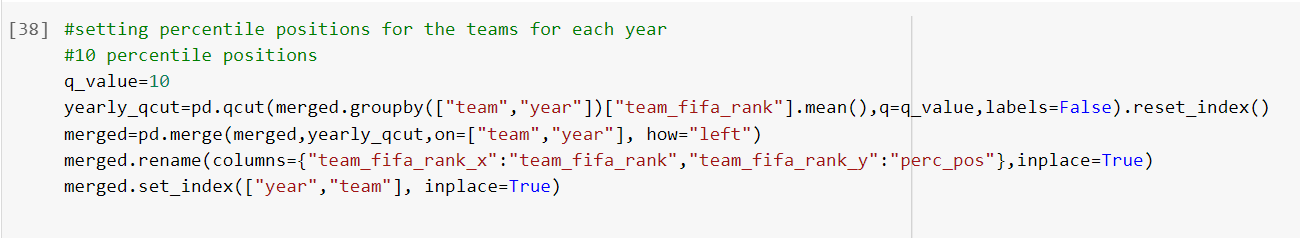
The important and non important features must be selected initially.

Then a merged table is created in the following way.



One more feature is deleted from the merge table because it will not be used in the model.

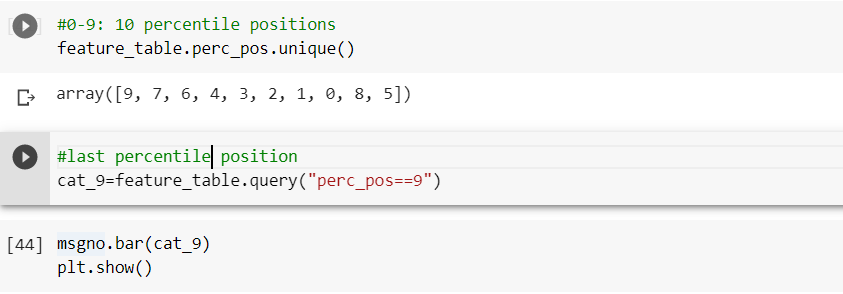
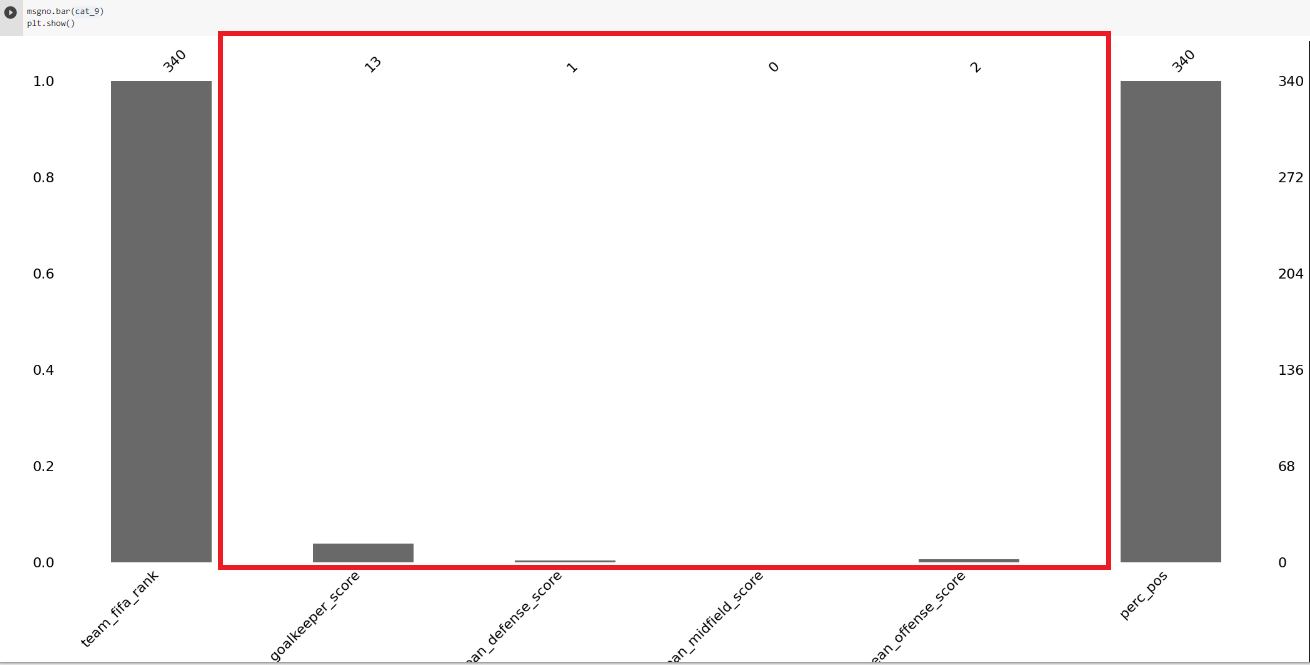
 Then10 percentile positions are created for each year by team\_fifa\_rank in order to impute missing data.



As previously the preprocessed function, create\_feature\_table function contains the feature engineering steps up to this point.

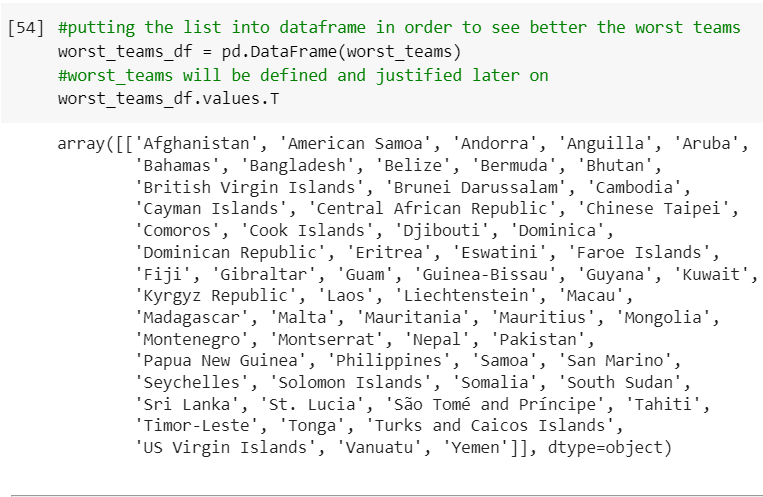


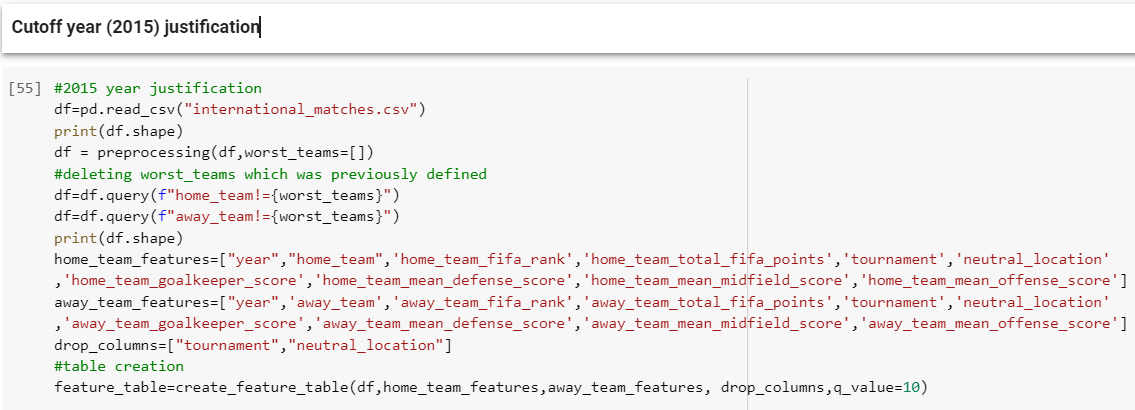
 It can be seen clearly that for the worst teams category (percentile 9) there is almost no data, averages based on years can not be determined so as next step it will be checked which of these teams is not among the ones qualified for the Qatar WC and delete them.

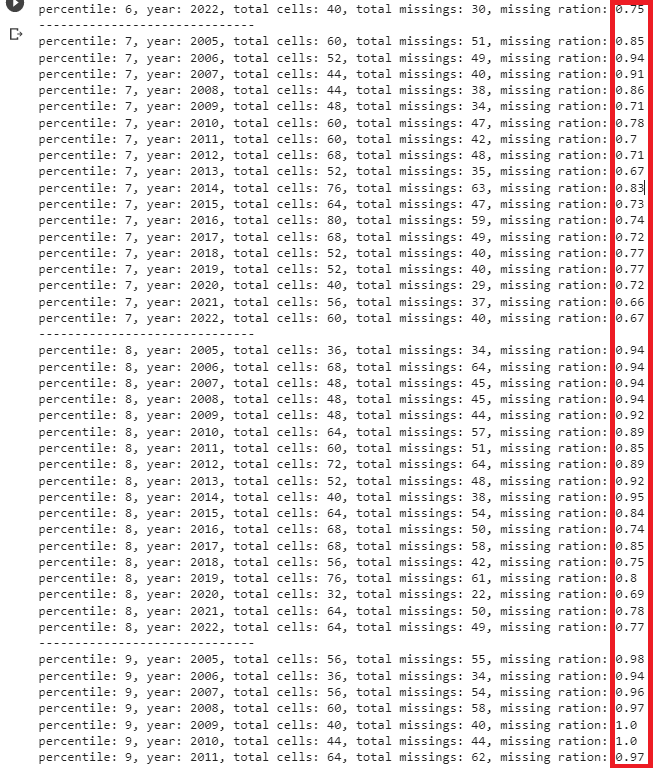


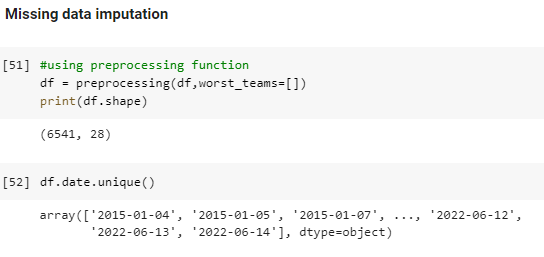
Not a single team participates in WC 2022, therefore, all worst\_teams can be excluded.

These worst\_teams will be excluded.





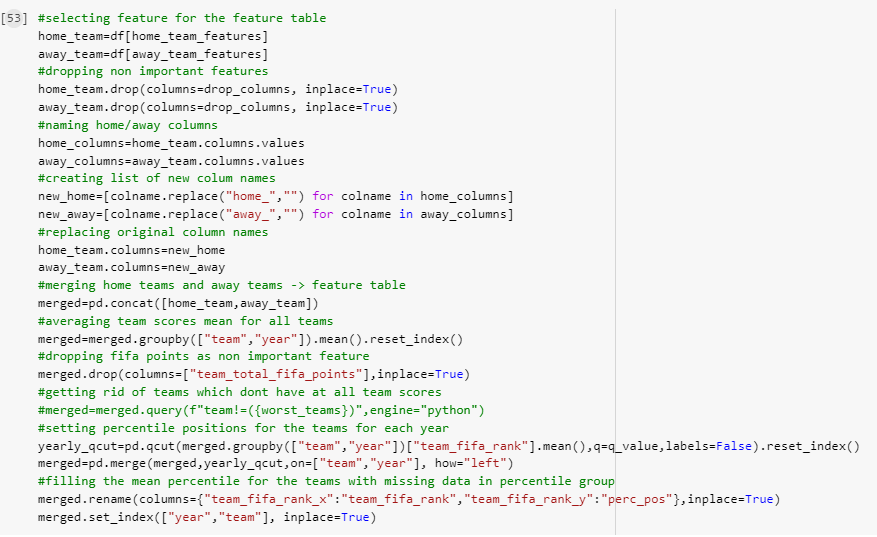


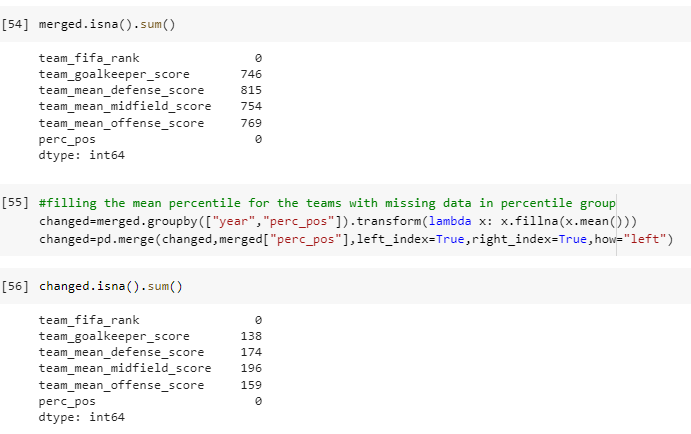


Based on the analysis above it can see that there are many missing data for some higher percentile categories (7th and 8th). There are multiple years before 2015 when the 90+% of the data is missing, therefore a cutoff before 2015 is placed. We need to make a tradoff between getting rid of too many data or mean imputation of too many data.

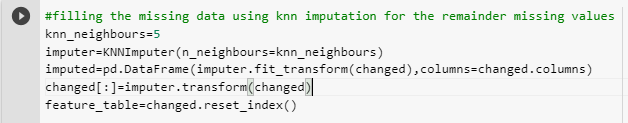
Cutoff year of 2015 is used for preprocessing which was justified before.



 The merged table is created in the following way.

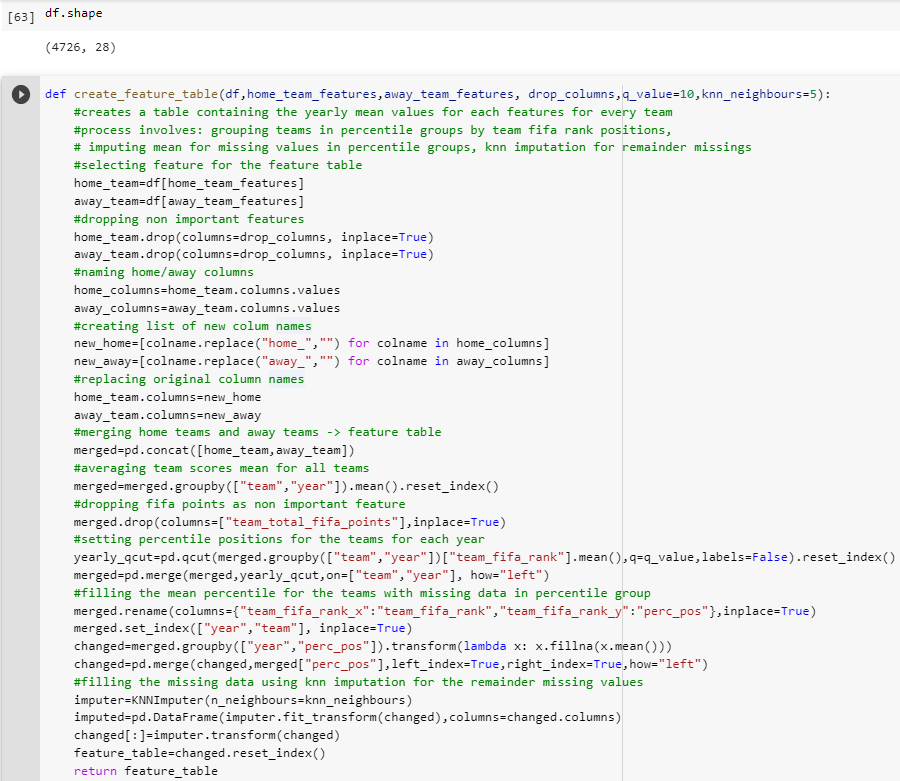
 The merged table has many missing values. Firstly, mean perc\_pos imputation is applied to fill up those values.

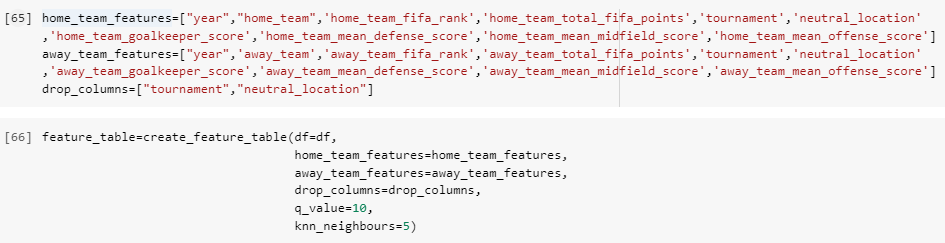
However, it is not possible to impute all values with only mean. The rest of the missing values are imputed by KNN imputation. KNN is an algorithm for matching a point in a space with multiple dimensions which its k nearest neigbours (here 5 neighbours). The underlying idea the use of KNN for missing data is that a point’s value can be approximated by the values of the points closest to it, given additional factors.

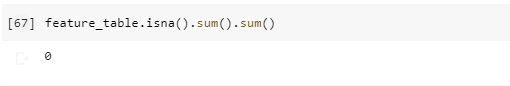
 All values are imputed here.

 Put everything together and using only preprocessing and create\_table functions to impute data which is exact same procedure as the above detailed steps. Using functions are more practical and easier to get the same result.

****

The feature table was created by the following function to contain all the necessary features (home and away teams).

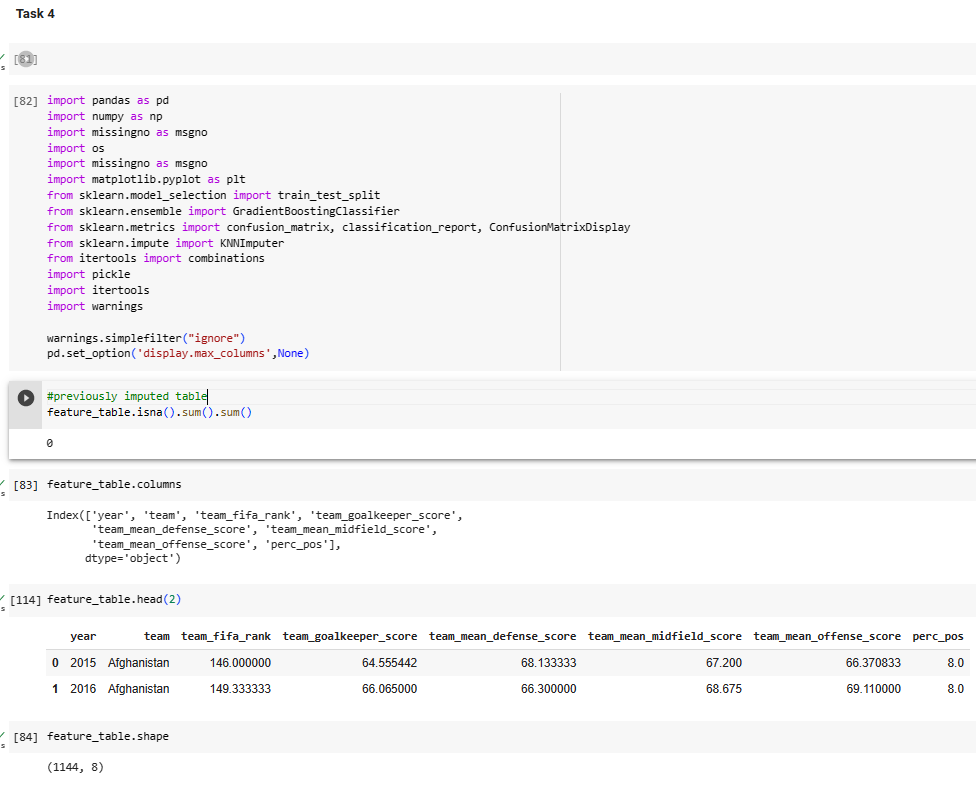
 Missing values are imputed using the two defined functions above because the final feature\_table does not contain any missing values anymore.

**

# **Task 4 – Model selection and training**

# **4.1 – Scored and received goal models**

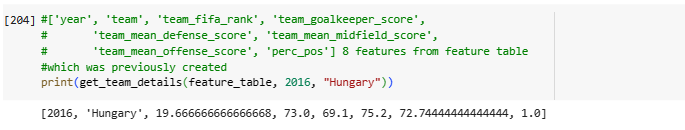
Two models are used to predict the scored and received goals for a game. This can be used for the group stage. However, in the knockout stage can be draw and in order to decide those game an additional shoout-out model will be used which will decide the draw games which team can go to the next stage.



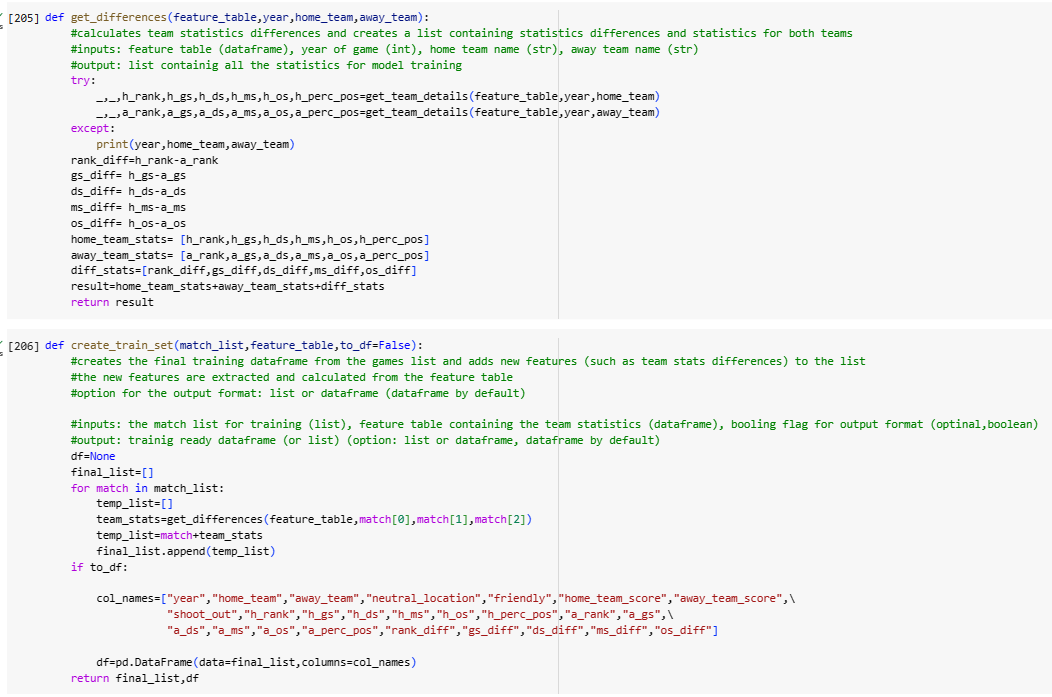
The different models are created to predict just the scored and received goals for a football game. The following functions are used to get match details and create the train dataset. Firstly, the match details are retrieved from the original dataset.



Then, team details can be retrieved from feature table using the above stated function. The team features can be found by using this function.

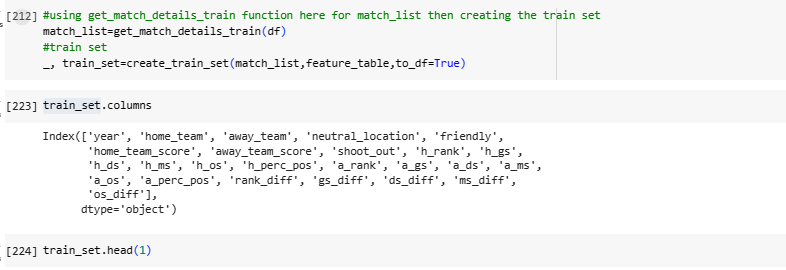


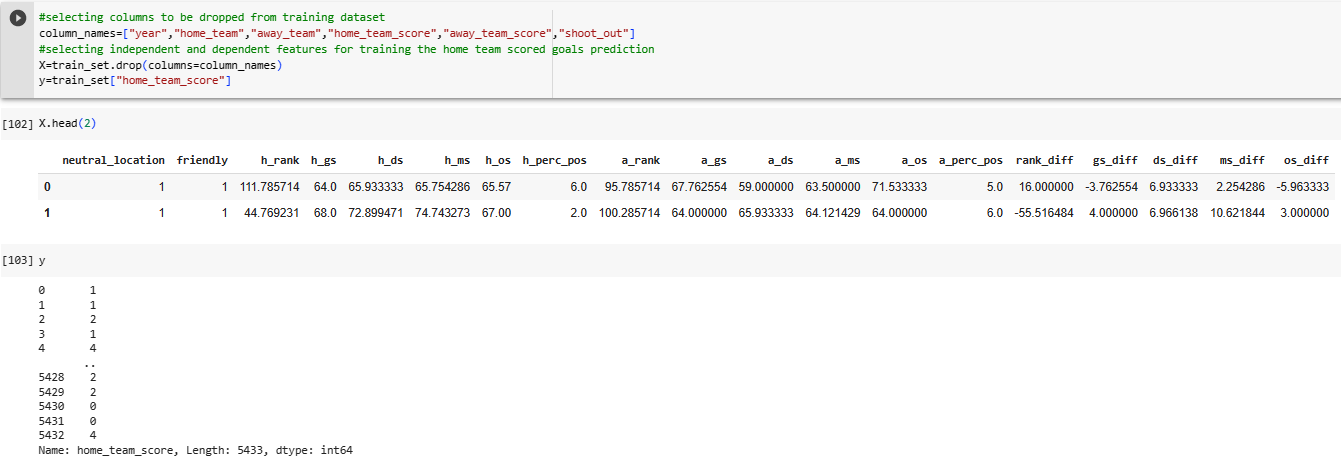
After that, train set can be created by the following functions.



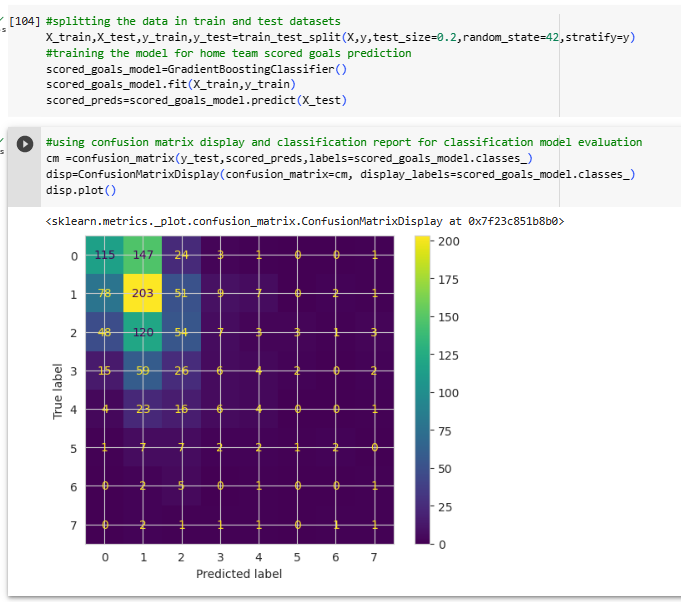


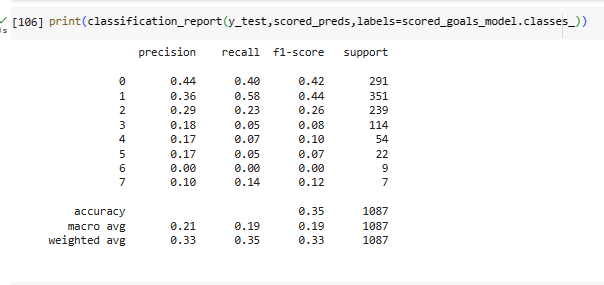
The train set can be derived from the original dataset using the functions above. The train\_set means a model ready dataset which will be divided into train and test dataset.



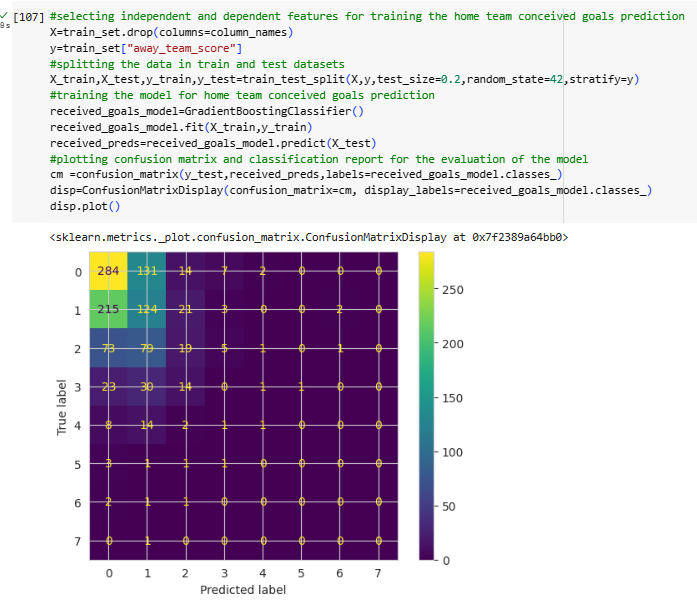


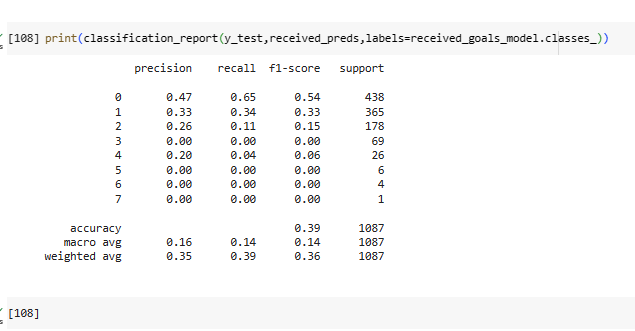
GradientBoosting Classifier is used because it is known to be effective for tabular data. The data can be divided in the following using stratify to get the similar data allocation in test and train data.





The model has an accuracy score of 0.35. In order to predict 0 and 1 and 2 and 3 goals, the f1-scores are higher. This is because to have more the 3 goals in word cup game can be extreme and there is not enough data for the model to train. Accordingly, the whole accuracy is lower. Maybe those values after more than 3 goals can be taken out to get more accurate score.

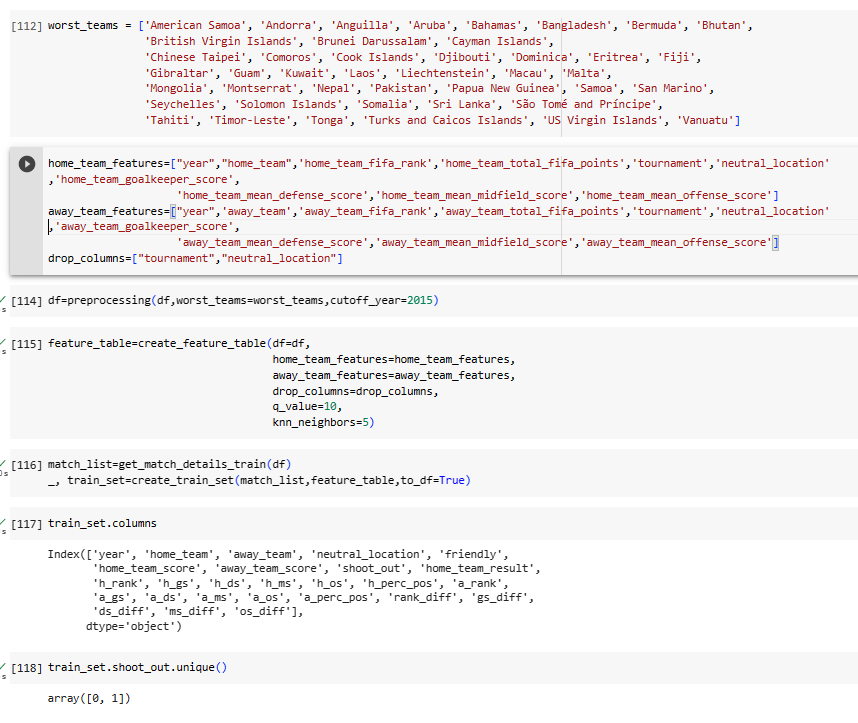
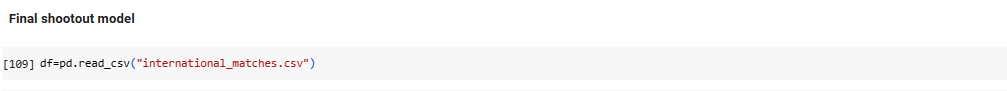




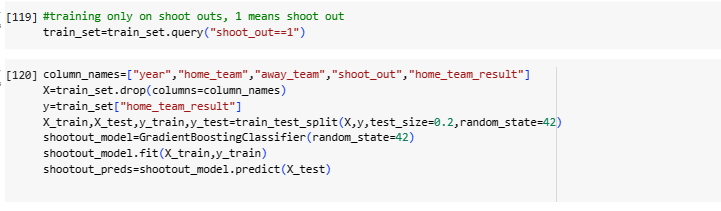
The model has an accuracy score of 0.39. In order to predict 0 and 1 and 2 and 3 goals, the f1-scores are higher. This is because to have more the 3 goals in word cup game can be extreme and there is not enough data for the model to train. Accordingly, the whole accuracy is lower. Maybe those values after more than 3 goals can be taken out to get more accurate score.

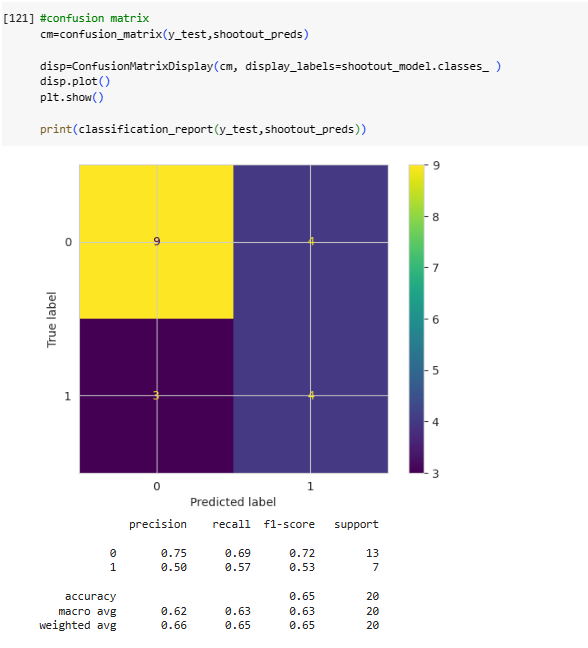
# **4.2 – Final shoot-out model**

The final model will predict a winning percentage for the selected 5 teams. All the previously created functions are used to build up the whole model.



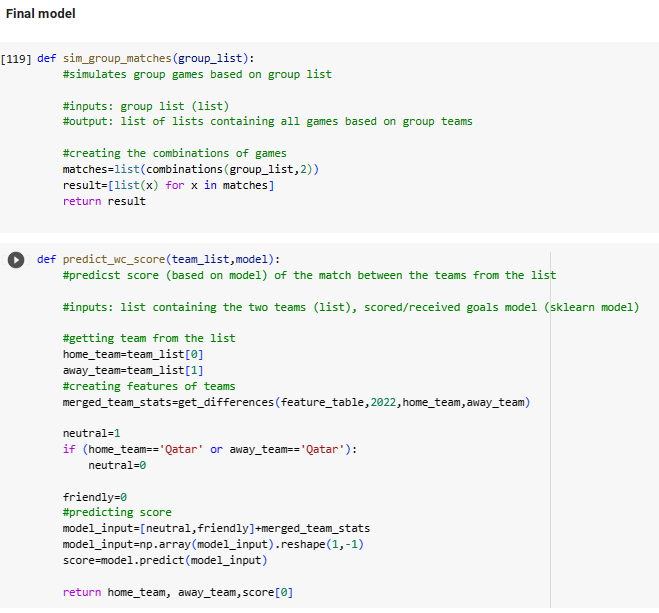
This model is only for the games with shootouts. Only for those games where there are shoot outs after the game.

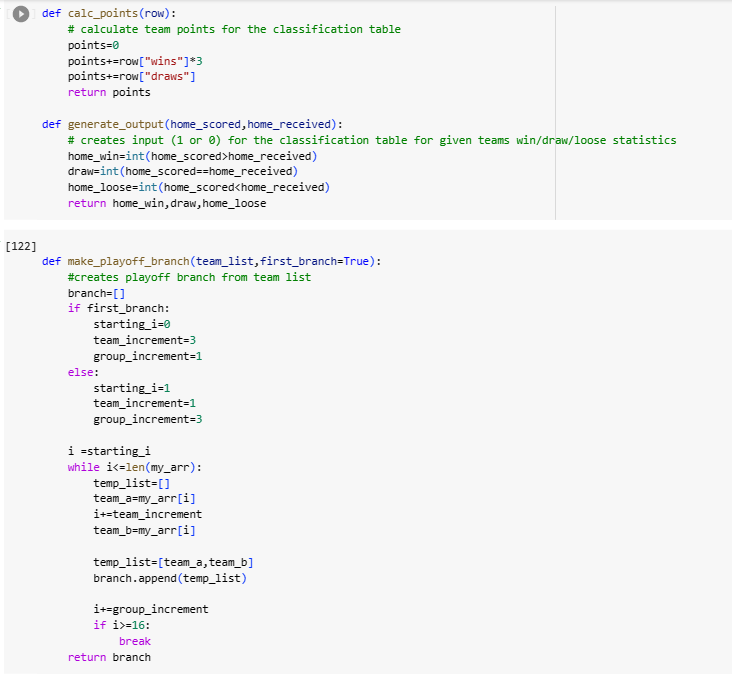




The final shoout model has a high accuracy rate.

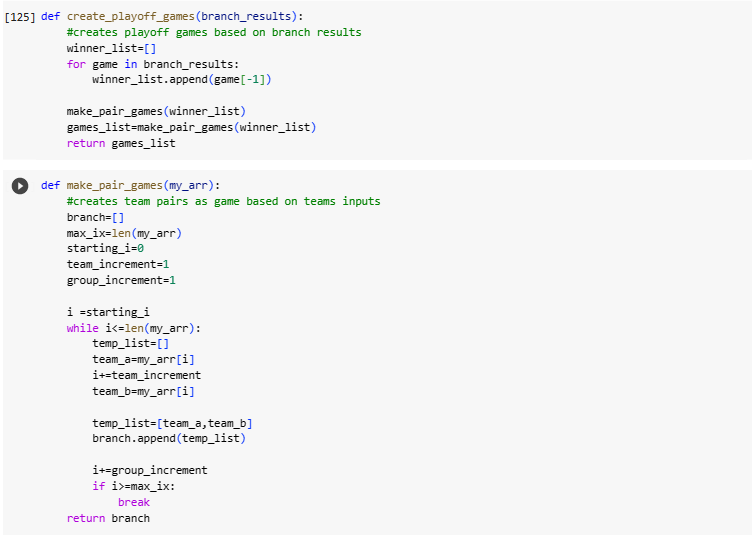
The following functions are created to create the final model.



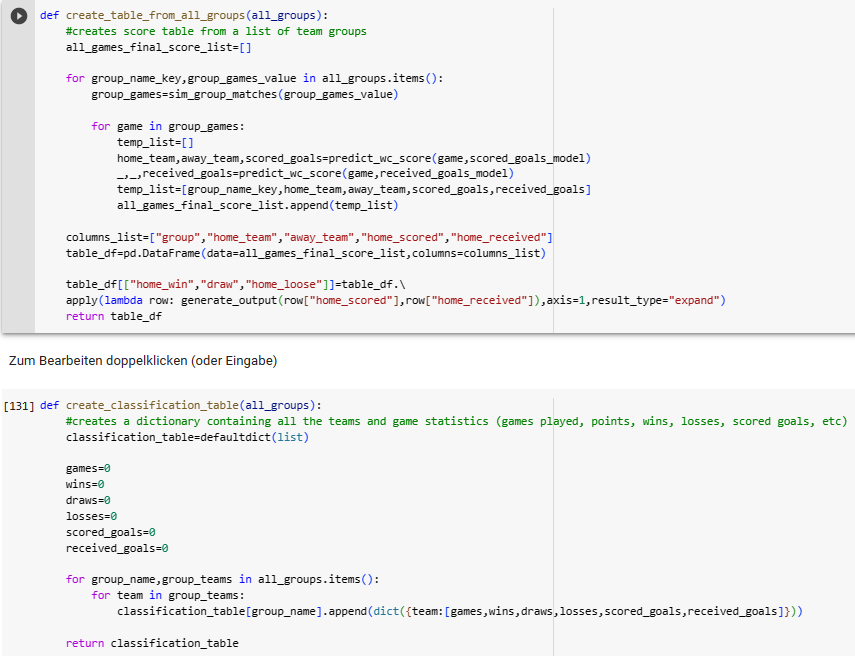


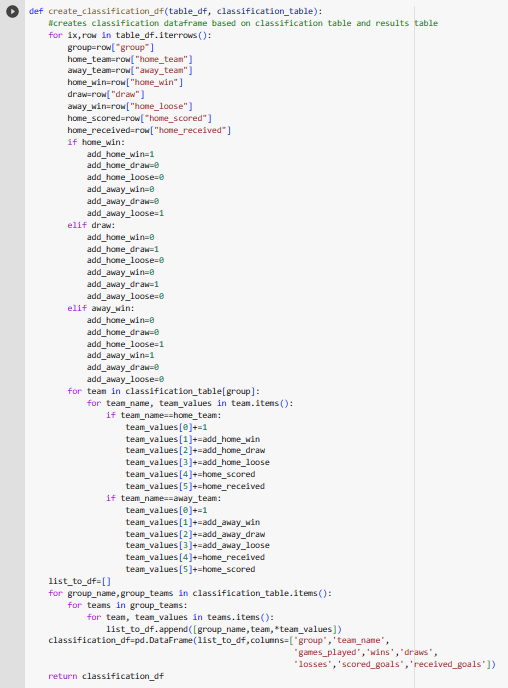
If the game result is draw after the group stage, the shoot\_out model is used.

The get\_branch\_results function create the scored and received goals for a branch.

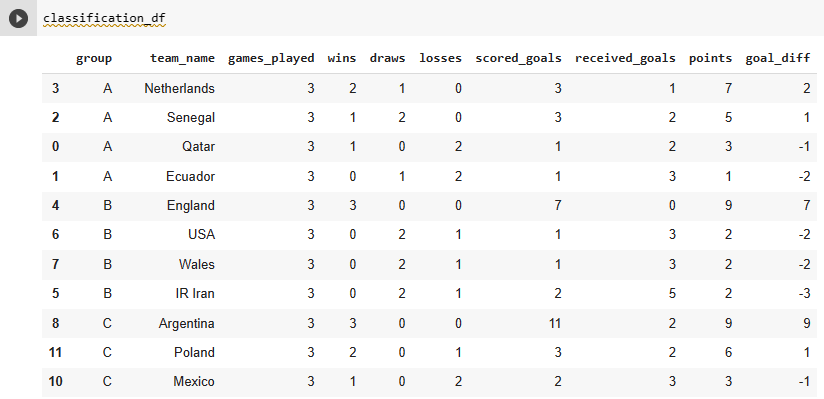
The above stated two functions create the matching for teams.

The original world cup groups are the above stated groups which will be used to simulate the world cup. The above stated function create the a table with all possible groups.





The classification table contains the following features.



After putting everything together, the tournament can be simulated one time using the above detailed functions to do that.

****

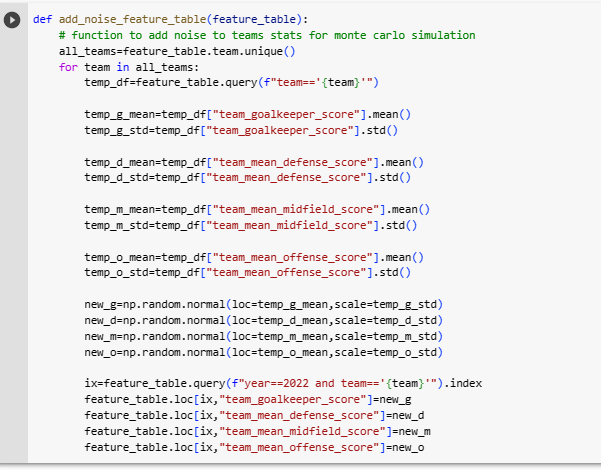
# **4.3 – Monte carlo simulation**

Monte Carlo Simulations (MCS) are algorithms utilised to model the likelihood of various results in an event that is difficult to forecast because of the random factors’ involvement. A football game can be simulated thousands of times applying precisely the same circumstances on the field in order to determine how the game might have progressed considering the conditions at the start.

In contrast to conventional forecasting models, MCS forecasts a variety of results according to a predicted range of values for input versus a fixed set of values for input. MCS builds a simulation of possible results by applying a distribution of probabilities, such as uniform and normal, for any parameter whose value is impacted by unpredictability. It next repeatedly recalculates the outcomes, appalling an another collection of random numbers within maximum and minimum points each time. This can be repeated many times to generate a many different probable results. MCS can be performed to determine the possibilities of different results of a football match, and therefore, make more accurate forecasts. To provide insight into the simulation, a range of parameters could be added to the simulation like injuries, the particular form of key players, the recent form squad etc (www.timeform.com., n.d.).

The conditions can be different for the teams. As a result of this, some noises can be added to the model to simulate MCS which is based on this concept. Accordingly, the simulation can be done many times by using the modified features for the teams. Players can be injured and have a different forms all the time. Accordingly match results can change according to the simulation.





# The simulation can be done many times depending on the runs choice. MCS is run here 5 times (runs=5) to simulate the world cup with the same teams but with different features (adding up different noises to them).

However, there are teams for the simulation. 5 teams are selected to get their winning percentage based on the simulation 5 teams. The simulation can be run any times which is the main concept of MCS.

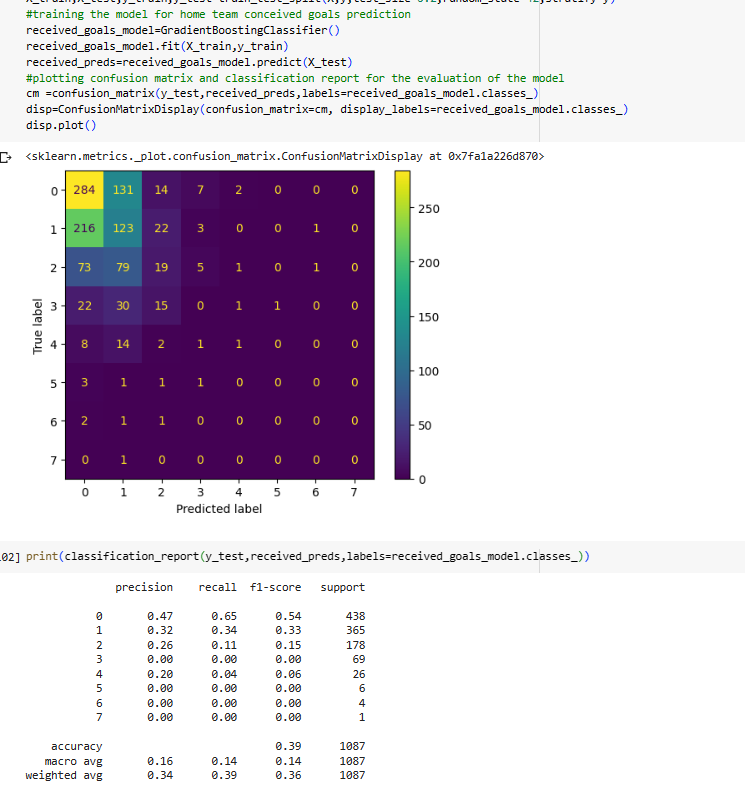
# 

# In this case, the following result occurred according to our simulation. France has the highest possibility to win the world cup after running 5 times the simulation.

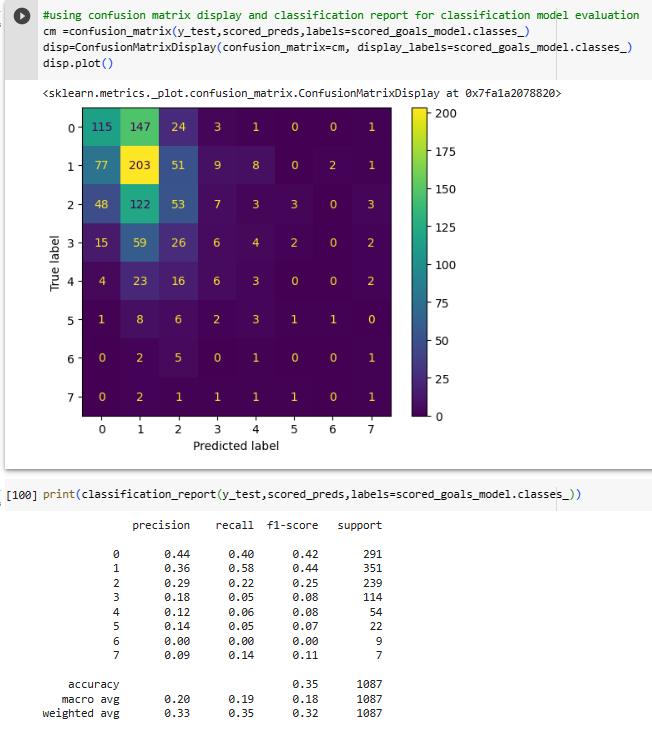
# **Task 5 – Model evaluation and visualization**

Hyperparameter Tuning

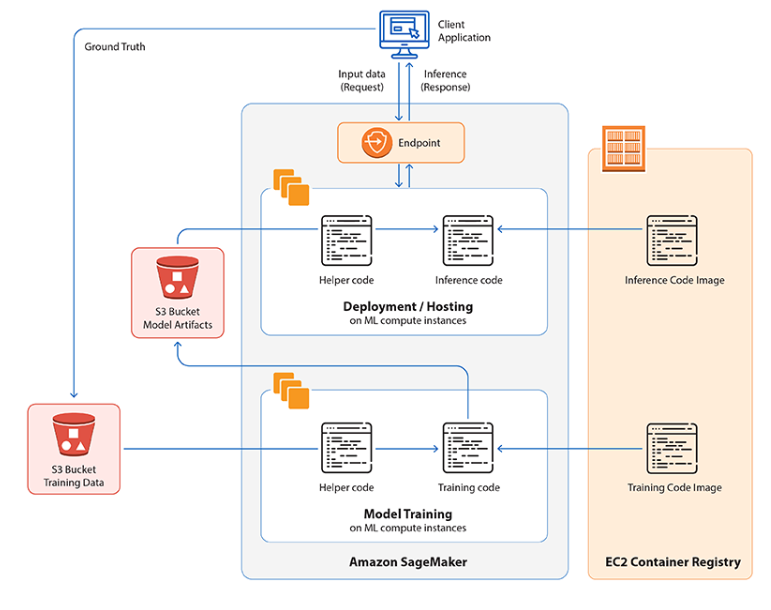
The method of selecting the most suitable set of hyperparameters for a model used for machine learning is known as a hyperparameter optimisation. It is essential stage in the development of a model because the choice of hyperparameters can have an important effect on the efficacy of the model as a whole. Before the learning process starts, hyperparameters are set to control the behaviour of the model over the training. The objective of hyperparameter optimisation is to determine the most effective combination of hyperparameter parameters. Tuning of hyperparameters is essential because changing hyperparameter parameters can lead to significantly varying outcomes. Choosing incorrect hyperparameters can cause under or overfitting in the model. Effective hyperparameters can greatly improve the model result, leading to improved accuracy rate and more robust model.



It can be seen that the model accuracy could be better, after looking at f1-score. Hyperparameter can be used to improve the model performance.

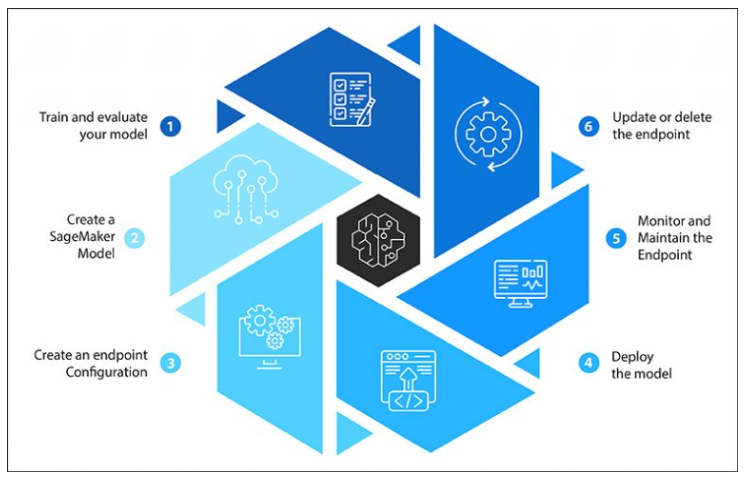


# **Task 6 – Model deployment**



source:

Machine learning has evolved into a crucial component of many enterprises currently. Despite this, deploying algorithms based on machine learning can be a difficult process, particularly when it comes to developing and handling the algorithms. Amazon SageMaker enters the process at this point. SageMaker facilitates the machine learning modelling method by offering a complete platfrom for the construction, training, and deployment.



Source:

**Model deployment applying SageMaker:**

1. step: Training and evaluating the model is the initial phase in deployment. SageMaker offers a Jupyter notebook for use in the development and evaluation. Training and evaluation code can be created in this location. After that, the model’s elements can be stored to Amazon S3.

2. step: After the first step, the subsequent step is to construct a SageMaker model. SageMaker algorithms are containers built with Docker that include machine learning algorithm. Indicate the exact spot of the algorithm elements in Amazon S3, the domain name of Docker container, and the code necessary for importing the algorithm to create the algorithm.

3. step: Following the creation of a SageMaker algorithm, an endpoint setup must be created. Endpoint setup is a configuration that describes the quantity and variety of entities essential for hosting the endpoint. Endpoints can be configured by applying the SageMaker interface or SageMaker API.

4. step: Model deployment can be done by constructing an endpoint based on the endpoint setup was generated earlier. SageMaker offers a completely controlled system to host the endpoint, containing automated scaling and load sharing. AWS deployment enables enterprises to benefit from cloud-based tools to deploy machine learning algorithms at scale.

5. step: Model monitoring can be performed by utilising Amazon CloudWatch. It offers metrics including latency and call rate. This data can also be utilised for optimising the efficacy of endpoint. Numerous companies have already implemented Amazon SageMaker to conduct A/B testing and compare the efficacy of various algorithms. Monitoring algorithms is vital to ensure the accurate performance continuously. SageMaker contains algorithm controlling capabilities, enabling users to discover and address possible problems before they result in major problems

6. step: Developing a new endpoint setup and the deployment of a new model will enable to update the endpoint. Furthermore, the endpoint can be deactivated by using SageMaker interface or SageMaker API (Softwebsolutions, 2023).

**Data ethics**, and **privacy** are only some of the ethical factors that must be seriously taken into account the responsible development and deployment of artificial intelligence (AI). Data ethics is an additional essential component of the development and deployment of AI. Data is the engine that drives AI, and it is important that data gathering and utilisation comply to ethical and legal standards. In order to prevent continuing social biases, businesses must ensure that the data utilised for training AI algorithms are relevant and objective. In addition, people must have the ability to manage the data, and their privacy must be secured all over the entire AI development and deployment procedure. Privacy is a basic human right that must be secured during the development and deployment processes. It is important to ensure that AI systems gather and utilise personal data in an ethical and appropriate manner, as they frequently gather and store enormous amounts of such data (Yaqoob, 2023).

## Appendix: Google Colab file attachment

https://colab.research.google.com/drive/1Bij8uwTqmmizSnYvac2L1Gg7PzWRxiOl?usp=sharing

## References

Adam, A. (2016). Generalised Linear Model for Football Matches Prediction. In *MLSA@ PKDD/ECML*.

Agarwal, V. (2015). Reserch on data preprocessing and categorization technique for smartphone review analysis. *International Journal of Computer Applications*, *131*(4), 30-36.

Athish, V.P., Rajeswari, D. and SS, S.N., (2023). Football Prediction System using Gaussian Naïve Bayes Algorithm. In *2023 Second International Conference on Electronics and Renewable Systems (ICEARS)* (pp. 1640-1643). IEEE.

Beal, R., Middleton, S.E., Norman, T.J. and Ramchurn, S.D., (2021). Combining machine learning and human experts to predict match outcomes in football: A baseline model. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 35, No. 17, pp. 15447-15451).

Challita, S., Zalila, F., Gourdin, C. and Merle, P., (2018). A precise model for google cloud platform. In *2018 IEEE international conference on cloud engineering (IC2E)* (pp. 177-183). IEEE.

Constantinou, A. C. (2019). Dolores: a model that predicts football match outcomes from all over the world. *Machine Learning*,*108*(1), 49-75.

Danisik, N., Lacko, P., and Farkas, M. (2018). Football match prediction using players attributes. In *2018 World Symposium on Digital Intelligence for Systems and Machines (DISA)* (pp. 201-206). IEEE.

Guptha, A., Murali, H. and Subbulakshmi, T., (2021). May. A Comparative Analysis of Security Services in Major Cloud Service Providers. In *2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS)* (pp. 129-136). IEEE.

Heaton, J., (2016). March. An empirical analysis of feature engineering for predictive modeling. In *SoutheastCon 2016* (pp. 1-6). IEEE.

KINALIOĞLU, İ.H. and Coşkun, K.U.Ş., (2020). Prediction of UEFA champions league elimination Rounds winners using machine learning algorithms. *Cumhuriyet Science Journal*, *41*(4), pp.951-967.

Lasek, J. (2016). EURO 2016 Predictions Using Team Rating Systems. In *MLSA@ PKDD/ECML*.

Noshiri, N., Khorramfar, M. and Halabi, T., (2021). Machine Learning-as-a-Service Performance Evaluation on Multi-class Datasets. In *2021 IEEE International Conference on Smart Internet of Things (SmartIoT)* (pp. 332-336). IEEE.

Pliuhin, V., Pan, M., Yesina, V. and Sukhonos, M., (2018). Using azure maching learning cloud technology for electric machines optimization. In *2018 International Scientific-Practical Conference Problems of Infocommunications. Science and Technology (PIC S&T)*(pp. 55-58). IEEE.

Rahman, M. A. (2020). A deep learning framework for football match prediction. *SN Applied Sciences*, *2*(2), 165.

Softwebsolutions (2023). *A guide to deploying ML models using Amazon SageMaker*. [online] Available at:<https://www.softwebsolutions.com/resources/deploy-machine-learning-models-using-amazon-sagemaker.html> [Accessed 17 May 2023].

Stübinger, J., Mangold, B., and Knoll, J. (2019). Machine learning in football betting: Prediction of match results based on player characteristics. *Applied Sciences*, *10*(1), 46.

Surbiryala, J. and Rong, C., (2019). Cloud computing: History and overview. In *2019 IEEE Cloud Summit* (pp. 1-7). IEEE.

Tesser, R.K., Marques, A. and Borin, E., (2021). Selecting efficient VM types to train deep learning models on Amazon SageMaker. In *2021 International Symposium on Computer Architecture and High Performance Computing Workshops (SBAC-PADW)* (pp. 20-27). IEEE.

Tiwari, E., Sardar, P., and Jain, S. (2020). Football match result prediction using neural networks and deep learning. In *2020 8th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO)*(pp. 229-231). IEEE.

[www.timeform.com](http://www.timeform.com/). (n.d.). *What Is The Monte Carlo Method? | Sports Betting | Timeform*. [online] Available at:<https://www.timeform.com/betting/advanced/monte-carlo-model>.

Yaqoob, T. (2023). *Ethical considerations in AI development and deployment*. [online] Cointelegraph. Available at:<https://cointelegraph.com/explained/ethical-considerations-in-ai-development-and-deployment>.

Zhang, Q., Xu, H., Wei, L., and Zhou, L. (2019). Prediction of football match results based on model fusion. In *Proceedings of the 2019 3rd International Conference on Innovation in Artificial Intelligence* (pp. 198-202).