Impact of an NBA star on his team performance

IRON HACK

Boubacar Traoré
Data Analytics 2022

CONTENTS OF **THIS PRESENTATION**

01

USE CASE

Giannis Antetokounmpo

STAR Player of the Milwaukee Bucks

DRAFTED IN 2013

2 MVP TITLE

1 CHAMPIONSHIP

Basket statistics

Points

assists

rebound

Plus_Minus

02

The DATA

THE PLAN

1. COLLECT THE DATA VIA DIRECT DOWNLOAD ON KAGGLE (2 datasets)
2. USE PANDAS TO ANALYSE THE DATA
3. MERGE THE TWO DATASETS

file	Games.csv	Games_details.csv
Content	Every NBA game from 2003 to 2022 with stats of the two teams	Every NBA game from 2003 to 2022 with stats for each player of the two teams
shape	25 796 rows, 21 columns	645 953 rows, 29 columns

DATA CLEANING

- Manage the two datasets separately
- Replace missing values (all the missing values in the first dataset where into 2003 season)
- Delete not relevant columns

```
games.GAME_STATUS_TEXT.value_counts()
```

Python

```
Final    25796
Name: GAME_STATUS_TEXT, dtype: int64
```

```
games = games.drop(['GAME_STATUS_TEXT'], axis=1)
```

Python

```
games.isna().sum().sort_values(ascending=False)
```

Python

```
null_data = games[games.isnull().any(axis=1)]
null_data
```

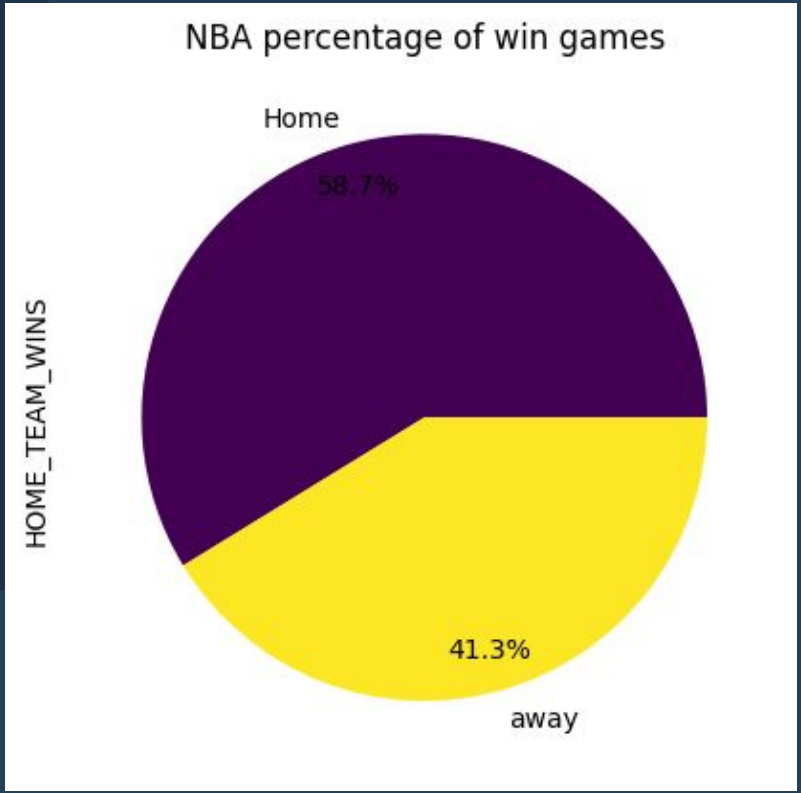
Python

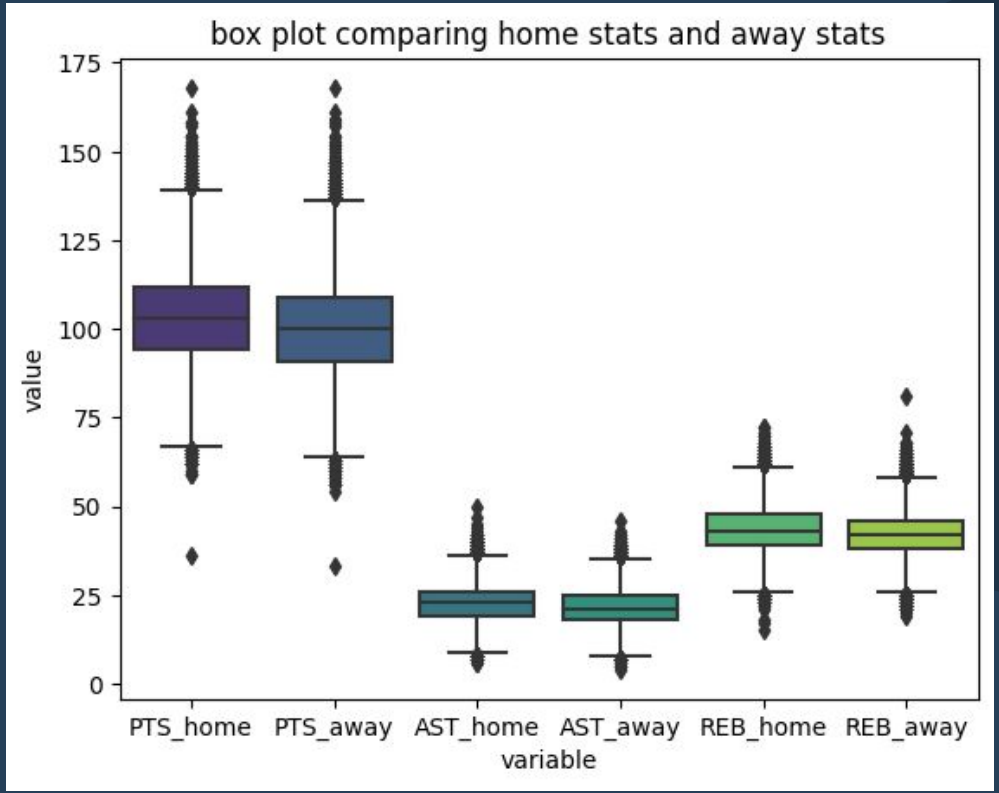
	GAME_DATE_EST	GAME_ID	HOME_TEAM_ID	VISITOR_TEAM_ID	SEASON	TEAM_ID_home	PTS_home	FG_PCT_home	FT_PCT_home	FG3_PCT_home	AST_home	REB_home	TEAM_ID_away	P
18320	2003-10-24	10300116	1610612753	1610612762	2003	1610612753	NaN	NaN	NaN	NaN	NaN	NaN	1610612762	
18321	2003-10-24	10300108	1610612737	1610612764	2003	1610612737	NaN	NaN	NaN	NaN	NaN	NaN	1610612764	
18322	2003-10-24	10300109	1610612738	1610612751	2003	1610612738	NaN	NaN	NaN	NaN	NaN	NaN	1610612751	
18323	2003-10-24	10300113	1610612759	1610612745	2003	1610612759	NaN	NaN	NaN	NaN	NaN	NaN	1610612745	
18324	2003-10-24	10300112	1610612749	1610612765	2003	1610612749	NaN	NaN	NaN	NaN	NaN	NaN	1610612765	

```
'14:08', '19:35', '27:39', '15:26', '27:45', '13:25', '12:00',
'15:10', '11:46', '19:27', '21:35', '16:57', '12:27', '4:43', '12',
'27', '19', '14', '28', '29'], dtype=object)
```

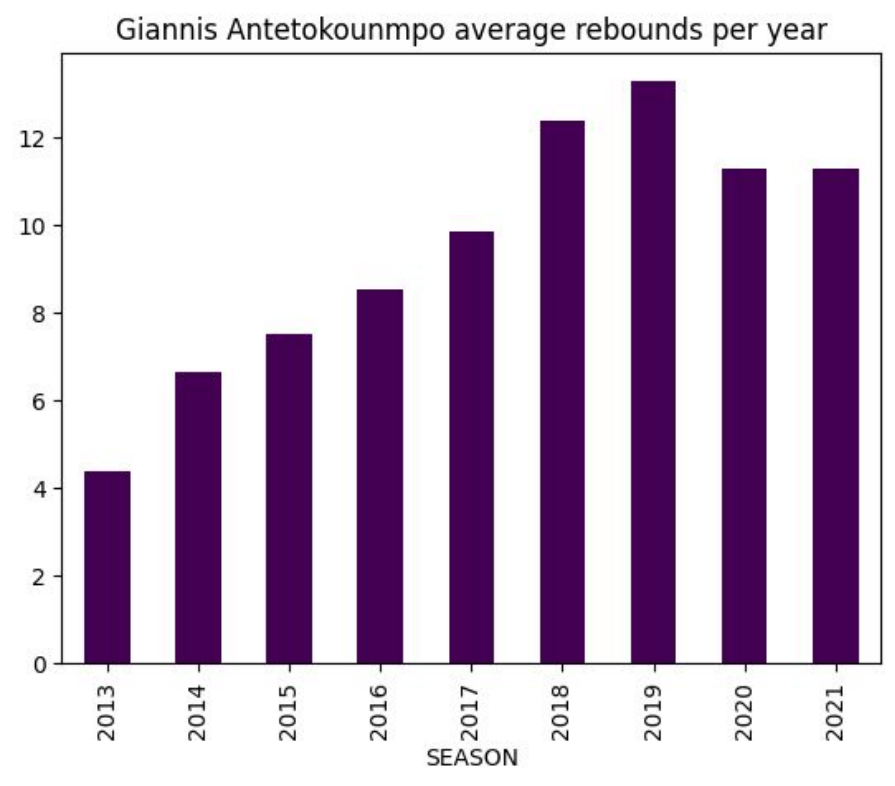
```
minute = giannis['MIN']
minute = minute.str.replace(':', '.')
minute = minute.astype('float')
minute = round(minute)
```

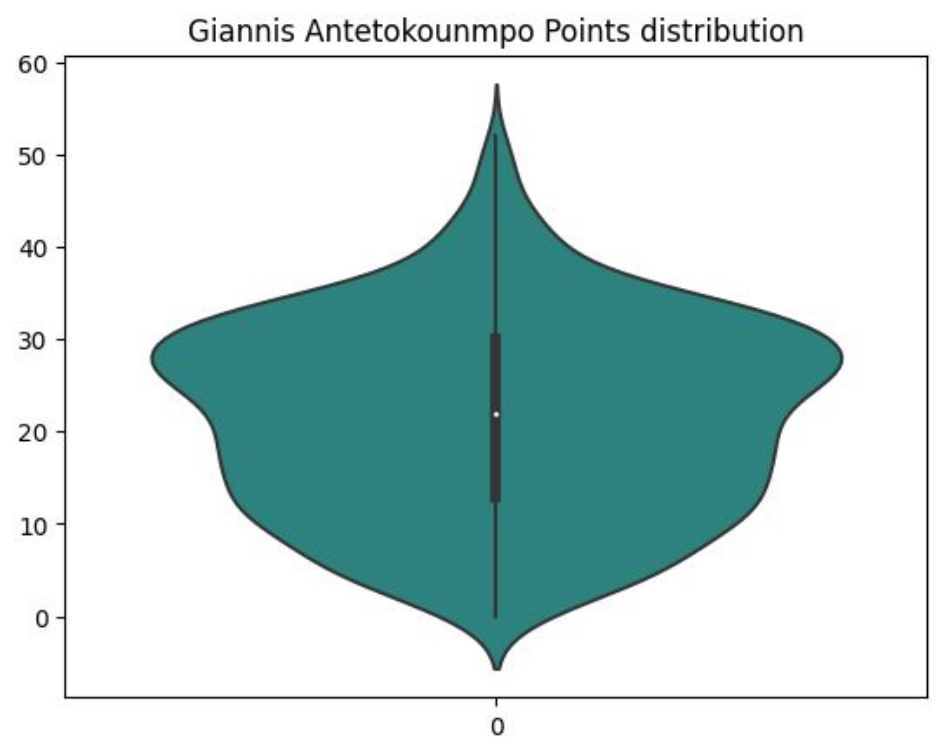
VISUALIZATION



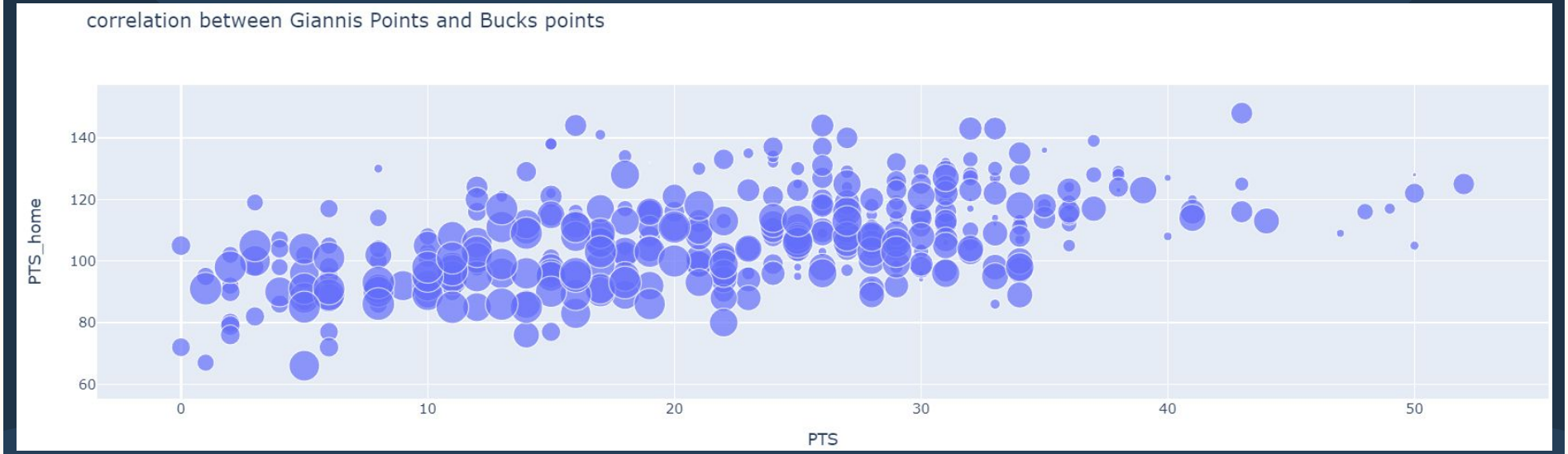


VISUALIZATION

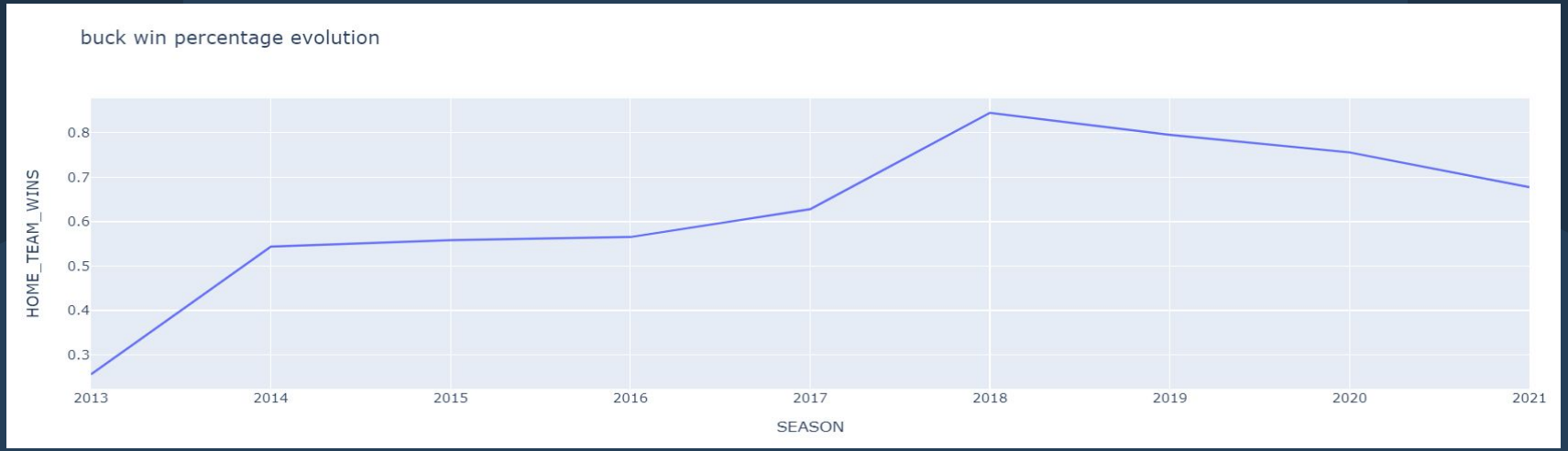




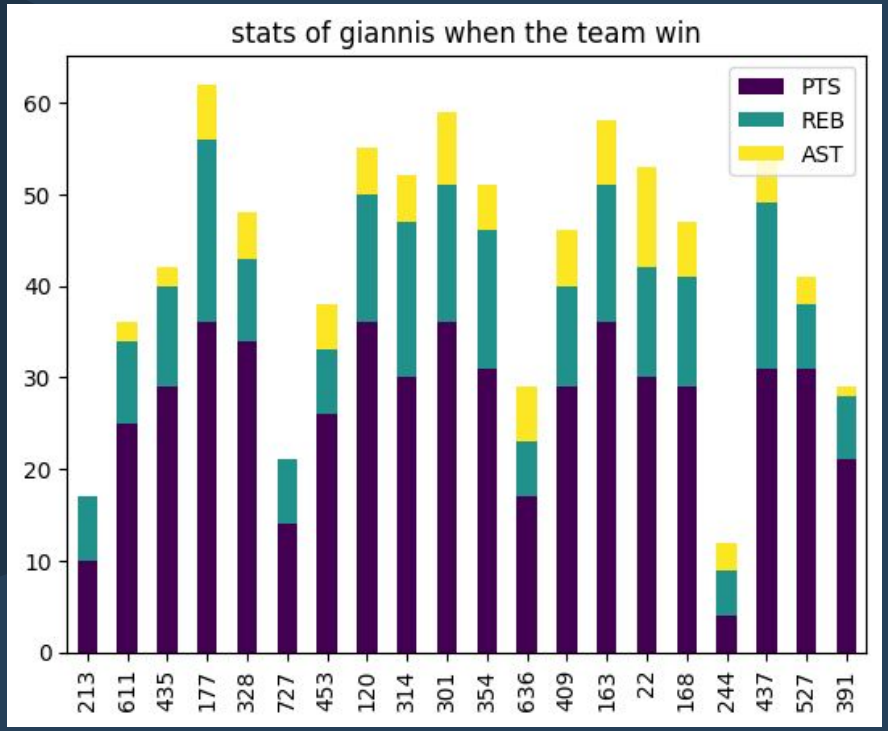
VISUALIZATION

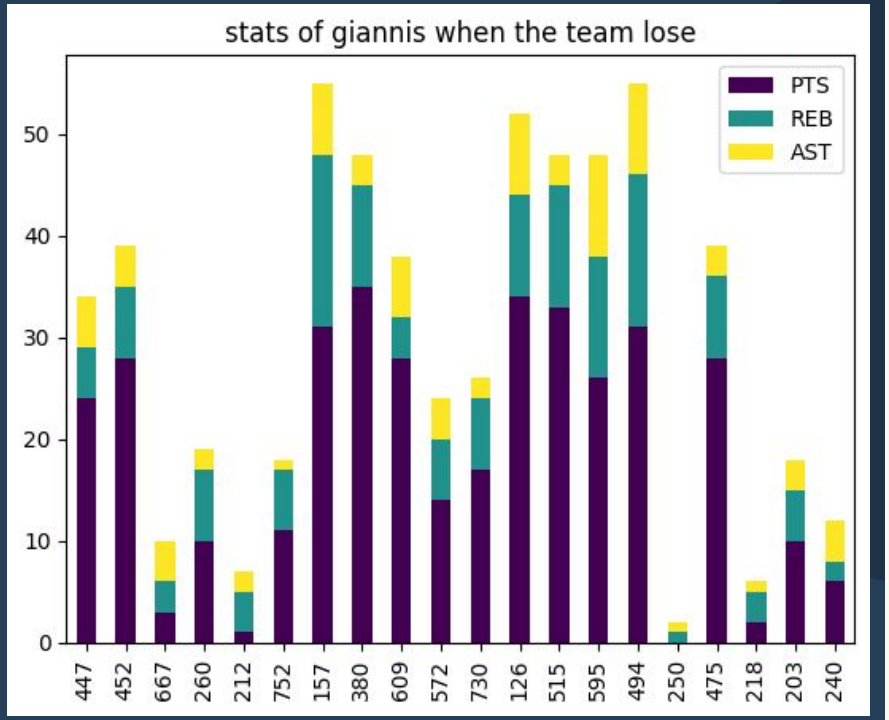



Giannis Plus_Minus evolution
PLUS_MINUS
10
5
0
2013
2014
2015
2016
2017
2018
2019
2020
2021
SEASON


buck win percentage evolution
HOME_TEAM_WINS
0.8
0.7
0.6
0.5
0.4
0.3
2013
2014
2015
2016
2017
2018
2019
2020
2021
SEASON

VISUALIZATION





03

MySQL Queries

ENTITY RELATIONSHIP MODEL

```sql
create database nba;
use nba;

SELECT TEAM_ID FROM players_sample
WHERE TEAM_CITY = "milwaukee";

# milwaukee team ID is 1610612749

```

Result Grid | Filter Rows: | Export: | Wra

TEAM_ID
1610612749
1610612749
1610612749
1610612749
1610612749
1610612749
1610612749

```sql

# milwaukee team ID is 1610612749

SELECT * FROM games_sample
WHERE TEAM_ID_home = 1610612749 OR TEAM_ID_away = 1610612749;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

MyUnknownColumn	GAME_DATE_EST	GAME_ID	GAME_STATUS_TEXT	HOME_TEAM_ID
3	2022-03-12	22101008	Final	1610612744
22	2022-03-09	22100984	Final	1610612749
35	2022-03-08	22100979	Final	1610612760
46	2022-03-06	22100961	Final	1610612749
63	2022-03-04	22100949	Final	1610612741
79	2022-03-02	22100936	Final	1610612749

```
12 •  SELECT * FROM players_sample AS players
13    WHERE PLAYER_NAME = "Giannis Antetokounmpo";
14 •  SELECT avg(PTS), avg(REB), avg(AST) FROM players_sample AS players
15    WHERE PLAYER_NAME = "Giannis Antetokounmpo";
16
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

avg(PTS)	avg(REB)	avg(AST)
37.666666666666664	9	5

```
create table giannis
select * from players_sample where PLAYER_NAME = "Giannis Antetokounmpo" ;
```

```
24 •  SELECT *
25    FROM games_sample gs
26    LEFT JOIN players_sample pl
27    using (GAME_ID)
28    having pl.PLAYER_NAME = "Giannis Antetokounmpo";
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

GAME_ID	MyUnknownColumn	GAME_DATE_EST	GAME_STATUS_TEXT	HOME_TEAM_ID	VISITOR_TEAM_ID	SEASON	TEAM_ID_home	PTS_home	FG_PCT_home	FT_PCT_home	FG3_PCT_hor
22101008	3	2022-03-12	Final	1610612744	1610612749	2021	1610612744	122	0.484	0.933	0.4
22100984	22	2022-03-09	Final	1610612749	1610612737	2021	1610612749	124	0.5	0.783	0.343
22100979	35	2022-03-08	Final	1610612760	1610612749	2021	1610612760	115	0.5	0.714	0.41

04

SUPERVISED MACHINE LEARNING MODELS

FEATURES SELECTION

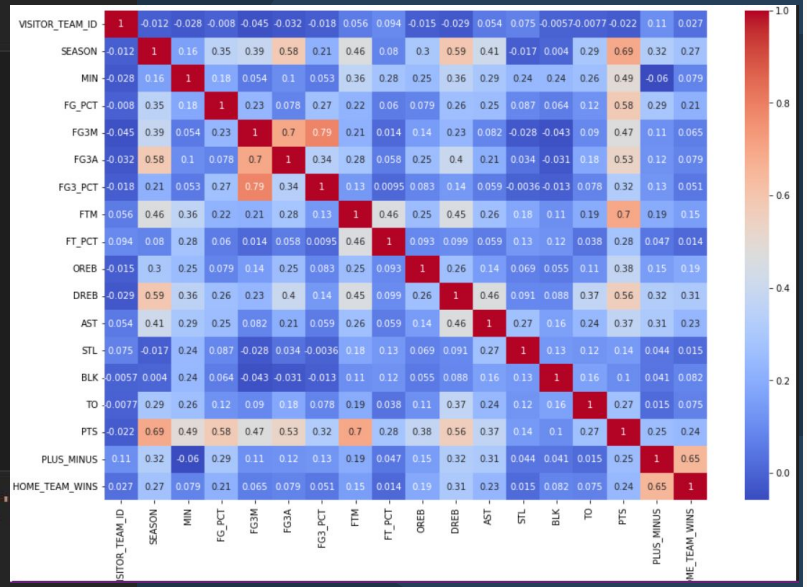
```
df.columns
```

✓ 0.7s

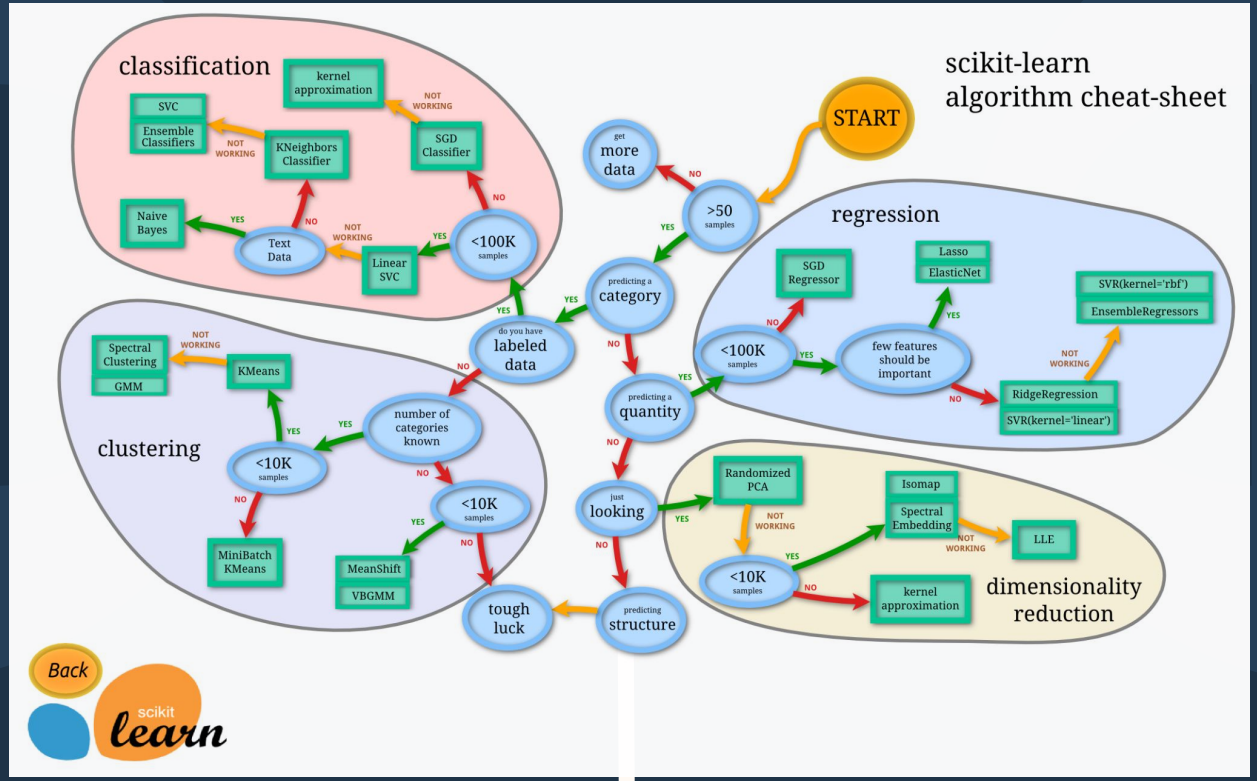
```
Index(['Unnamed: 0', 'GAME_DATE_EST', 'GAME_ID', 'HOME_TEAM_ID',
       'VISITOR_TEAM_ID', 'SEASON', 'PTS_home', 'FG_PCT_home', 'FT_PCT_home',
       'FG3_PCT_home', 'AST_home', 'REB_home', 'PTS_away', 'FG_PCT_away',
       'FT_PCT_away', 'FG3_PCT_away', 'AST_away', 'REB_away', 'HOME_TEAM_WINS',
       'TEAM_ID', 'TEAM_CITY', 'PLAYER_ID', 'PLAYER_NAME', 'START_POSITION',
       'MIN', 'FGA', 'FG_PCT', 'FG3M', 'FG3A', 'FG3_PCT', 'FTM', 'FTA',
       'FT_PCT', 'OREB', 'DREB', 'REB', 'AST', 'STL', 'BLK', 'TO', 'PTS',
       'PLUS_MINUS', 'colscatt'],
      dtype='object')
```

```
df = df[['VISITOR_TEAM_ID','SEASON','MIN', 'FG_PCT', 'FG3M', 'FG3A', 'FG3_PCT'
         'FT_PCT', 'OREB', 'DREB', 'AST', 'STL', 'BLK', 'TO', 'PTS',
         'PLUS_MINUS','HOME_TEAM_WINS']]
```

✓ 0.4s

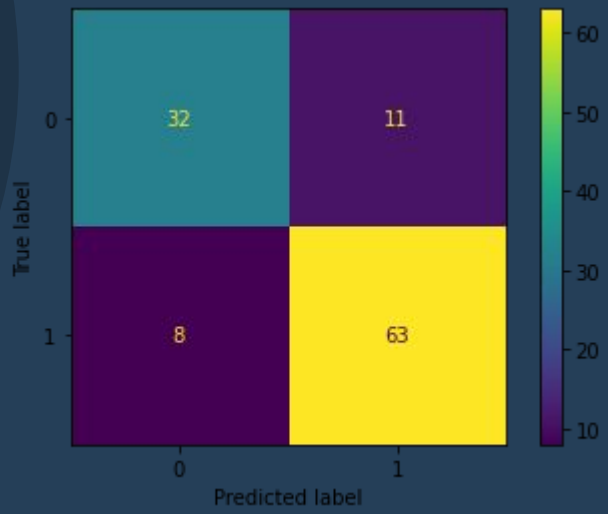


Models SELECTION

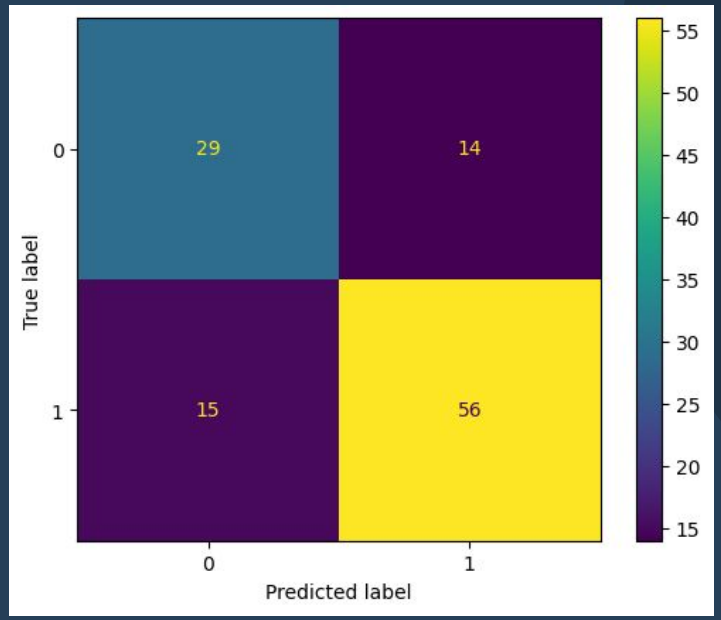


Models comparison

TPOT : EXTRA TREES CLASSIFIER



KNN Classifier



Models score

Model	Accuracy score	Precision score
Extra trees classifier	0.824	0.849
K nearest neighbour	0.684	0.701

05

Challenges

1 Manage my Time

2 Define the most fitting machine learning model

Thanks !