PROJECT – PYTHON FOR DATA ANALYSIS – DIA1

NATACHA BATMANABANE
NICOLAS BERLIOZ
LANA BONHOMME


STARCRAFT
REMASTERED

GAME PRESENTATION

With three races, four modes, and endless ways to play, StarCraft II is the ultimate real-time strategy experience

1

BUILD YOUR BASE

2

TRAIN YOUR ARMY

3

ATTACK

OUR GOAL

PREDICT YOUR LEAGUE

Bronze | Silver | Gold | Platinum | Diamond | Master | Grand Master

How to determine a player's League based on their statistics ?

HOW ?

WITH YOUR IN-GAME STATS

HOW DOES IT WORK ?

1. Our dataset
2. Data pre-processing
3. Data visualization
4. Modeling
5. Application
6. Conclusion

OUR DATASET

Our dataset contains 20 columns and 3395 rows.

The columns are :

GameID : Unique ID number for each player
LeagueIndex
Age
HoursPerWeek
TotalHours
APM : Actions per minute
SelectByHotkeys : Number of unit or building selections made using hotkeys per timestamp
AssignToHotkeys : Number of units or buildings assigned to hotkeys per timestamp
UniqueHotkeys : Number of unique hotkeys used per timestamp
MinimapAttacks : Number of attack actions on minimap per timestamp
MinimapRightClicks : Number of right-clicks on minimap per timestamp
NumberOfPACs : Number of PAC per timestamp
GapBetweenPACs : Mean duration (in ms) between PACs
ActionLatency : Mean latency from the onset of a PACs to their first action (ms)
ActionsInPAC : Mean number of actions within each PAC
TotalMapExplored : Number of game grid viewed per timestamp
WorkersMade : Number of workers trained per timestamp
UniqueUnitsMade : Unique unites made per timestamp
ComplexUnitsMade : Number of complex units trained pe timestamp
ComplexAbilitiesUsed : Abilities requiring specific targeting instructions used per timestamp

Dataset link : https://archive.ics.uci.edu/ml/datasets/SkillCraft1+Master+Table+Dataset#

* PAC : Perception/Action Cycle

DATA PRE-PROCESSING

Age	HoursPerWeek	TotalHours
?	?	?
?	?	?
?	?	?
?	?	?
?	?	?

No NaN value in out dataset but presence of « ? »
Replacement with « NaN »

```
Nombre de valeurs nulles :

Age             55
HoursPerWeek    56
TotalHours      57
```

Age	HoursPerWeek	TotalHours
NaN	NaN	NaN
NaN	NaN	NaN
NaN	NaN	NaN
NaN	NaN	NaN
NaN	NaN	NaN

We can also observe that the NaN values are only on the Professional League (= 8)

LeagueIndex
5
5
4
3
3

Creation of a new column ‘LeagueIndex’
From the ‘League’ column
To simplify the use of the data

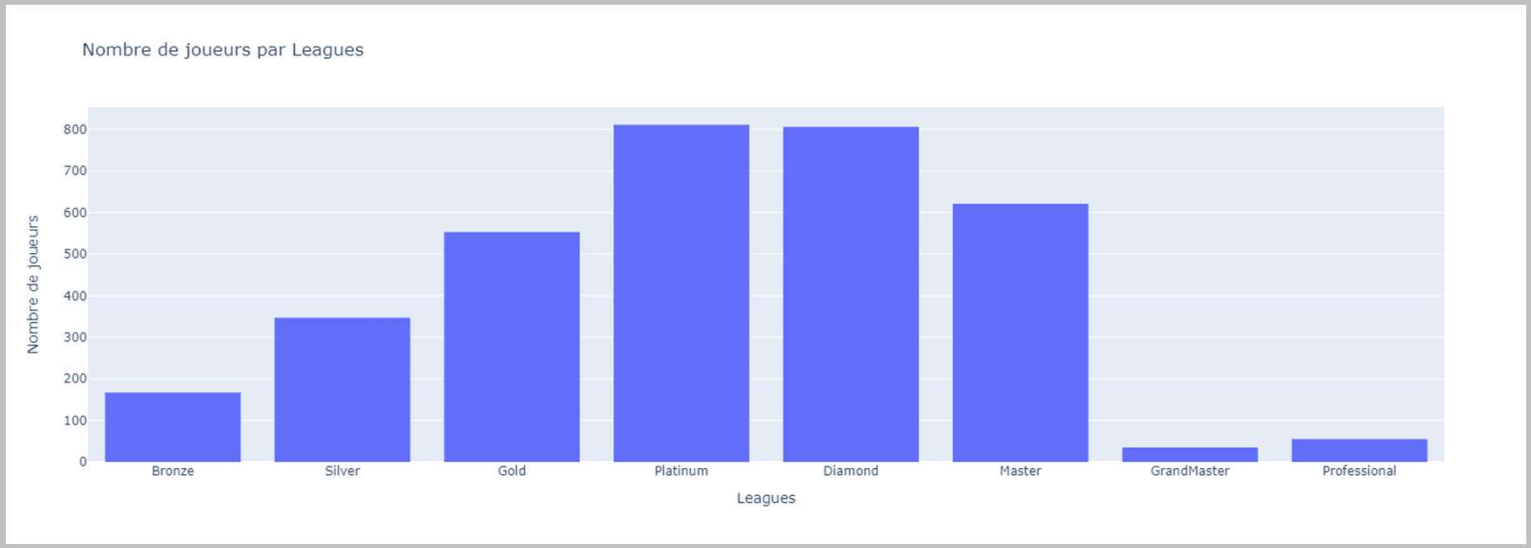
```
1:"Bronze"
2:"Silver"
3:"Gold"
4:"Platinum"
5:"Diamond"
6:"Master"
7:"GrandMaster"
8:"Professional"
```

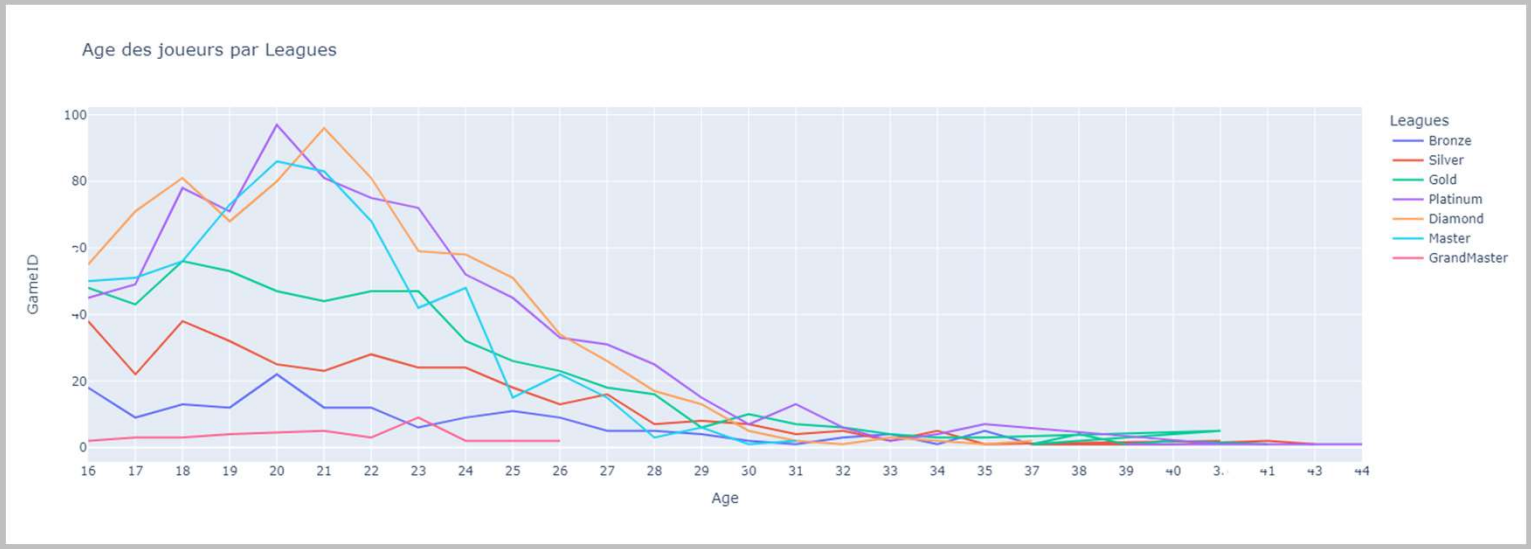
```
Age             object
HoursPerWeek    object
TotalHours      object
```

Conversion of the columns ‘Age’, ‘HoursPerWeek’ and ‘TotalHours’
From Object to Int

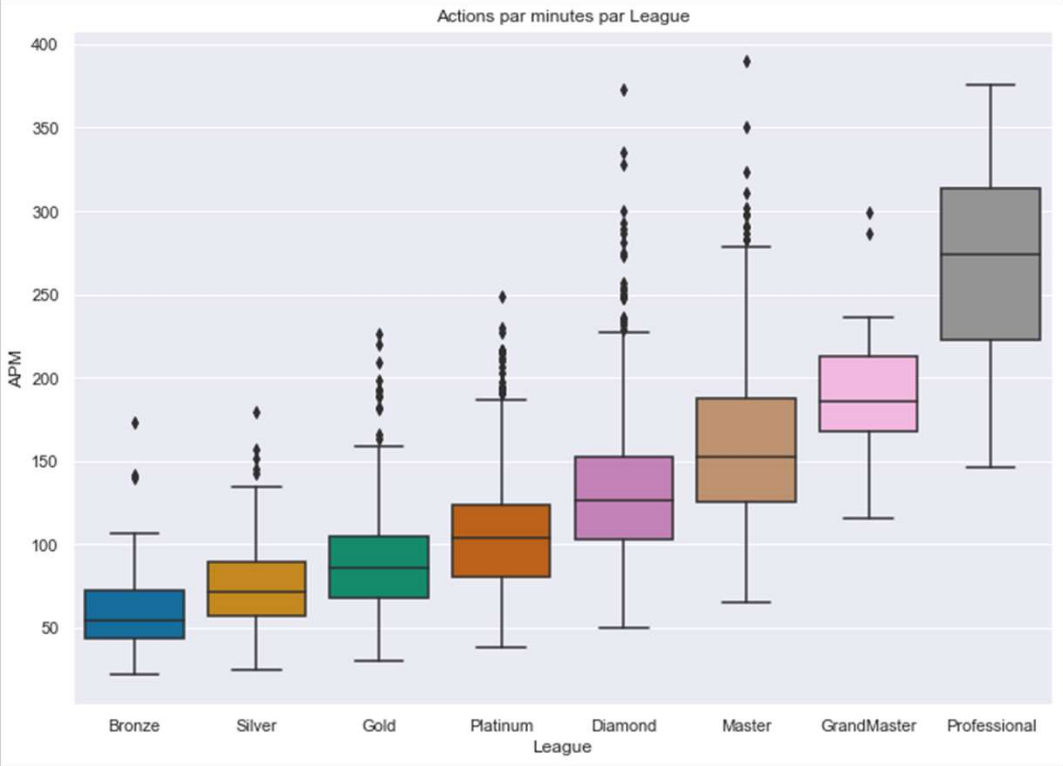
```
Age             int32
HoursPerWeek    int32
TotalHours      int32
```

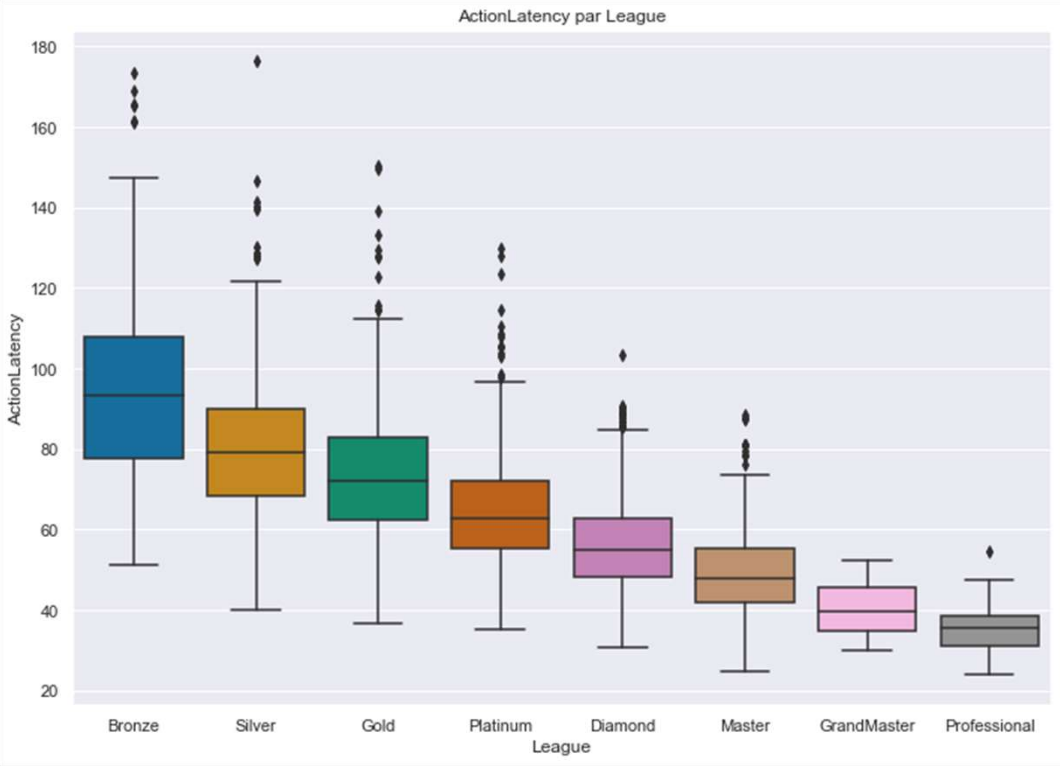
DATA VISUALIZATION



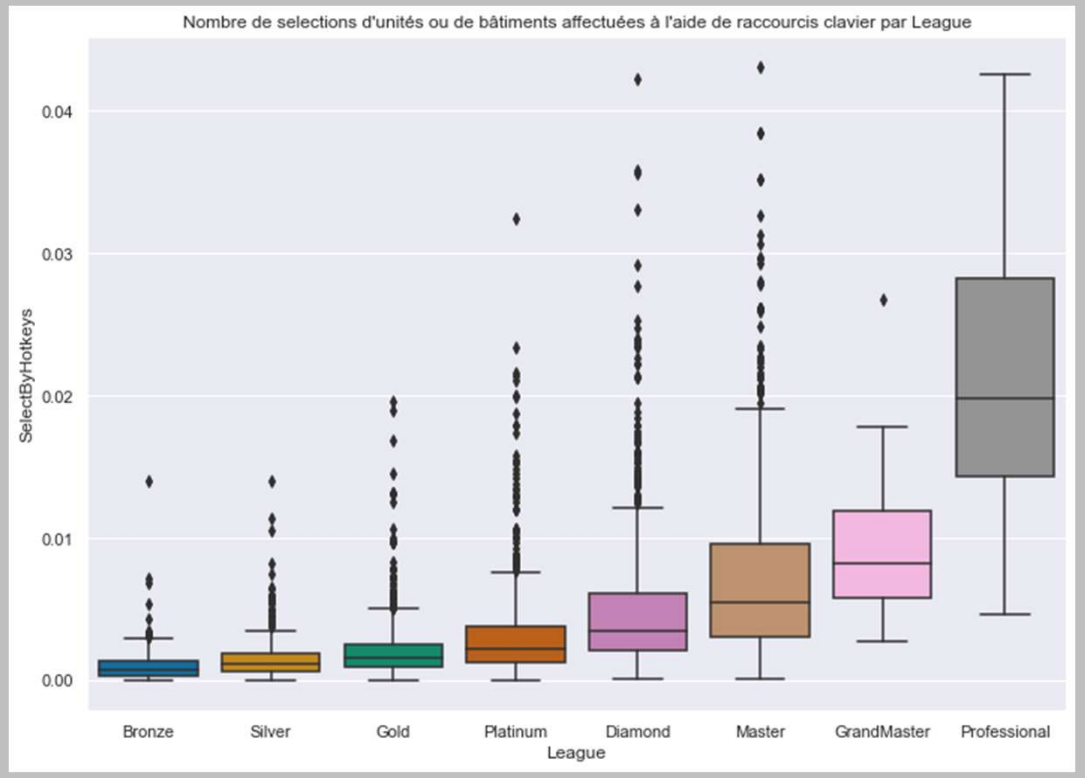


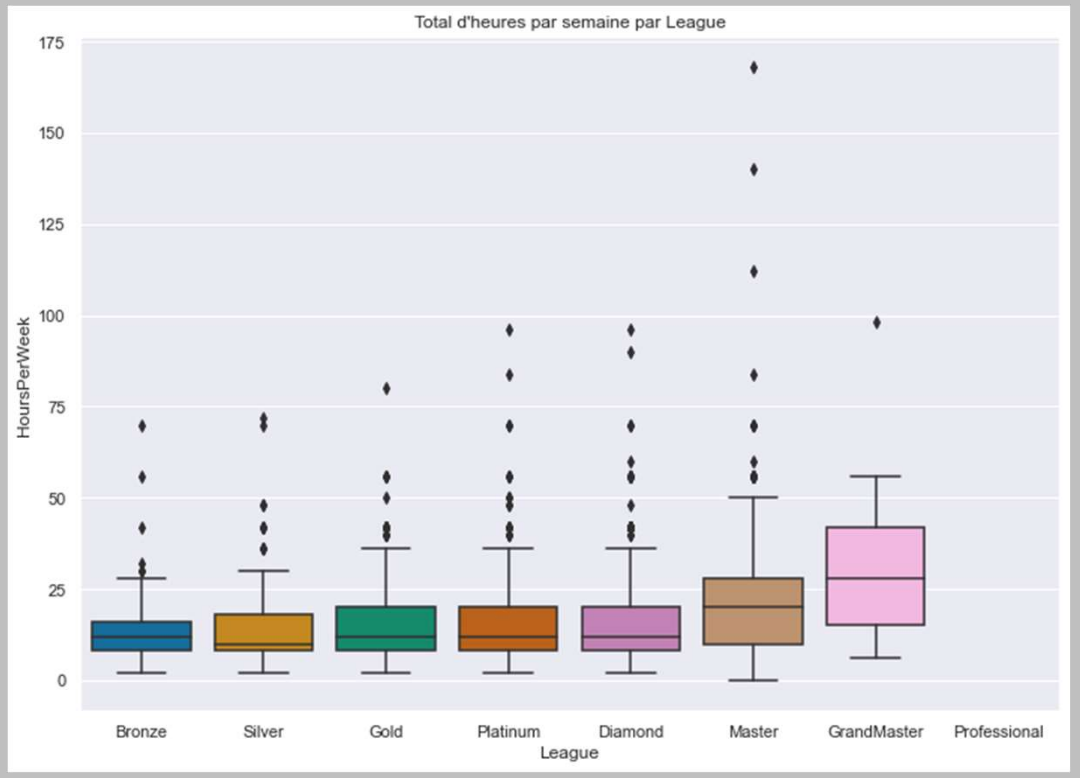
DATA VISUALIZATION



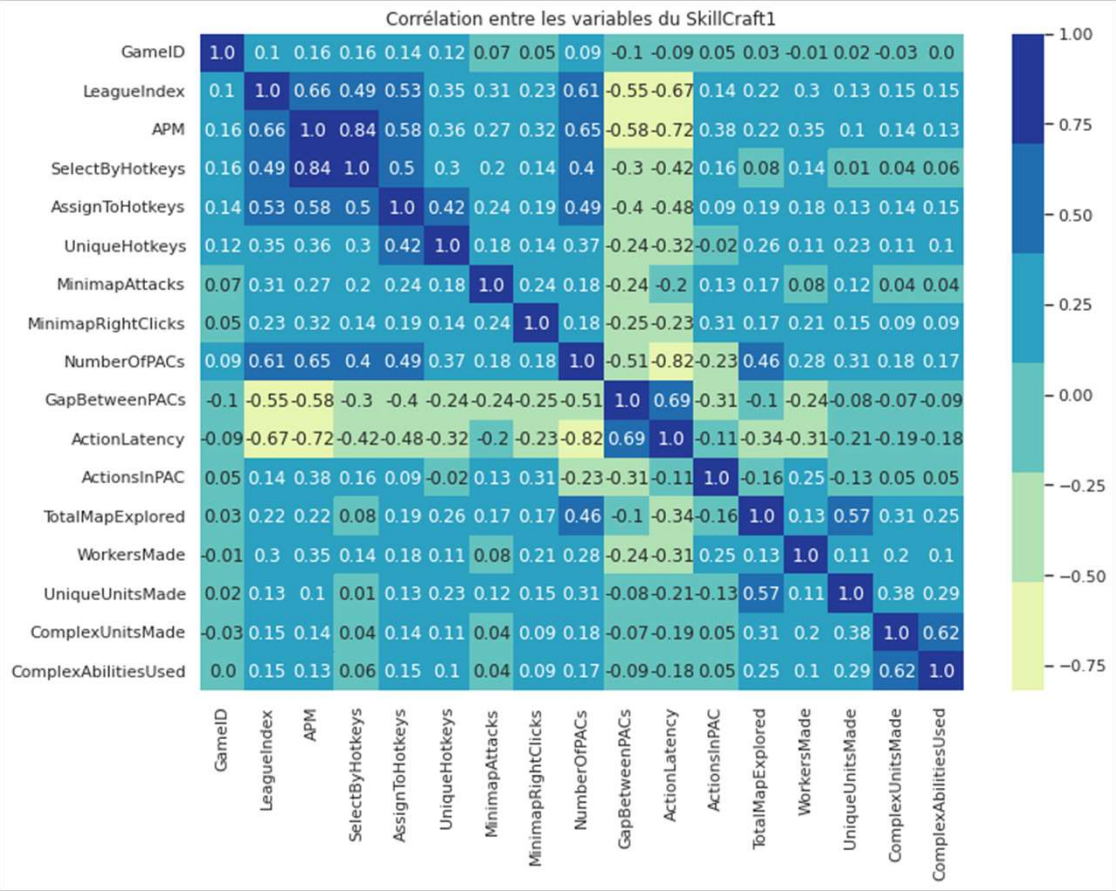


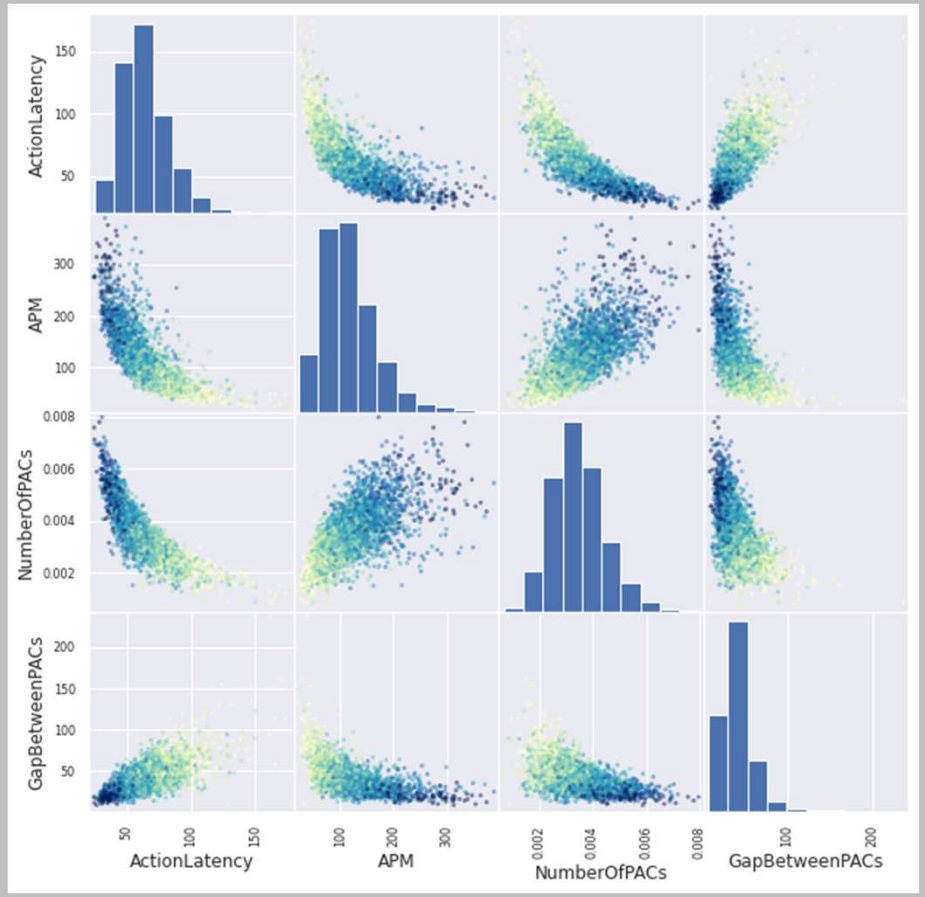
DATA VISUALIZATION





DATA VISUALIZATION





MODELING

1

Removal of rows with "NaN" values, i.e. the Professional League.
We end up with a dataset of 3338 rows (originally 3395).
Then we converted the columns 'Age', 'HoursPerWeek', 'TotalHours' to integer

```
SC_sansNull = SkillCraft
SC_sansNull.dropna(inplace = True)
```

```
SC_sansNull.shape
```

```
(3338, 21)
```

```
Age             int32
HoursPerWeek    int32
TotalHours      int32
```

2

Separation of training data for our model

```
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size = 0.2)
```

We have 2670 rows for the X and 668 for the Y

MODELING

3 The different models

MULTIPLE LINEAR REGRESSION

Model with 11 columns :

- HoursPerWeek
- APM
- SelectByHotkeys
- AssignToHotkeys
- UniqueHotkeys
- MinimapAttacks
- NumberOfPACs
- GapBetweenPACs
- ActionLatency
- ActionsInPAC
- WorkersMade

RESULT

```
Accuracy =  0.4041916167664671
Confusion matrix =
 [[  2  10  17   3   0   0   0]
 [  1   8  36  23   1   0   0]
 [  0   6  50  42   6   1   0]
 [  0   4  36 105  37   2   0]
 [  0   1   7  56  73  12   0]
 [  0   0   0  21  70  29   2]
 [  0   0   0   0   1   3   3]]
```

MODELING

3 The different models

GRADIENT BOOSTING CLASSIFIER

Model with 11 columns :

- HoursPerWeek
- APM
- SelectByHotkeys
- AssignToHotkeys
- UniqueHotkeys
- MinimapAttacks
- NumberOfPACs
- GapBetweenPACs
- ActionLatency
- ActionsInPAC
- WorkersMade

RESULT

```
Best  accuracy =  0.3962546816479401
Confusion matrix :
 [[13  9  4  1  0  0  0]
 [22 17 18  7  3  0  0]
 [ 6 17 26 22 12  1  0]
 [ 2 19 42 73 39 14  0]
 [ 0  2 12 56 70 59  0]
 [ 0  0  4  8 21 57  9]
 [ 0  0  0  0  1  2  0]]
```

MODELING

3 The different models

RANDOM FOREST CLASSIFIER
+
SMOTE

Model with 11 columns :

- HoursPerWeek
- APM
- SelectByHotkeys
- AssignToHotkeys
- UniqueHotkeys
- MinimapAttacks
- NumberOfPACs
- GapBetweenPACs
- ActionLatency
- ActionsInPAC
- WorkersMade

SMOTE

```
sm = SMOTE(k_neighbors = 3 ,random_state=42)
X_res, Y_res = sm.fit_resample(X, Y)
```

4	811	5	811
5	804	4	811
6	621	3	811
3	553	2	811
2	347	1	811
1	167	7	811
7	35	6	811

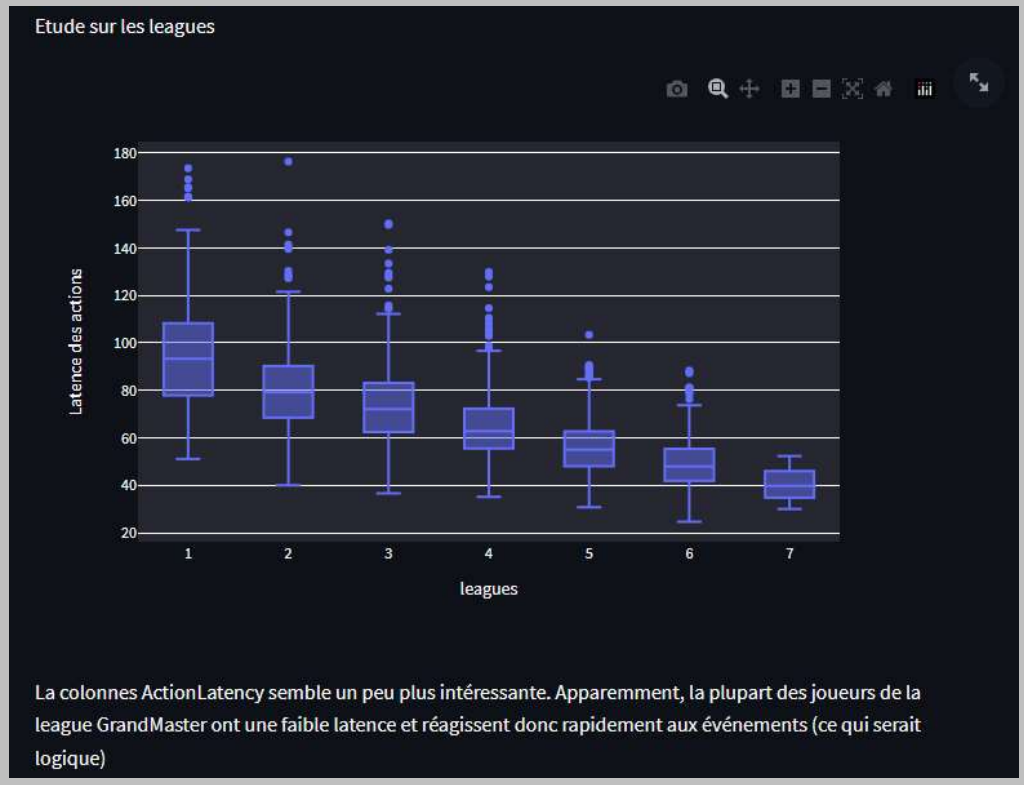
RESULT

```
Random Forest :
Best  accuracy =  0.6315821229700062
Best parameters :  {'max_depth': 50, 'n_estimators': 350}
Confusion matrix :
 [[  0   0   0   0   0   0   0]
 [  0   0   0   0   0   0   0]
 [ 24  17  15  14   6   0   0]
 [ 14  39  55  71  97  82  90]
 [  0   0   0   0   0   0   0]
 [138  89  78  96  77  67  67]
 [  0   0   0   0   0   0   0]]
```

API

Usage of Streamlit framework

Display the selected element



API

Usage of Streamlit framework

Your league with your statistics

Nombre de cycles Perception/Attaque (NumberOfPACs)

0,00 − +

Intervalles des cycles Perception/Attaque (GapBetweenPACs)

45,17 − +

Temps de réaction (ActionLatency)

64,75 − +

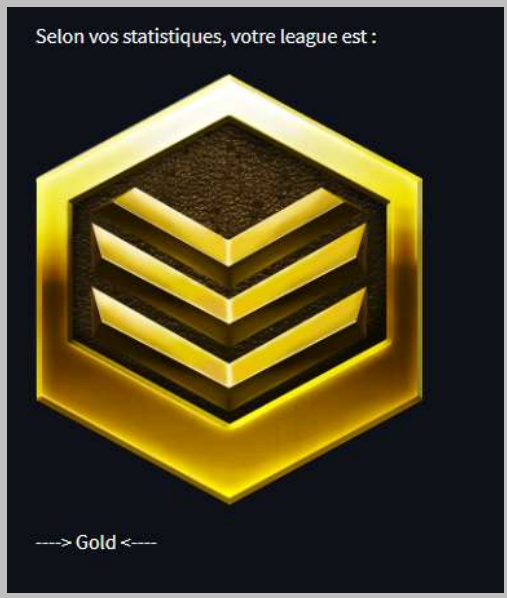
Actions durant les cycles Perception/Attaque (ActionsInPAC)

4,53 − +

Ouvriers générés (WorkersMade)

0,00 − +

Prédire



CONCLUSION

The best model is the random Forest with a score accuracy = 0.63.
This is the one we use in our API to determine the league of players.

We wanted to know how to determine the league of a player from these statistics.

The features to determine the league of a player we used are: 'HoursPerWeek', 'Actions per minute', 'SelectByHotkeys', 'AssignToHotkeys', 'UniqueHotkeys', 'MinimapAttacks', 'NumberOfPACs', 'GapBetweenPACs', 'ActionLatency', 'ActionsInPAC', 'WorkersMade'