AC Milan

Associazione Calcio Milan

Project made by Ruoxi, Nhan, Diana and Felipe

Main Objective

Team Overview

- ❏ Identify weaknesses and strength
- ❏ Determine position that needs improvement

Performance Review

- ❏ Identify underperformed/overpaid players with **predictive model**

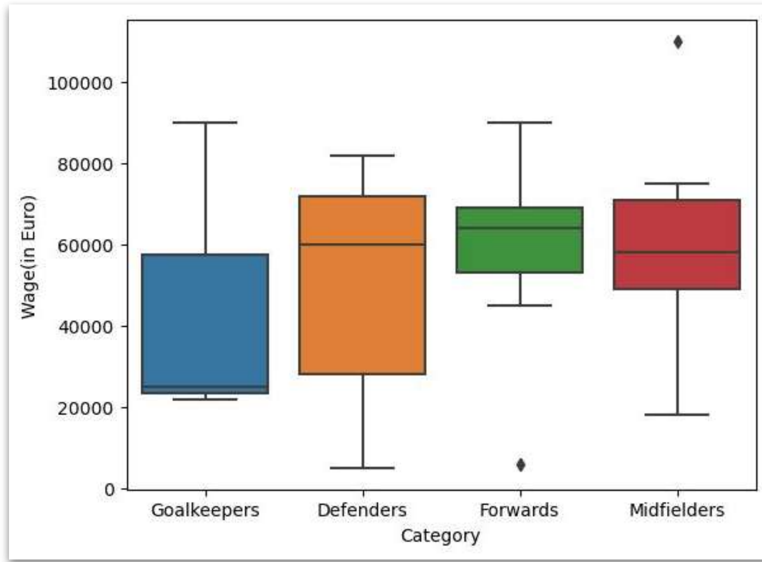
Development strategy

- ❏ Selection criteria for new players

Team Overview

AC Milan players profile per category

Category	Full Name	Overall	Potential	Wage(in Euro)	Age	Value(in Euro)
Defenders	11	78.0	82.0	60000.0	24.0	12500000.0
Forwards	8	79.0	82.0	64000.0	27.5	12250000.0
Goalkeepers	3	76.0	76.0	25000.0	36.0	825000.0
Midfielders	9	78.0	85.0	58000.0	22.0	22500000.0



- ❑ 2 out of 11 defenders going to retired
- ❑ Aging goalkeepers
- ❑ Young players not yet reach their potentials

Data Preparation

Remove null, duplicates

```
players.dtypes
all_players.describe()
players.isnull().value_counts()
players.shape
✓ 0.0s
(18539, 89)
```

```
players.duplicated().value_counts()
✓ 0.0s
False    18420
True       119
dtype: int64
```

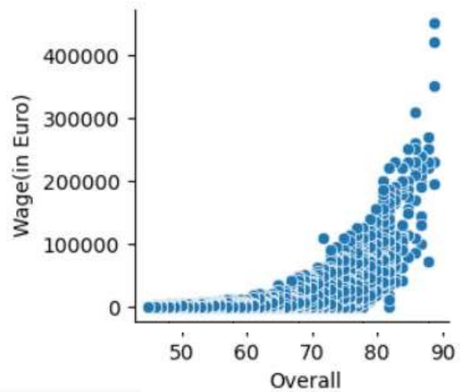
```
#Let's drop the duplicates from the dataframe
players = players.drop_duplicates()
players.shape
✓ 0.0s
(18420, 89)
```

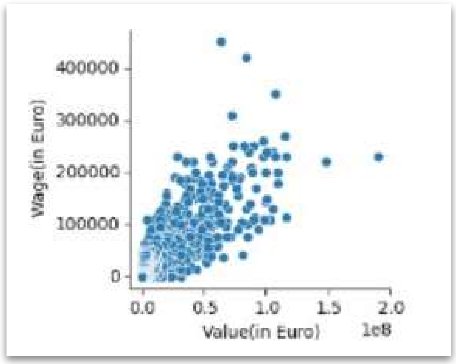
Removing outliers (retired, extreme pay)

```
#There are also players with wage or value equal to 0, those players need to be out of the model building as well
cleaned_all_players = all_players[(all_players['Value(in Euro)']>0)& (all_players['Age'] <= 40) & (all_players['Wage(in E
```

Performance and budget review

Salary Predictive Model



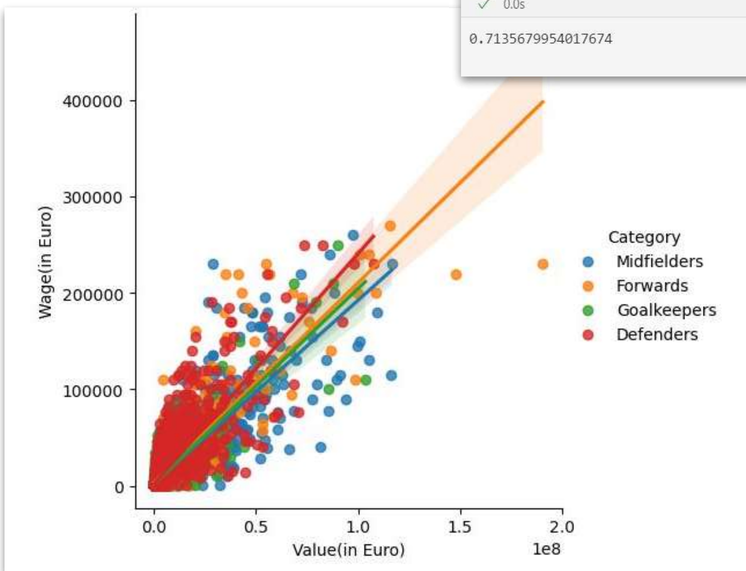



```
wage_model = LinearRegression()
wage_result = wage_model.fit(cleaned_all_players[['std_overall', 'std_value']], cleaned_all_players['Wage(in Euro)'])
print(wage_result.intercept_)
print(wage_result.coef_)
print(wage_result.score(cleaned_all_players[['std_overall','std_value']], cleaned_all_players['Wage(in Euro)']))

8618.757865937068
[ 3966.34962404 13112.11597396]
0.7133240572284634
```

- ➔ We identify independent variables with high correlation with Wage(in Euro)
- ➔ We consider to minimize multicollinearity
- ➔ We split the dataset into training data(the rest) and test data(AC Milan)
- ➔ The independent variables are standardized

Performance and budget review




```
enc_position = pd.get_dummies(cleaned_all_players[['std_overall', 'std_value','Category']])
test_model = LinearRegression()
test_result = test_model.fit(enc_position, cleaned_all_players['Wage(in Euro)'])
test_result.score(enc_position, cleaned_all_players['Wage(in Euro)'])
#The result does not really improve the R2 compared to the model with only 'std_overall' and 'std_value', so we'll use the first model built for the prediction
```

✓ 0.0s

```
0.7135679954017674
```

Performance and budget review

Salary Predictive Model - Outcomes

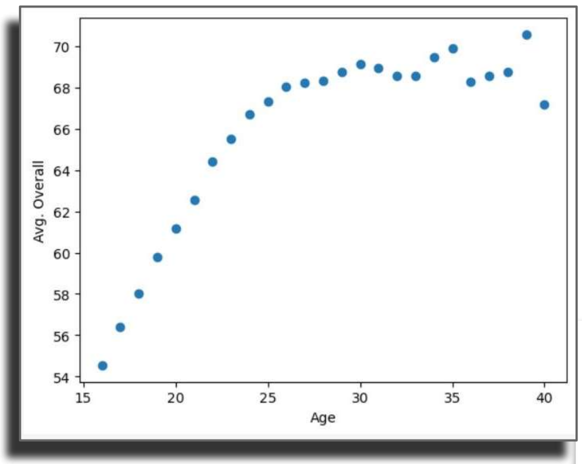
```
'''Then we want to see how the distribution of our club players are compared to the prediction regression model'''
ac_milan['predicted_wage'] =wage_model.predict(ac_milan[['std_overall', 'std_value']])
ac_milan
```

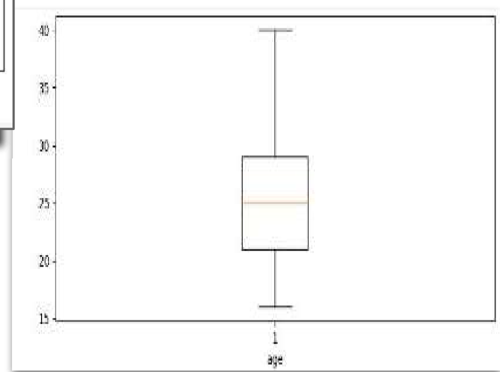
	Full Name	Overall	Potential	Value(in Euro)	Age	Wage(in Euro)	Release Clause	std_overall	std_value	Category	predicted_wage
37	Mike Maignan	87	90	80000000	26	90000	142000000	3.132508	10.092449	Goalkeepers	153376.745053
86	Theo Hernández	85	90	76000000	24	82000	134900000	2.837087	9.569235	Defenders	145344.555625
128	Rafael da Conceição Leão	84	90	66500000	23	90000	126400000	2.689376	8.326601	Forwards	128465.125877
131	Sandro Tonali	84	90	62500000	22	75000	118800000	2.689376	7.803387	Midfielders	121604.680526
145	Fikayo Tomori	84	90	60500000	24	75000	115000000	2.689376	7.541780	Defenders	118174.457850
222	Ismaël Bennacer	82	86	40000000	24	63000	71000000	2.393955	4.860307	Midfielders	81842.931347
249	Zlatan Ibrahimović	82	82	0	40	63000	0	2.393955	-0.371835	Forwards	13238.477834
255	Olivier Giroud	82	82	13000000	35	66000	22100000	2.393955	1.328611	Forwards	35534.925226
256	Simon Kjær	82	82	14500000	33	73000	24700000	2.393955	1.524817	Defenders	38107.592233
395	Davide Calabria	80	83	25500000	25	60000	45300000	2.098534	2.963656	Defenders	55802.072872
471	Ante Rebić	80	80	21000000	28	78000	35700000	2.098534	2.375040	Forwards	48084.071851
643	Alessandro	79	79	12500000	31	60000	21300000	1.950823	1.263210	Defenders	32910.753441

```
- players who have less predicted wage, and contract unil is more than 2023
n[(ac_milan['predicted_wage']*3/2<ac_milan['Wage(in Euro)']) & (ac_milan['Contract Until']> '2023')]
```

- ➔ The list narrow downs the list of players we might want to replace
- ➔ Other factors such as age, potential to take into account for transfer considerations
- ➔ The materiality of budget release once the player being transferred out

Development strategy





Selection criteria

- Find young players (<27 yrs old)

- Players who has high potential ranking.

- Players with current overall above our team average (same position category).

We will favor players that meet criteria above and does not have release clause.

“Technicalities”

Example code for a LW players selection :

```
# Filter top 3 "LW" players with "Release Clause" < "Value(in Euro)" and age<27
filtered_LW = players[(players['Best Position'] == 'LW') & (players['Release Clause'] ==0) & (players['Age'] < 27)]

# Sort the filtered DataFrame by "Overall" in descending order
sorted_LW = filtered_LW.sort_values(by='Overall', ascending=False)

# Extract the top 3 players
top_3_players = sorted_LW.head(3)

# Display the top 3 players
display(top_3_players)
```

Full Name	Overall	Potential	Value(in Euro)	Positions Played	Best Position	Age
Nemanja Radonjić	74	75	5000000	LW,RW,CF	LW	26
Christos Tzolis	72	80	5000000	LW,RW	LW	20
Abdessamad Ezzalzouli	70	82	3800000	LW,RW	LW	20

Challenges

Comprehending the significance of all data
Determining which columns to omit that are not essential for our analysis
Identification of the most effective criteria for this process

Example of a new data frame

	Full Name	Overall	Potential	Value(in Euro)	Best Position	Age	Wage(in Euro)	Release Clause
0	Lionel Messi	91	91	54000000	CAM	35	195000	99900000
1	Karim Benzema	91	91	64000000	CF	34	450000	131199999
2	Robert Lewandowski	91	91	84000000	ST	33	420000	172200000
3	Kevin De Bruyne	91	91	107500000	CM	31	350000	198900000
4	Kylian Mbappé	91	95	190500000	ST	23	230000	366700000

	Known As	Full Name	Release Clause	Club Position	Contract Until
0	L. Messi	Lionel Messi	99900000	RW	2023
1	K. Benzema	Karim Benzema	131199999	CF	2023
2	R. Lewandowski	Robert Lewandowski	172200000	ST	2025
3	K. De Bruyne	Kevin De Bruyne	198900000	CM	2025
4	K. Mbappé	Kylian Mbappé	366700000	ST	2024