Business Report: Fantasy Sports Clustering Analysis

Extended Project Report

Submitted to

greatlearning
Learning for Life

By

Subhadeep Seal

In Partial Fulfillment of PDP-DSBA

TEXAS McCombs
The University of Texas at Austin
McCombs School of Business

CONTENTS

1. PROBLEM STATEMENT

1.1. Context

Fantasy sports are online gaming platforms where participants draft and manage virtual teams of real professional sports players. Based on the performance of the players in the real world, players are allotted points in the fantasy sports platform every match. The objective is to create the best possible team with a fixed budget to score maximum fantasy points, and users compete against each other over an entire sports league or season. Some of these fantasy sports require actual financial investments for participation, with the chances of winning monetary rewards as well as free match-day tickets periodically.

The fantasy sports market has seen tremendous growth over the past few years, with a valuation of $18.6 billion in 2019. The football (soccer) segment led in terms of market share in 2019, with over 8 million participants worldwide, and is expected to retain its dominance over the next couple of years. Digitalization is one of the primary factors driving the growth of the fantasy sports market as it allows participants the opportunity to compete on a global level and test their skills. With an increase in smart-phone usage and availability of fantasy sports apps, this market is expected to witness a global surge and reach a $48.6 billion valuation by 2027.

1.2. Problem Definition

OnSports wants to determine appropriate starting prices for players in the upcoming English Premier League season by understanding performance trends and clustering players based on their potential using past season data.

1.3. Objective

OnSports is a fantasy sports platform that has fantasy leagues for many different sports and has witnessed an increasing number of participants globally over the past 5 years. For each player, a price is set at the start, and the price keeps changing over time based on the performance of the players in the real world. With the new English Premier League season about to start, they have collected data of the past season and want to analyze it to determine the price of each player for the start of the new season. OnSports have hired you as a data scientist and asked you to conduct a cluster analysis to identify players of different potentials of each player based on previous season performance. This will help them understand the patterns in player performances and fantasy returns and decide the exact price to be set for each player for the upcoming football season.

1.4. Data Description

The data comprises player stats like the number of goals scored, goals created, minutes played, fantasy points scored in the previous season, etc. The detailed data dictionary is given below.

Data Dictionary

- Player_Name: Name of the player
- Club: Club in which the player plays
- Position: The position in which the player plays
- Goals_Scored: Number of goals scored by the player in the previous season
- Assists: Number of passes made by the player leading to goals in the previous season
- Total_Points: Total number of fantasy points scored by the player in the previous season
- Minutes: Number of minutes played by the player in the previous season
- Goals_Conceded: Number of goals conceded by the player in the previous season
- Creativity: A score, computed using a range of stats, that assesses player performance in terms of producing goals scoring opportunities for other players
- Influence: A score, computed using a range of stats, that evaluates a player's impact on a match, taking into account actions that could directly or indirectly affect the match outcome
- Threat: A score, computed using a range of stats, that gauges players who are most likely to score goals
- Bonus: Total bonus points received (The three best-performing players in each match receive additional bonus points based on a score computed using a range of stats. 3 points are awarded to the highest-scoring player, 2 to the second-best, and 1 to the third.)
- Clean_Sheets: Number of matches without conceding a goal in the previous season.

2. DATA OVERVIEW

- We will view the 10 sample rows of the dataset.

	Player_Name	Club	Position	Goals_Scored	Assists	Total_Points	Minutes	Goals_Conceded	Creativity	Influence	Threat	Bonus	Clean_Sheets
441	Mark Noble	West Ham United	Midfielder	0	0	27	701	15	88.6	80.4	7	0	0
363	Sean Longstaff	Newcastle United	Midfielder	0	1	41	1405	26	182.8	179.2	148	1	2
31	Anwar El Ghazi	Aston Villa	Midfielder	10	0	111	1604	22	426.1	500.4	726	13	5
132	Olivier Giroud	Chelsea	Forward	4	0	47	740	5	112.0	161.4	403	6	4
90	Chris Wood	Burnley	Forward	12	3	138	2741	43	323.2	595.8	1129	16	9
249	Vontae Daley-Campbell	Leicester City	Defender	0	0	0	0	0	0.0	0.0	0	0	0
65	Danny Welbeck	Brighton and Hove Albion	Forward	6	4	89	1541	18	269.7	319.8	595	15	6
445	Ryan Fredericks	West Ham United	Defender	1	1	28	564	9	166.8	155.2	96	0	1
117	Christian Pulisic	Chelsea	Midfielder	4	3	82	1731	21	378.8	361.4	724	3	7
415	Ryan Sessegnon	Tottenham Hotspurs	Defender	0	0	0	0	0	0.0	0.0	0	0	0

Table 1: First 5 & last 5 rows of the dataset

2.1. Shape of the Dataset

- The dataset contains 476 rows & 13 columns.

2.2. Check the type of data

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 476 entries, 0 to 475
Data columns (total 13 columns):
 #   Column          Non-Null Count  Dtype  
---  ------          --------------  -----  
 0   Player_Name     476 non-null    object 
 1   Club            476 non-null    object 
 2   Position        476 non-null    object 
 3   Goals_Scored    476 non-null    int64  
 4   Assists         476 non-null    int64  
 5   Total_Points    476 non-null    int64  
 6   Minutes         476 non-null    int64  
 7   Goals_Conceded  476 non-null    int64  
 8   Creativity      476 non-null    float64
 9   Influence       476 non-null    float64
 10  Threat          476 non-null    int64  
 11  Bonus           476 non-null    int64  
 12  Clean_Sheets    476 non-null    int64  
dtypes: float64(2), int64(8), object(3)
memory usage: 48.5+ KB
```

Table 2: Data types

- There are 3 object data types, 8 integer data types, and 2 float data type in the dataset. Player_Name, Club & Position are the only object type columns the rest are numerical. All these features could be good predictors understanding performance trends.

2.3. Check for missing values

	0
Player_Name	0
Club	0
Position	0
Goals_Scored	0
Assists	0
Total_Points	0
Minutes	0
Goals_Conceded	0
Creativity	0
Influence	0
Threat	0
Bonus	0
Clean_Sheets	0

dtype: int64

Table 3: Missing Values

- There are no missing values in the dataset.

2.4. Statistical summary of the dataset

	count	unique	top	freq	mean	std	min	25%	50%	75%	max
Player_Name	476	476	Willy Boly	1	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Club	476	17	Arsenal	30	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Position	476	4	Midfielder	195	NaN	NaN	NaN	NaN	NaN	NaN	NaN
Goals_Scored	476.0	NaN	NaN	NaN	1.907563	3.455562	0.0	0.0	0.5	2.0	23.0
Assists	476.0	NaN	NaN	NaN	1.752101	2.708563	0.0	0.0	0.0	2.0	14.0
Total_Points	476.0	NaN	NaN	NaN	58.516807	51.293559	0.0	10.0	48.0	94.25	244.0
Minutes	476.0	NaN	NaN	NaN	1336.909664	1073.773995	0.0	268.75	1269.5	2256.25	3420.0
Goals_Conceded	476.0	NaN	NaN	NaN	19.157563	15.946171	0.0	4.0	18.0	31.0	68.0
Creativity	476.0	NaN	NaN	NaN	195.97605	251.478541	0.0	8.3	96.95	296.95	1414.9
Influence	476.0	NaN	NaN	NaN	294.617647	267.779681	0.0	46.5	233.1	499.5	1318.2
Threat	476.0	NaN	NaN	NaN	224.962185	318.240377	0.0	5.75	104.5	298.25	1980.0
Bonus	476.0	NaN	NaN	NaN	4.718487	6.252625	0.0	0.0	2.0	7.0	40.0
Clean_Sheets	476.0	NaN	NaN	NaN	4.745798	4.394312	0.0	0.0	4.0	8.0	19.0

Table 4: Statistical summary

In the above table we can see the counts, mean, standard deviation, minimum value and maximum value of numerical features.

2.5. Observations and Insights:

- Goals_Scored: Number of goals scored by the player in the previous season.
  - Over 25% of players have scored no goals with a median of 0.5 goals.
  - **Difference between 75th percentile and max could indicate a possible outlier**, but more likely representing strikers who get most goals.
- Assists: Number of passes made by the player leading to goals in the previous season.
  - Over 50% of players have no assist.
- Total_Points: Total number of fantasy points scored by the player in the previous season.
  - Median number of points is 48 with a min of 0 and max of 244.
  - **Difference between 75th percentile and max could indicate a possible outlier.**
- Minutes: Number of minutes played by the player in the previous season.
  - Looks to be normally distributed with an average of 1336min (22.27h), a min of 0 and max of 3420.
- Goals_Conceded: Number of goals conceded by the player in the previous season.
  - Looks to be normally distributed with an average of 19, a min of 0 and a max of 68
- Creativity: A score computed using a range of stats, which assesses player performance in terms of producing goal scoring opportunities for other players.
  - **Difference between 75th percentile and max could indicate a possible outlier**
- Influence: A score computed using a range of stats that evaluates a player's impact on a match, taking into account actions that could directly or indirectly affect the match outcome.
  - Looks to be relatively normally distributed with a slight right skew.
- Threat: A score computed using a range of stats that gauges players who are most likely to score goals.
  - Very heavily right skewed, which makes sense provided strikers score most goals.
- Bonus: Total bonus points received. The three best performing players in each match receive additional bonus points based on a score computed using a range of stats. 3 points are awarded to the highest scoring player, 2 to the second best, and 1 to the third.
  - Very heavily right skewed perhaps indicating consistently high performance players.
- Clean_Sheets: Number of matches without conceding a goal in the previous season.
  - More than 25% of players have 0 matches without conceding a goal, who are likely substitutes who do not get game time.

3. EXPLORATORY DATA ANALYSIS (EDA)

3.1. Univariate Analysis

Revealed distributions of 'Goals_Scored', 'Assists', 'Goals_Conceded', 'Clean_Sheets', 'Minutes', 'Total_Points', 'Creativity', 'Influence', 'Threat' & 'Bonus' , 'Club' & 'Position'. **Histogram-Box plots & Labeled Bar-plots** for each distribution are as follows:

Goals_Scored:

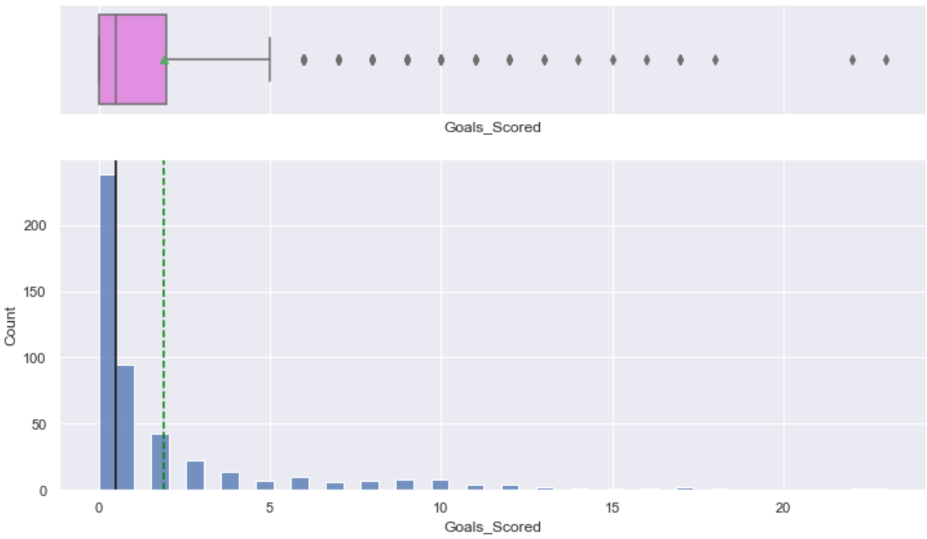

Fig-1

Assists:

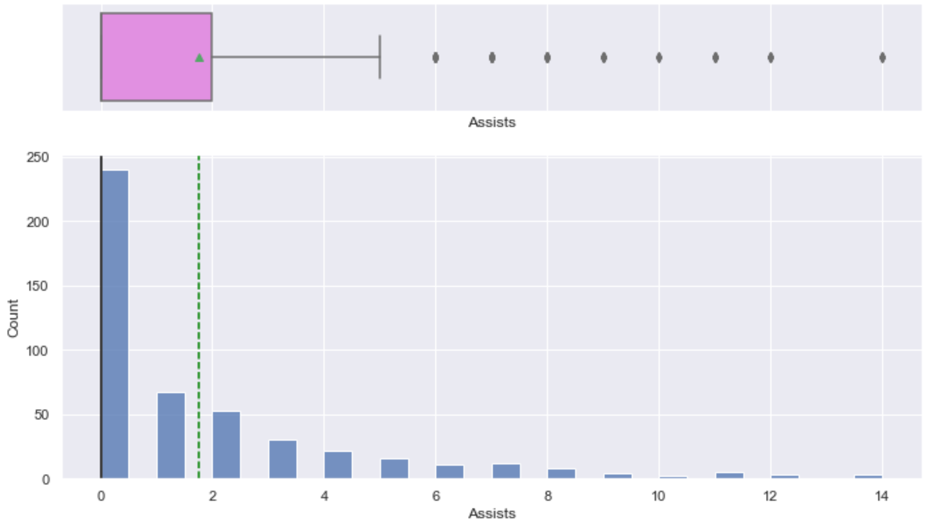

Fig-2

Goals_Conceded:

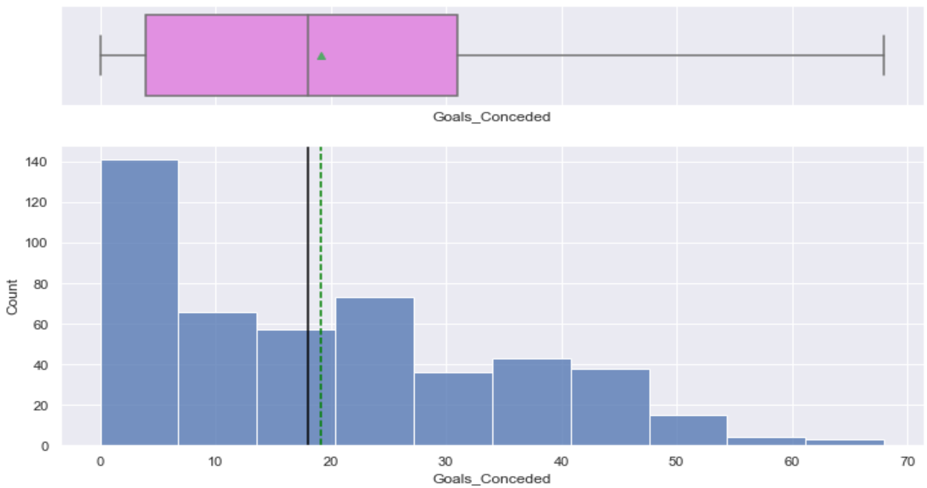

Fig-3

Clean_sheets:

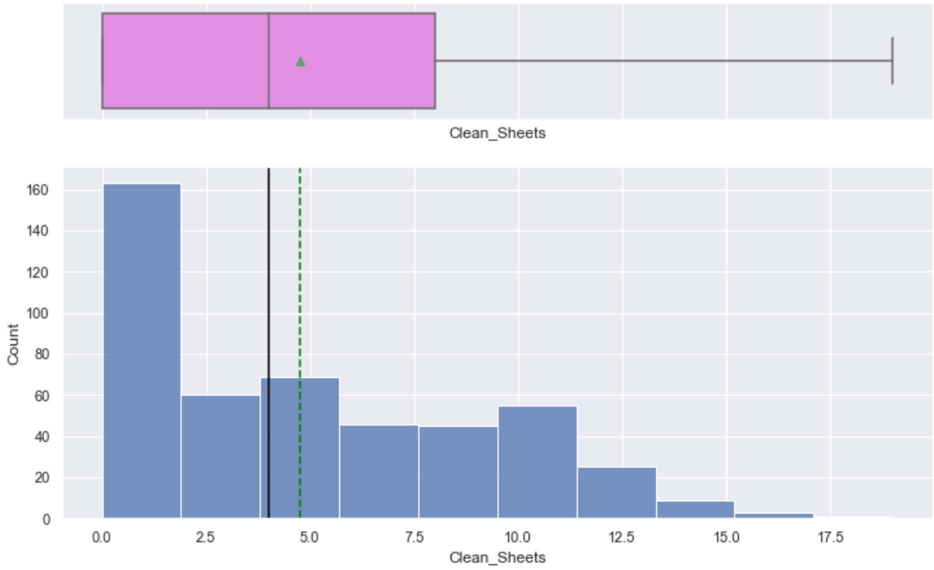

Fig-4

Minutes:

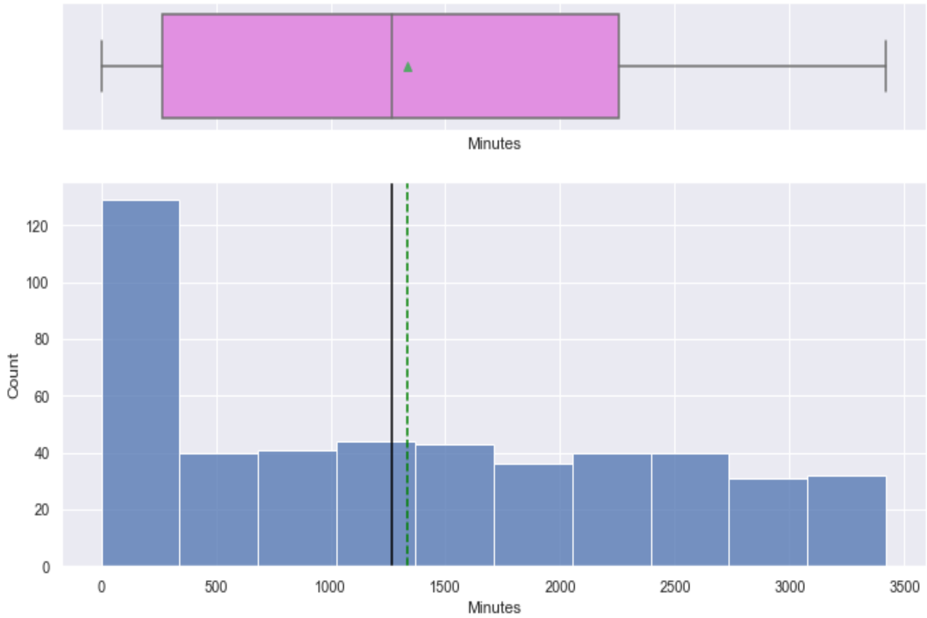

Fig-5

Total_Points:

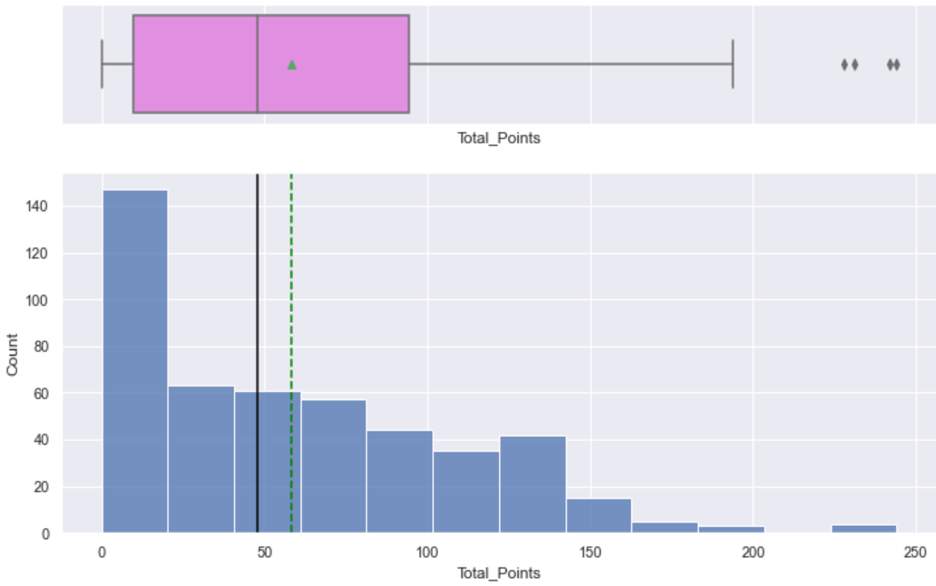

Fig-6

Creativity:

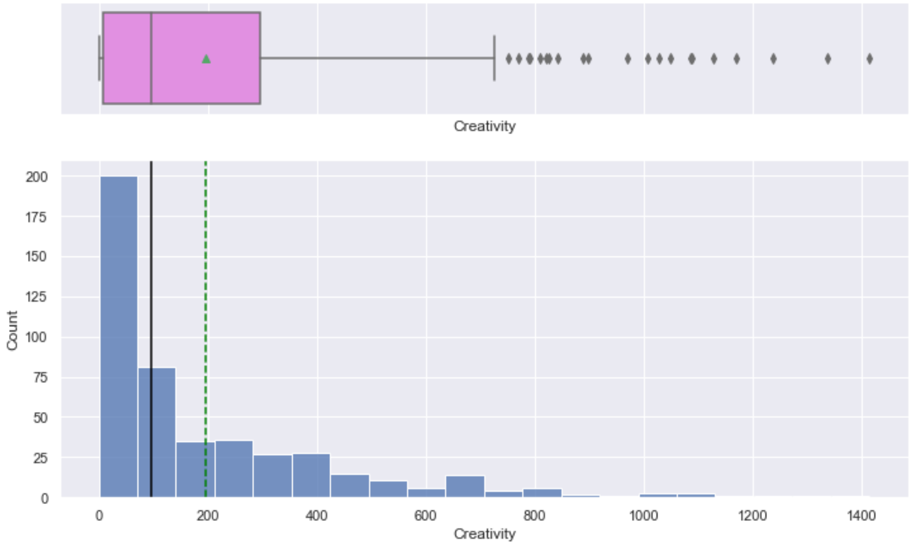

Fig-7

Influence:

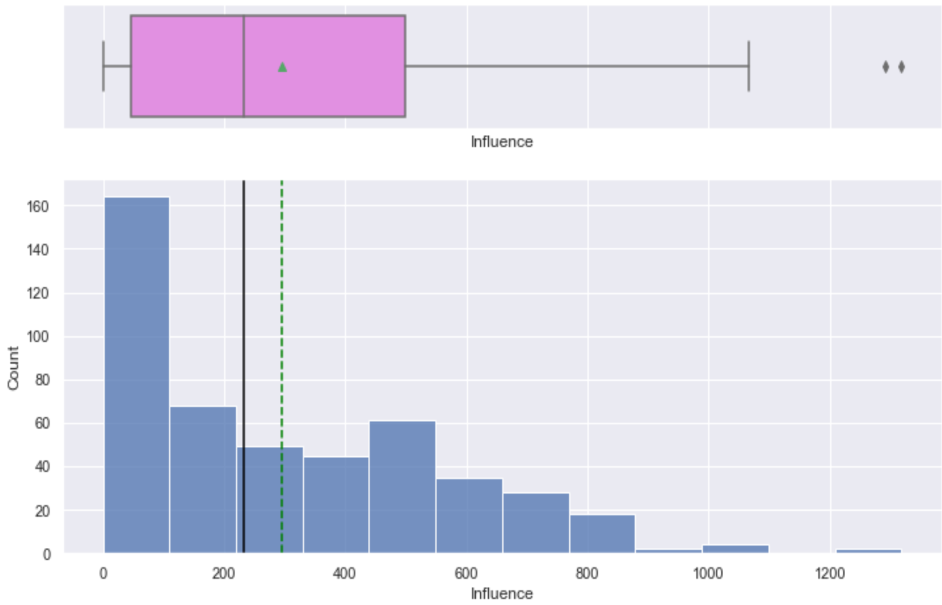

Fig-8

Threat:

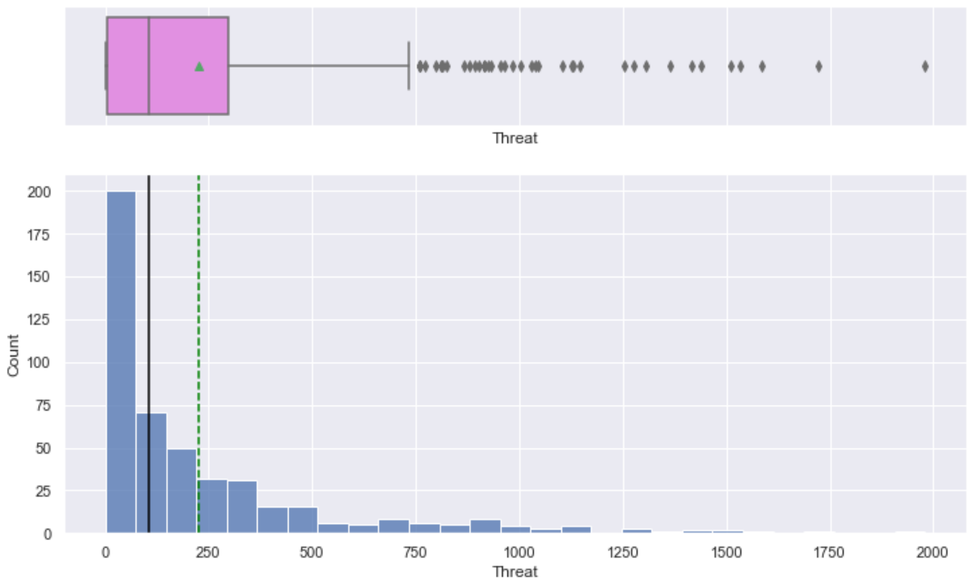

Fig-9

Bonus:

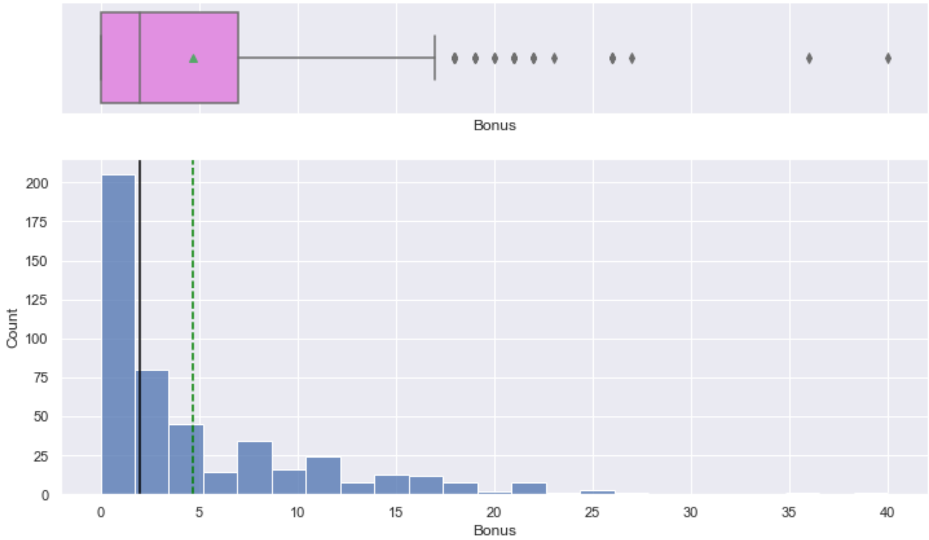

Fig-10

Observations and Insights: The **right skewed nature** is consistent through all plots indicate this is **not likely due to presence of outliers** but rather a natural imbalance in the players. This imbalance likely stems from one of two factors:

- Players who are higher performers.
- Certain positions that tend to rank higher on the measurable features.

Club:

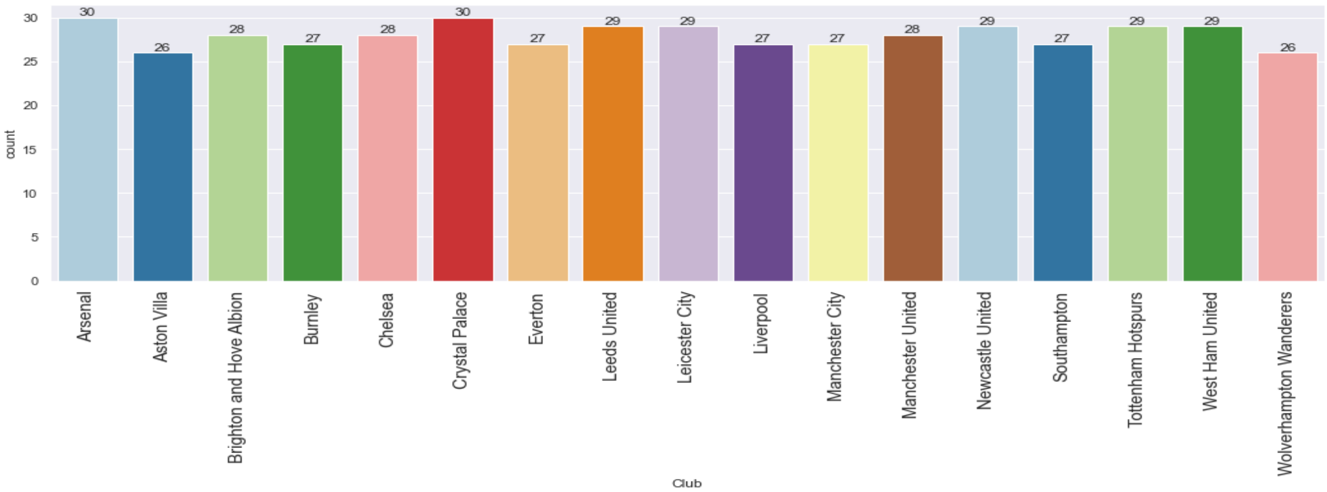

Fig-11

Relatively **uniform distribution** of players from each club should help to minimize potential errors from imbalanced data.

Position:

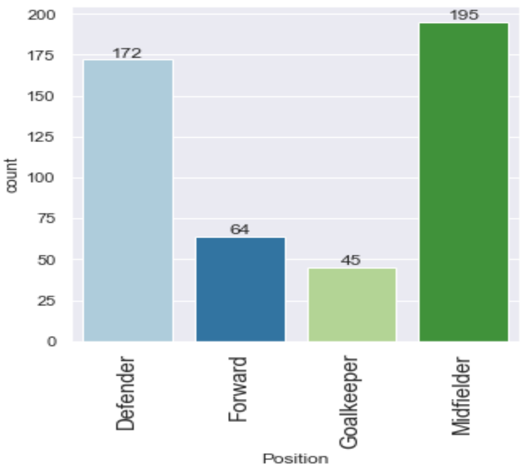

Fig-12

Of the 11 players on the field depending on the formation there is 1 goalie, 3-5 defenders, 4-5 midfielders, 1-3 forwards.

- The split shown above matches those ratios with positions ranked as:
  - Number of Midfielders > Defenders > Forwards > Goalkeepers.
- Given the number of Clubs and Goal Keepers, each club has on average 2-3 goal keepers.

3.2. Bivariate Analysis

We are done with univariate analysis. Let's explore the data a bit more with bivariate analysis.

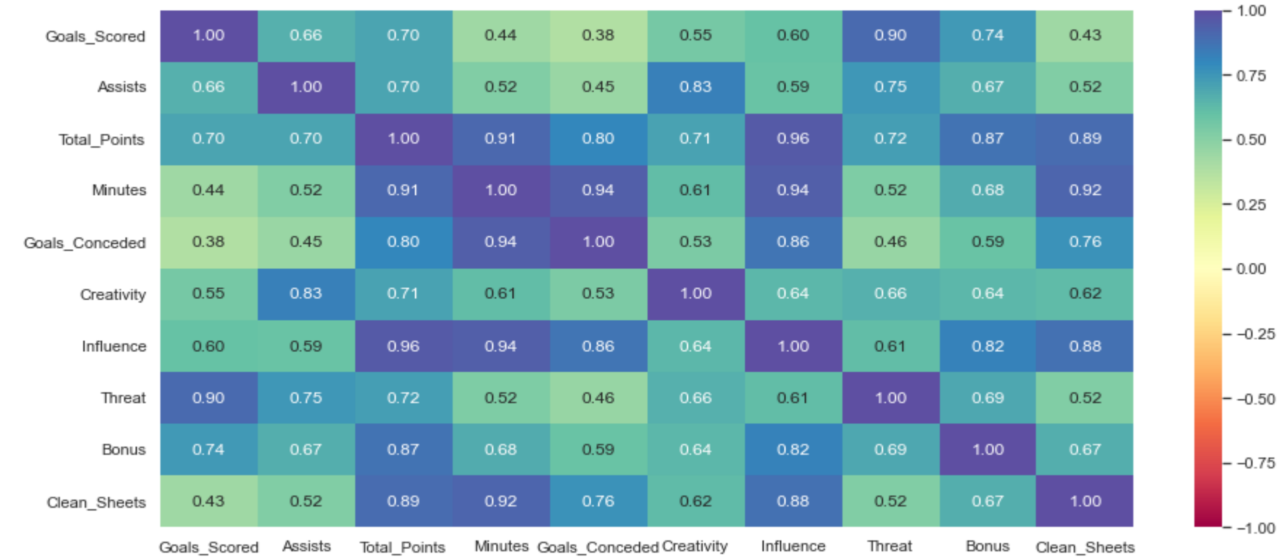

Fig-13

Observations and Insights: There is a high correlation (>= 0.7 or <= -0.7) between:

- Correlation between Assists, Goals_Scored, and Total_Points, which makes sense given the first 2 contribute to the 3rd and those likely to score are also likely to get assists.
- Big correlation (.91) between minutes played and Total_points, which makes sense as this gives players more chances.
- Correlation between Goals_Conceded, Total_ Points, Minutes, which echo our above observation that those without goals conceded are likely not getting game time.
- Correlation between Creativity and Assits, given creativity is a measure of, "a score computed using a range of stats that assesses player performance in terms of producing goal scoring opportunities for other players" that is likely mostly measured by Assits.

Could continue, but the **most relevant observation is that many of these features are highly correlated**.

Checking which players from which team have scored the most fantasy points on average.

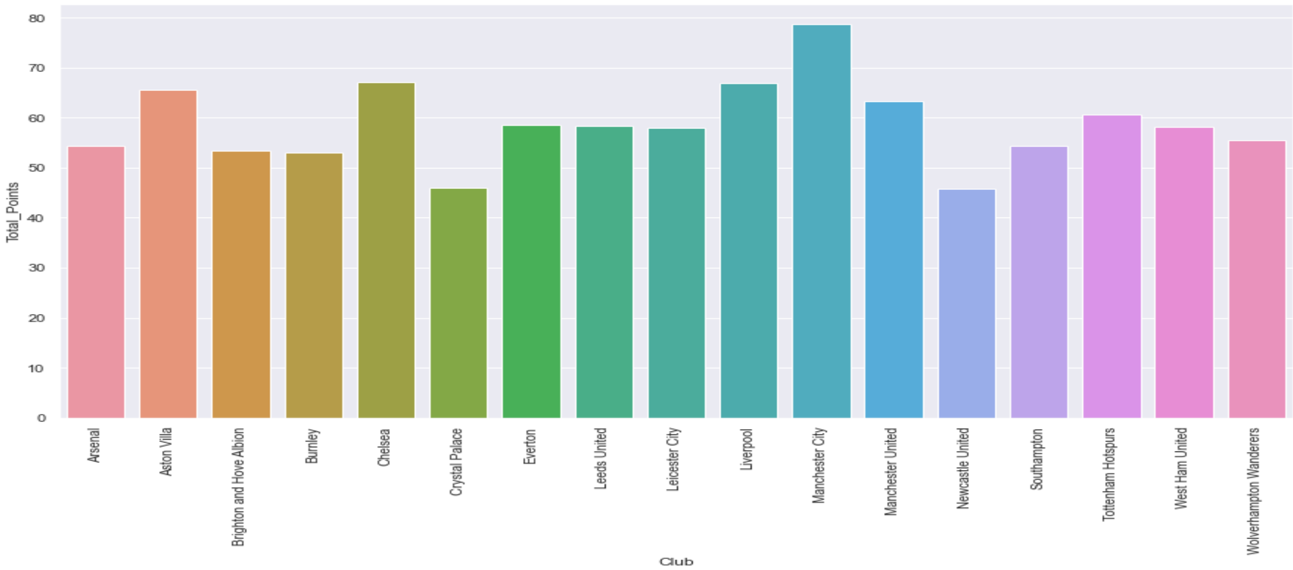

Fig-14

Manchester City is the leader in points while Crystal Palace and Newcastle United have the lowest number of points.

We hypothesized that players in different positions have scored more goals. Check which positions tend to score more fantasy points on average.

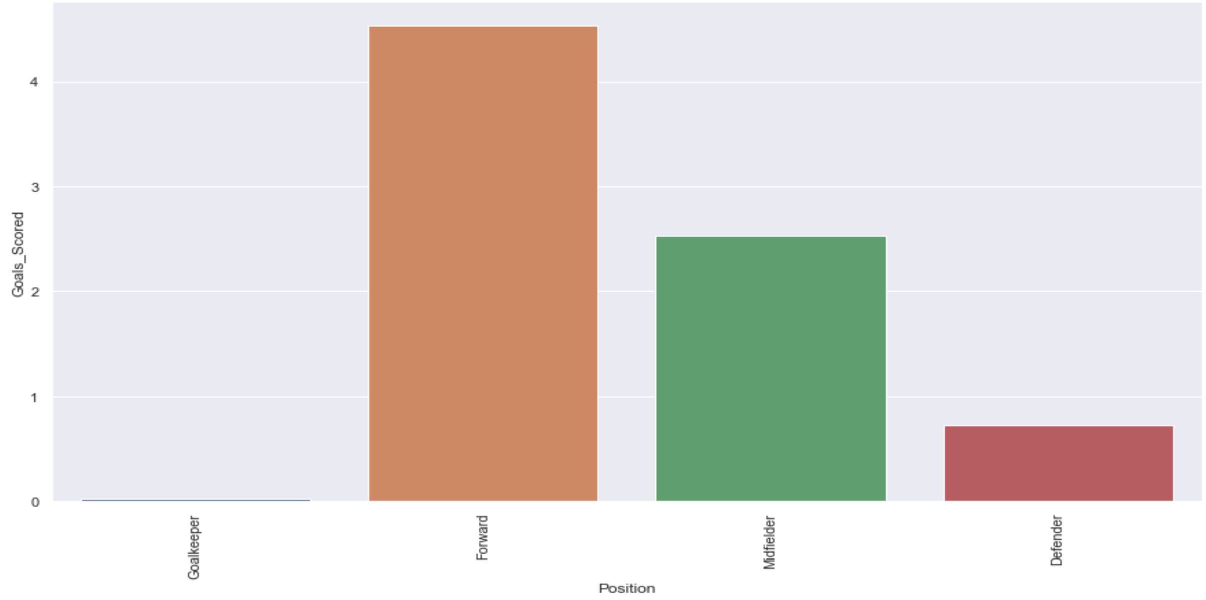

Fig-15

We will check if the same is true for the number of points.

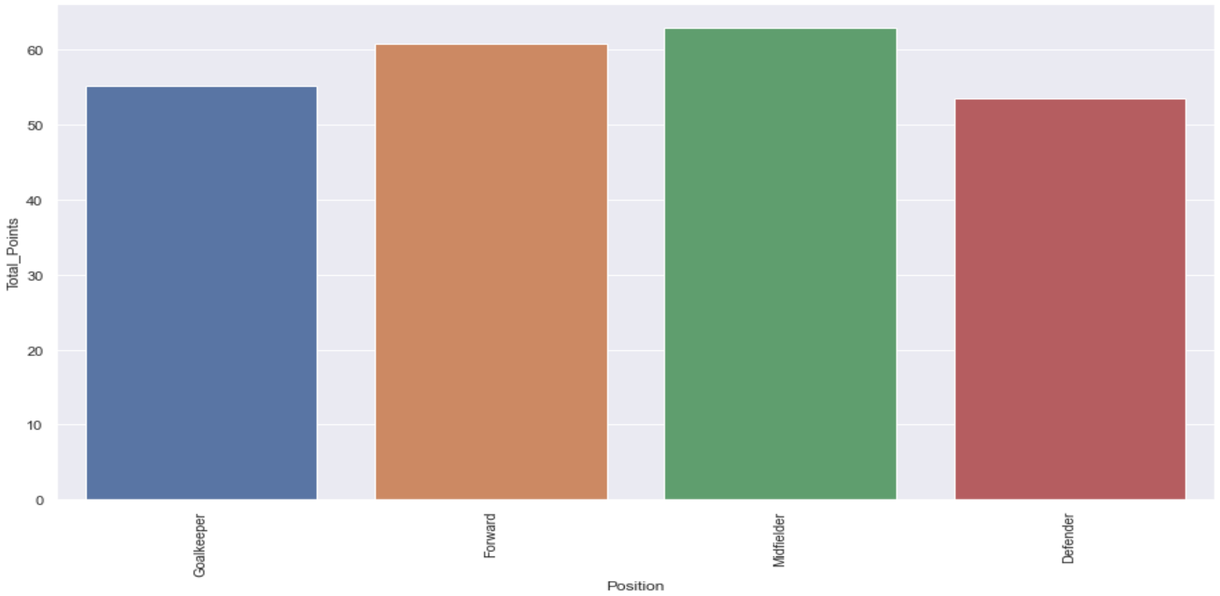

Fig-16

Total number of points is much more evenly distributed through the positions, with Midfielders > Forwards > Defenders > Goalkeepers.

Now we'll see which players scored the most fantasy points last season for different positions of play.

	Player_Name	Club	Position	Total_Points
36	Emiliano Martinez	Aston Villa	Goalkeeper	186
403	Harry Kane	Tottenham Hotspurs	Forward	242
315	Bruno Fernandes	Manchester United	Midfielder	244
223	Stuart Dallas	Leeds United	Defender	171

Table 5

Let's see the top 5 players with the most fantasy points last season for different positions of play.

	Player_Name	Club	Position	Total_Points
0	Emiliano Martinez	Aston Villa	Goalkeeper	186
1	Ederson Moares	Manchester City	Goalkeeper	160
2	Illan Meslier	Leeds United	Goalkeeper	154
3	Hugo Lloris	Tottenham Hotspurs	Goalkeeper	149
4	Nick Pope	Burnley	Goalkeeper	144
5	Alisson Becker	Liverpool	Goalkeeper	140
0	Harry Kane	Tottenham Hotspurs	Forward	242
1	Patrick Bamford	Leeds United	Forward	194
2	Jamie Vardy	Leicester City	Forward	187
3	Ollie Watkins	Aston Villa	Forward	168
4	Dominic Calvert-Lewin	Everton	Forward	165
5	Roberto Firmino	Liverpool	Forward	141
0	Bruno Fernandes	Manchester United	Midfielder	244
1	Mohamed Salah	Liverpool	Midfielder	231
2	Heung-Min Son	Tottenham Hotspurs	Midfielder	228
3	Sadio Mane	Liverpool	Midfielder	176
4	Marcus Rashford	Manchester United	Midfielder	174
5	Jack Harrison	Leeds United	Midfielder	160
0	Stuart Dallas	Leeds United	Defender	171
1	Andrew Robertson	Liverpool	Defender	161
2	Trent Alexander-Arnold	Liverpool	Defender	160
3	Aaron Cresswell	West Ham United	Defender	153
4	Aaron Wan-Bissaka	Manchester United	Defender	144
5	Ruben Dias	Manchester City	Defender	142

Table 6

4. DATA PREPROCESSING

4.1. Outlier Check

- Plot box-plots of all numerical columns to check for outliers.

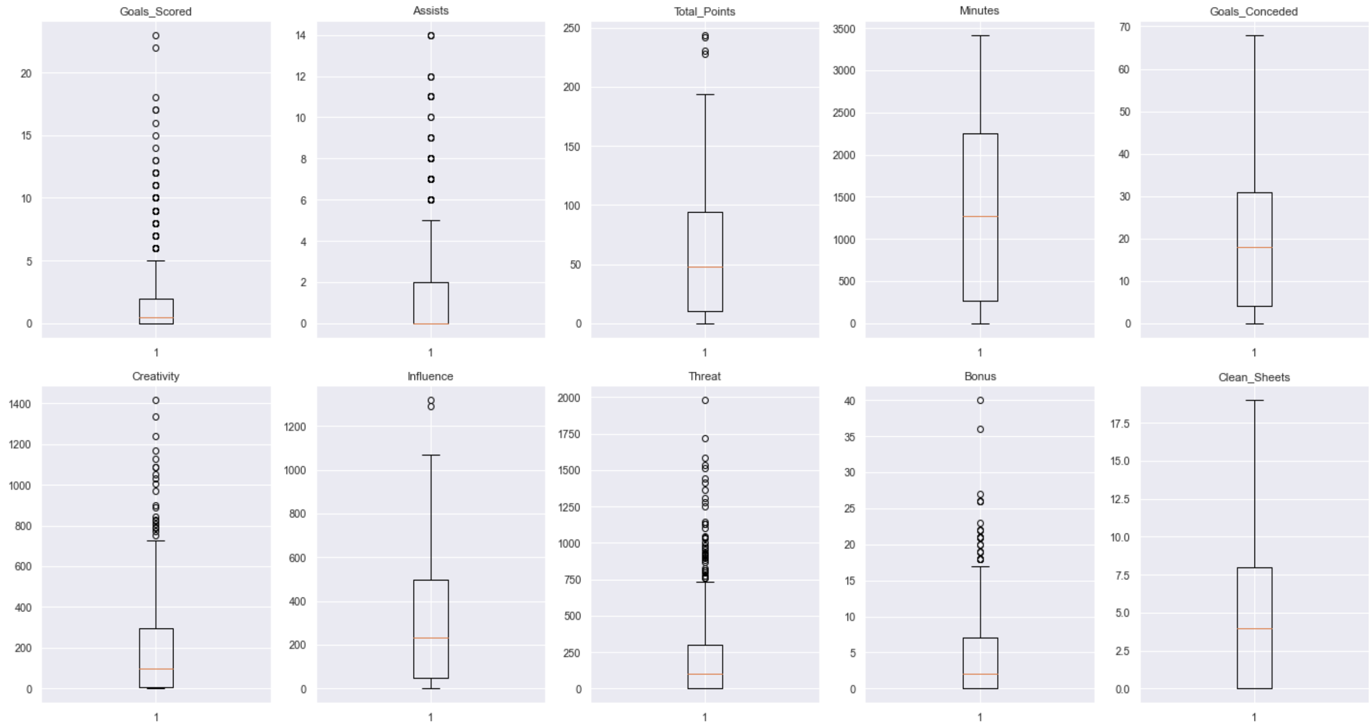

Fig-17

4.2. Scaling:

- Let's Scale the data before proceeding with clustering.

5. K-MEANS CLUSTERING

5.1. Checking Elbow Plot:

```
Number of Clusters: 1   Average Distortion: 2.7730371100978024
Number of Clusters: 2   Average Distortion: 1.8635736785898263
Number of Clusters: 3   Average Distortion: 1.5612774038101598
Number of Clusters: 4   Average Distortion: 1.3542782238901414
Number of Clusters: 5   Average Distortion: 1.2931541699741687
Number of Clusters: 6   Average Distortion: 1.2258495435854948
Number of Clusters: 7   Average Distortion: 1.16048401421345
Number of Clusters: 8   Average Distortion: 1.109804758457438
Number of Clusters: 9   Average Distortion: 1.0797310475776052
Number of Clusters: 10  Average Distortion: 1.017436992641063
Number of Clusters: 11  Average Distortion: 1.0208747020267823
Number of Clusters: 12  Average Distortion: 0.985073440903088
Number of Clusters: 13  Average Distortion: 0.9602766985773116
Number of Clusters: 14  Average Distortion: 0.9413187781558083
```

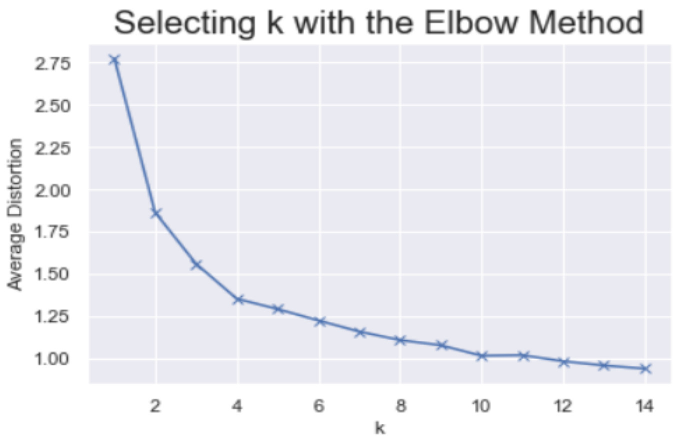

Fig-18

Observations and Insights:

- **We will move ahead with k = 4 as this is when the graph starts to move nearly parallel to the X-axis.**

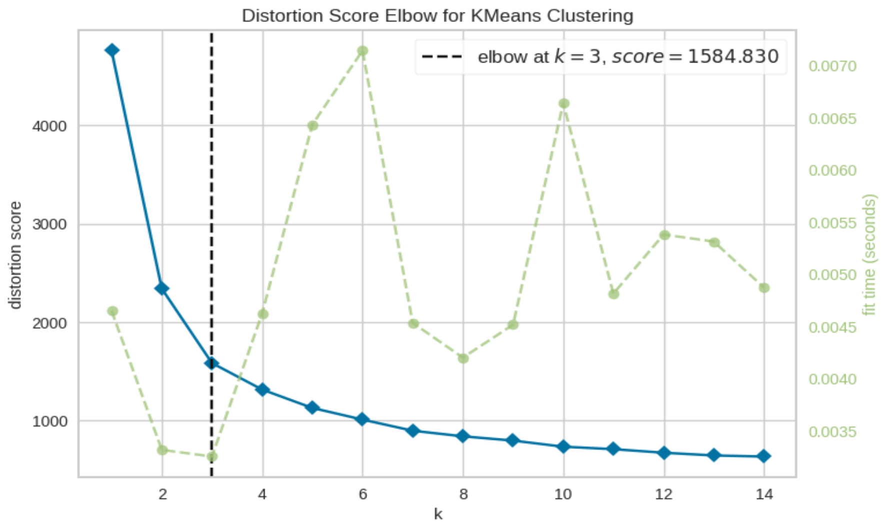

Fig-19

5.2. Let's check the silhouette scores:

```
For n_clusters = 2, the silhouette score is 0.4836571714922111)
For n_clusters = 3, the silhouette score is 0.4665236609739676)
For n_clusters = 4, the silhouette score is 0.4040931072281439)
For n_clusters = 5, the silhouette score is 0.4070983623658953)
For n_clusters = 6, the silhouette score is 0.40839879230248816)
For n_clusters = 7, the silhouette score is 0.39312884435988815)
For n_clusters = 8, the silhouette score is 0.36800806205696396)
For n_clusters = 9, the silhouette score is 0.35773768325622457)
For n_clusters = 10, the silhouette score is 0.34496887560165534)
For n_clusters = 11, the silhouette score is 0.34408141510921864)
For n_clusters = 12, the silhouette score is 0.33721609962712795)
For n_clusters = 13, the silhouette score is 0.33674923502462223)
For n_clusters = 14, the silhouette score is 0.33017194766090385)
```

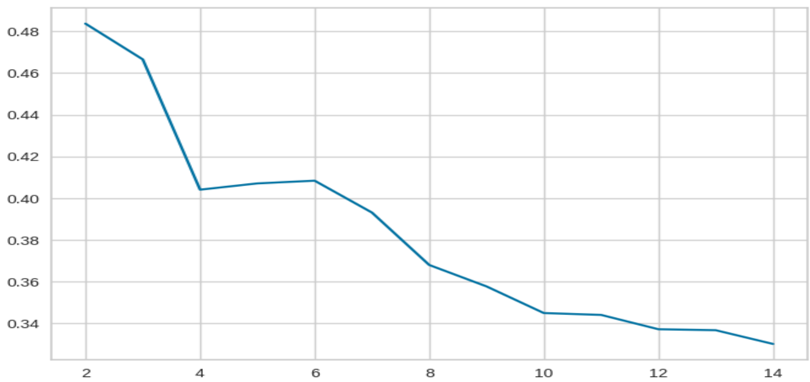

Fig-20

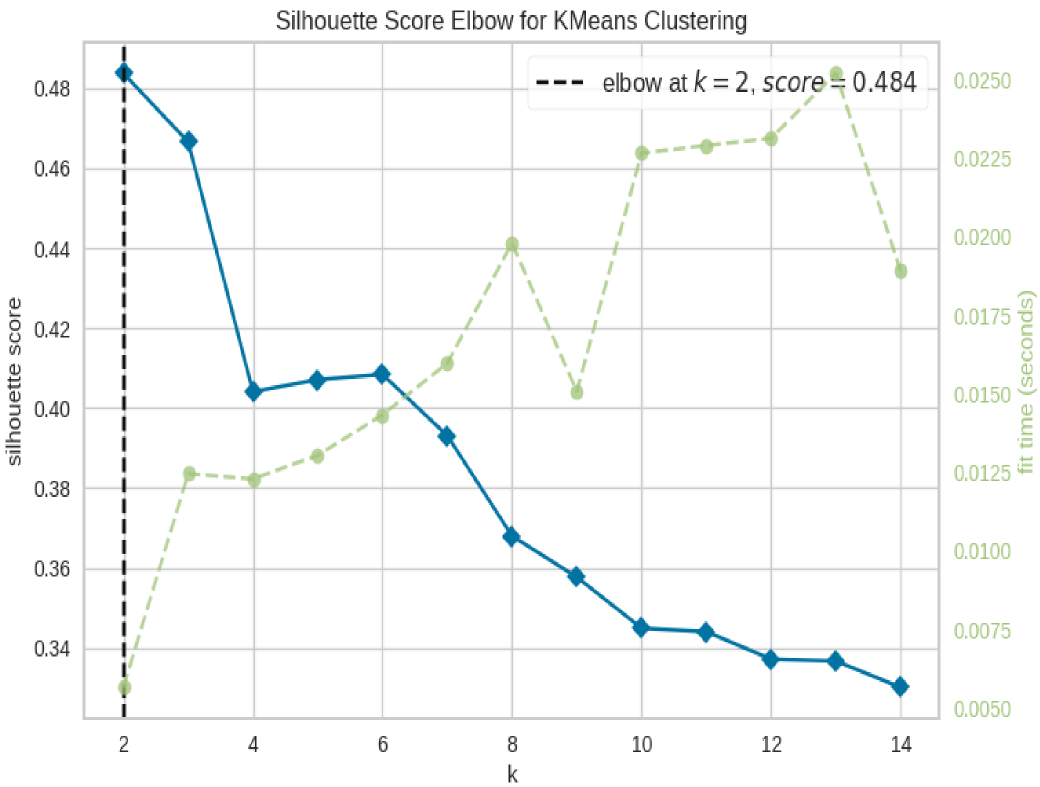

Fig-21

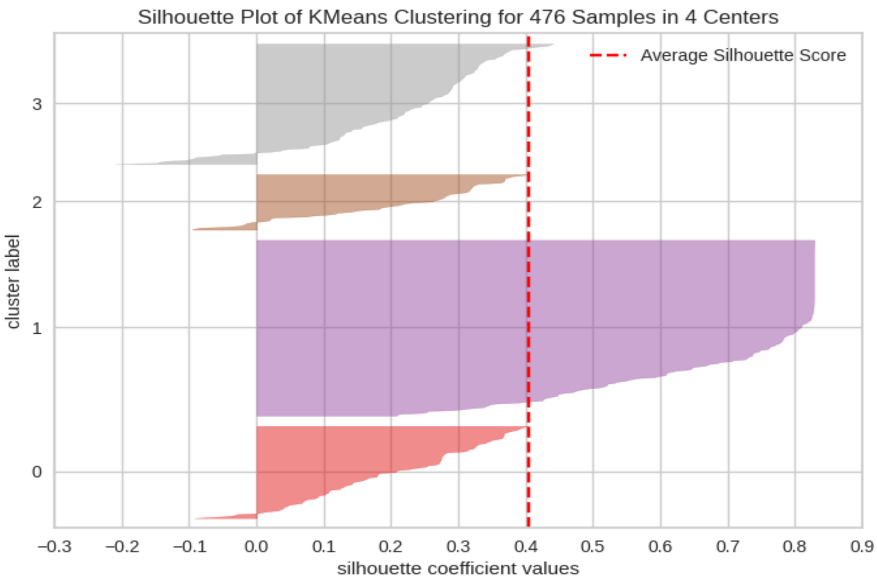

Fig-22

5.3. Creating Final Model:

```
KMeans
KMeans(n_clusters=4, random_state=1)
```

5.4. Cluster Profiling:

- Creating the "count_in_each_segment" feature in K-Means cluster profile.

KM_segments	Goals_Scored	Assists	Total_Points	Minutes	Goals_Conceded	Creativity	Influence	Threat	Bonus	Clean_Sheets	count_in_each_segment
0	1.363636	1.878788	103.525253	2670.555556	37.525253	265.671717	579.185859	199.636364	7.676768	10.020202	99
1	0.148936	0.202128	9.824468	238.750000	3.930851	28.171809	43.164894	30.244681	0.409574	0.558511	188
2	9.183333	6.716667	142.150000	2457.266667	33.516667	623.141667	664.133333	880.533333	16.266667	9.250000	60
3	1.503876	1.604651	56.038760	1392.736434	20.573643	188.358915	270.818605	223.255814	3.356589	4.705426	129

Table 7

- Finding the players in each cluster.

```
KM_segments  Position
0            Defender      47
             Forward       16
             Goalkeeper     3
             Midfielder    62
1            Defender      50
             Goalkeeper    17
             Midfielder    32
2            Defender       5
             Forward       20
             Midfielder    36
3            Defender      70
             Forward       28
             Goalkeeper    25
             Midfielder    65
Name: Player_Name, dtype: int64
```

Table 8

5.5. Let's plot the box-plot:

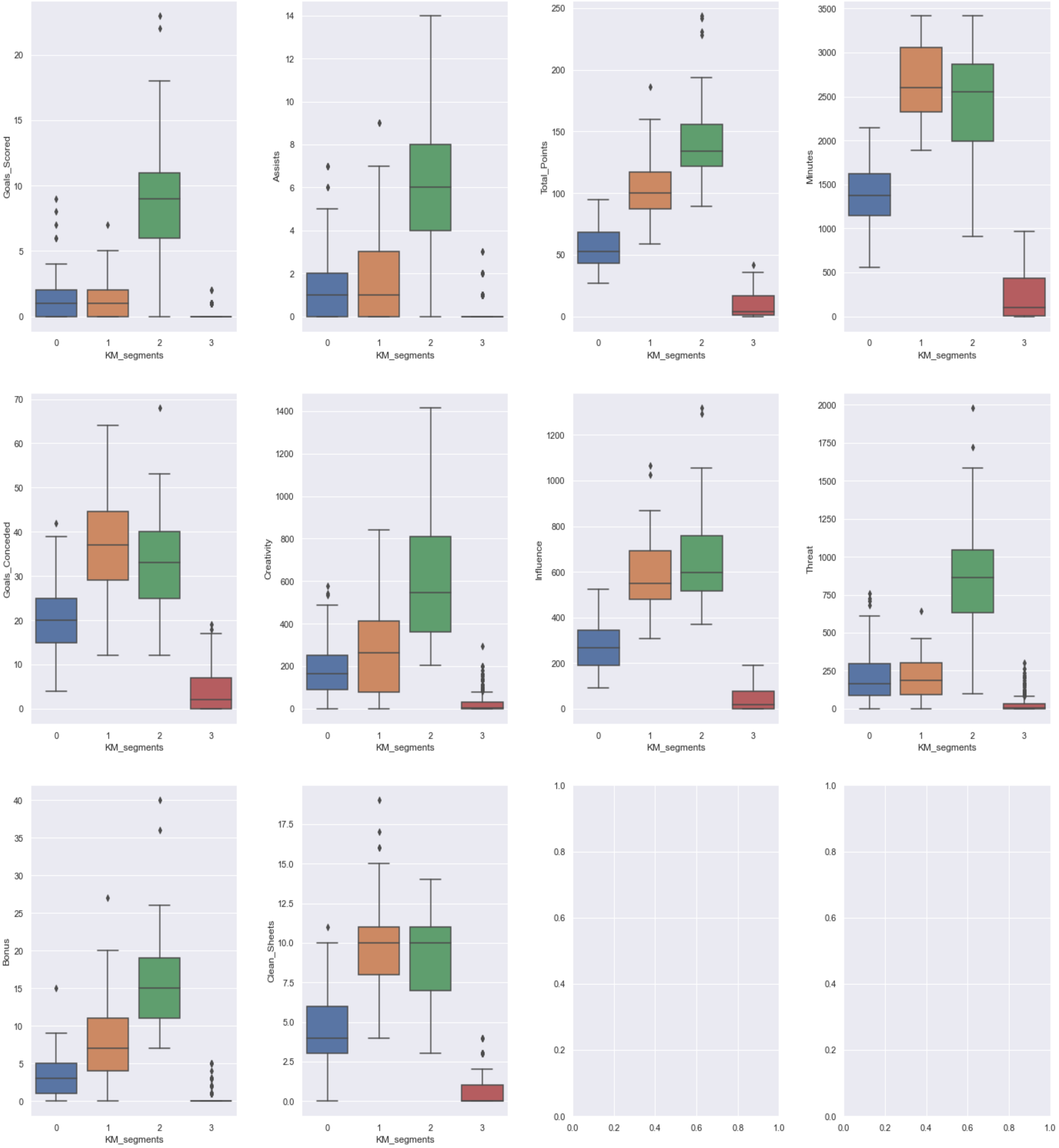

Fig-22

5.6. Characteristics of each cluster:

Cluster 0

- There are 128 players in this cluster.
- Most of the players in this cluster have a few goals and assists, and the total fantasy points scored in the previous season are low.
- Most of the players in this cluster had a moderate game time, a low creativity score, a low influence score, and a moderate threat score.
- Most of the players in this cluster received low bonus points.

Cluster 1

- There are 99 players in this cluster.
- Most of the players in this cluster have a few goals and assists, and the total fantasy points scored in the previous season are moderate.
- Most of the players in this cluster had a high game time, a moderate creativity score, a high influence score, and a moderate threat score.
- Most of the players in this cluster received moderate bonus points.

Cluster 2

- There are 61 players in this cluster.
- Most of the players in this cluster have lots of goals and assists, and the total fantasy points scored in the previous season are high.
- Most of the players in this cluster had a high game time, a high creativity, influence, and scores.
- Most of the players in this cluster received high bonus points.

Cluster 3

- There are 188 players in this cluster.
- Players in this cluster, except a few, have no goals and assists, and did not score any fantasy points scored in the previous season.
- Most of the players in this cluster had a low game time, and low creativity, influence, and threat scores.
- Players in this cluster, except a few, received no bonus points.

From this we can say that:

- **Cluster 2** are the **high value players** who have performed exceptionally well last season.
- **Cluster 1** are the **moderate value players** who have performed well last season.
- **Cluster 0** are the **low value players** who have performed poorly last season despite getting game time last season*.
- **Cluster 3** from the 0-low values and game time we can assume these are the **bench players** that don't get much game time through the season.

6. HIERARCHICAL CLUSTERING

Hierarchical Clustering is computationally more expensive, but potentially improves on K-means. Rather than centering on a mean of a pre set number of clusters, hierarchical clustering builds a hierarchy of clusters.

6.1. Computing Cophenetic Correlation:

```
Cophenetic correlation for Euclidean distance and single linkage is 0.8430175514228705.
Cophenetic correlation for Euclidean distance and complete linkage is 0.741204129226176.
Cophenetic correlation for Euclidean distance and average linkage is 0.8476499945585418.
Cophenetic correlation for Euclidean distance and weighted linkage is 0.862458135106748.
Cophenetic correlation for Chebyshev distance and single linkage is 0.8397660913391951.
Cophenetic correlation for Chebyshev distance and complete linkage is 0.8083029497725449.
Cophenetic correlation for Chebyshev distance and average linkage is 0.8590072179300738.
Cophenetic correlation for Chebyshev distance and weighted linkage is 0.8367206550474544.
Cophenetic correlation for Mahalanobis distance and single linkage is 0.8065008904132245.
Cophenetic correlation for Mahalanobis distance and complete linkage is 0.6583135946488975.
Cophenetic correlation for Mahalanobis distance and average linkage is 0.774780063243405.
Cophenetic correlation for Mahalanobis distance and weighted linkage is 0.6486408054242727.
Cophenetic correlation for Cityblock distance and single linkage is 0.8299646528677203.
Cophenetic correlation for Cityblock distance and complete linkage is 0.8493041408810342.
Cophenetic correlation for Cityblock distance and average linkage is 0.8127162760037657.
Cophenetic correlation for Cityblock distance and weighted linkage is 0.8553115836932642.
****************************************************************************************************
Highest cophenetic correlation is 0.862458135106748, which is obtained with Euclidean distance and weighted linkage.
```

- **Let's explore different linkage methods with Euclidean distance only.**

```
Cophenetic correlation for single linkage is 0.8430175514228706.
Cophenetic correlation for complete linkage is 0.7412041292261761.
Cophenetic correlation for average linkage is 0.8476499945585415.
Cophenetic correlation for centroid linkage is 0.8068296032280465.
Cophenetic correlation for ward linkage is 0.577773844586155.
Cophenetic correlation for weighted linkage is 0.8624581351067481.
****************************************************************************************************
Highest cophenetic correlation is 0.8624581351067481, which is obtained with weighted linkage.
```

- **Let's view the dendrograms for the different linkage methods with Euclidean distance only.**

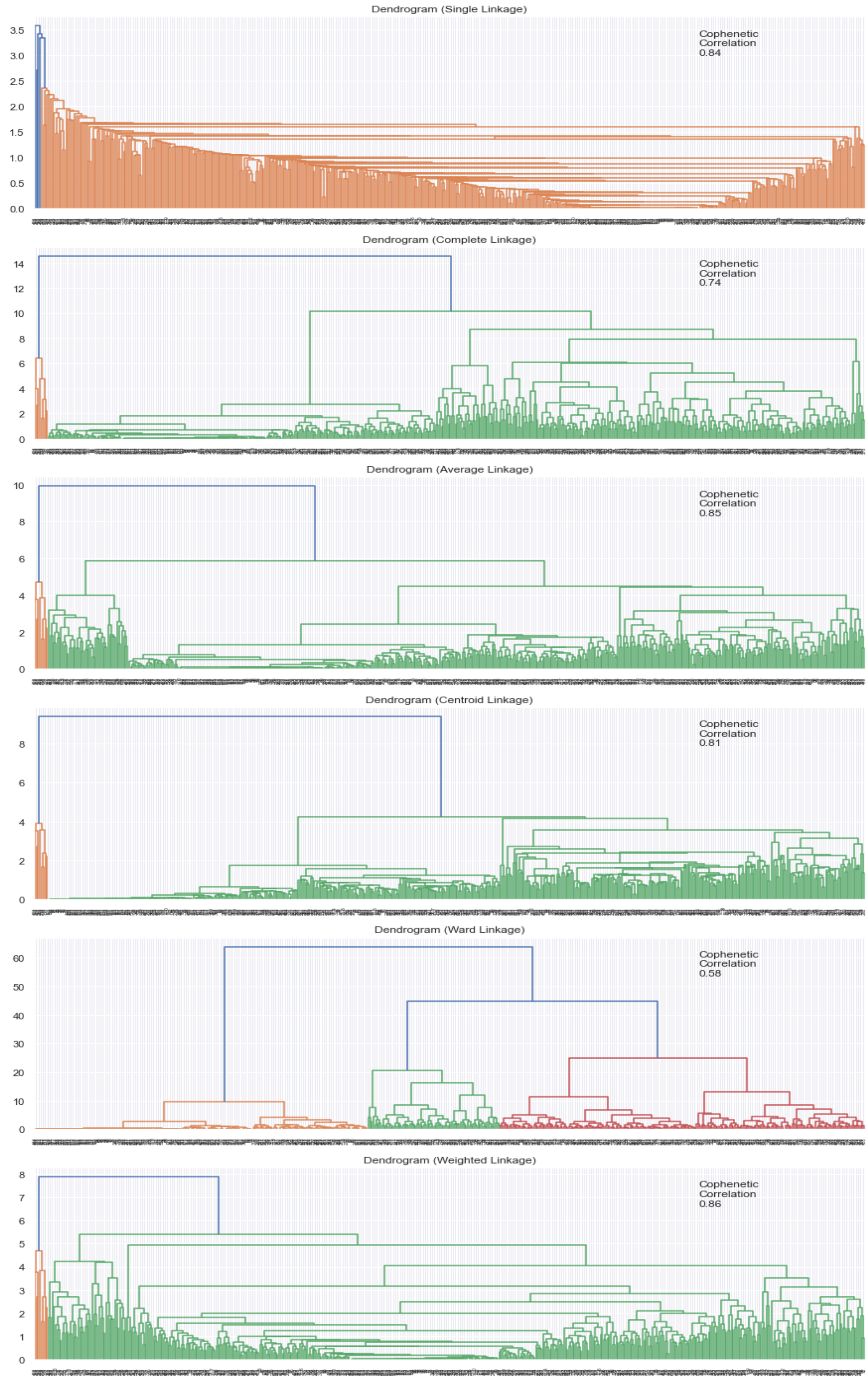

Fig-23

6.2. Creating and comparing cophenetic correlations for different linkage methods:

	Linkage	Cophenetic Coefficient
4	ward	0.577774
1	complete	0.741204
3	centroid	0.806830
0	single	0.843018
2	average	0.847650
5	weighted	0.862458

Table 9

6.3. Creating model using sklearn:

```
AgglomerativeClustering
AgglomerativeClustering(linkage='average', n_clusters=4)
```

6.4. Cluster Profiling:

- Creating the "count_in_each_segment" feature in Hierarchical Cluster profile.

HC_segments	Goals_Scored	Assists	Total_Points	Minutes	Goals_Conceded	Creativity	Influence	Threat	Bonus	Clean_Sheets	count_in_each_segment
0	0.881517	1.139810	47.969194	1205.945498	17.580569	148.574408	249.536967	131.753555	3.293839	4.182464	422
1	16.800000	9.200000	189.000000	3033.200000	44.000000	494.340000	860.720000	1591.600000	21.800000	10.800000	5
2	8.565217	5.826087	129.391304	2238.934783	29.760870	543.273913	586.234783	861.739130	14.021739	8.739130	46
3	19.333333	13.000000	238.000000	3101.000000	37.000000	1041.300000	1221.000000	1294.666667	34.000000	12.666667	3

Table 10

- Finding the players in each cluster.

HC_segments	Position	Player_Name
0	Defender	171
	Forward	43
	Goalkeeper	45
	Midfielder	163
1	Forward	4
	Midfielder	1
2	Defender	1
	Forward	16
	Midfielder	29
3	Forward	1
	Midfielder	2

dtype: int64

Table 11

5.5. Let's plot the box-plot:

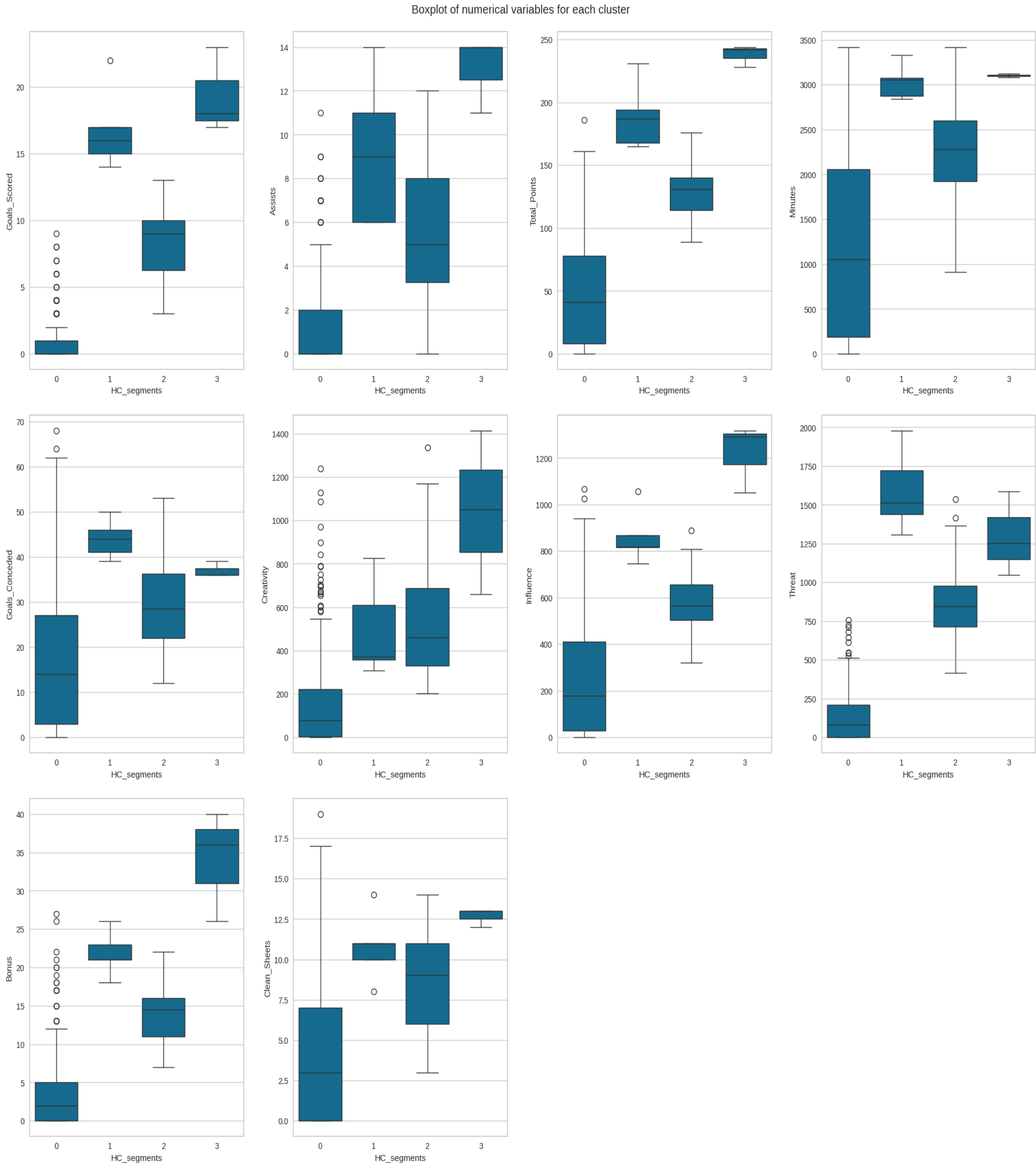

Fig-24

7. K-MEANS VS HIERARCHICAL CLUSTERING

Comparison of cluster profiles from Hierarchical and K-Means algorithms:

- The number of players in each cluster has changed, but the cluster profiles are largely the same.
- A slight change is it seems to be valuing offensive players slightly higher.

8. ACTIONABLE INSIGHTS & RECOMMENDATIONS

8.1. Choosing the Best Algorithm

Based on the silhouette score, we can see that K-Means algorithm is giving the best score on the data.

8.2. Insights

- The players who have a greater influence on the outcome of the game typically play for a longer duration in every game and score more fantasy points.
    - This is also likely arising from a primary analysis. Each team does analysis of their players and gives better players more game time. More game time also allows those players to score more fantasy points but is also a reflection of that initial analysis that they are better players.
- The players can be sold for more money whosoever have higher goals scored, creativity and influence.
- Since there is a drop at K = 4 in the elbow plot, we selected K as 4 for clustering.
    - This indicates that the clusters could possibly be separated into 5 groups if needed. However, from our analysis 4 groups seem to represent a distinct spread of players.
- We implemented 2 algorithms, but we have chosen K-Means algorithm as the final algorithm because it has the highest silhouette score of 0.40.

8.3. Recommendations:

- Cluster 0 players are the top players for fantasy. They fetch more points and have a higher chance of getting bonus points too. These players should be priced higher than the others so that it will be difficult to accommodate too many of them in the same team (because of the fixed budget) and fantasy managers have to make wise choices.
- Cluster 1 players are players who do not play many minutes, most likely come on as substitutes and fetch lesser fantasy points as a result. These players should be priced low and can be good differential picks.
- Cluster 2 are the players who are influential in their team's play but do not tend to score or assist much, resulting in lesser fantasy points than the Cluster 0 players. These players should be priced somewhere between the Cluster 0 and Cluster 1 players.
- Cluster 3 has the players who are in the squad to provide backup in case any of the starting 11 players get injured. They get lower game time and barely get any fantasy points. These players should be priced the lowest amongst the 4 clusters.
- Player performances from previous seasons should be taken into account and fantasy prices from the previous season should be referred to as a benchmark to determine the price for the upcoming season.
- OnSports should conduct cluster analysis separately for each of the playing positions to arrive at a better fantasy pricing strategy, given that football is heavily biased towards offensive players.