

Teman Icis

NBA Stats

Anggota Kelompok:

- | | | |
|--------------|--------------------------------|-----------|
| • 2006473421 | - Mohammad Rizky Chairul Azizi | (KASDD C) |
| • 2006484596 | - Zeta Prawira Syah | (KASDD C) |
| • 2006486001 | - Syahdan Putra Adriatama | (KASDD C) |
| • 2006530141 | - Rakha Rayhan Nusyura | (KASDD C) |

Doa Mohon diberi Kemudahan

رَبَّنَا إِاتِنَا مِنْ لَدُنْكَ رَحْمَةً وَهَيْئَةً لَنَا مِنْ

أَمْرِنَا رَشَدًا

Artinya: "Ya Tuhan kami, berikanlah rahmat kepada kami dari sisi-Mu dan sempurnakanlah bagi kami petunjuk yang lurus dalam urusan kami ini." (QS. Al-Kahfi: 10).

Summary

Pada proyek ini, kelompok kami melakukan exploratory data analysis (EDA) serta memecahkan masalah klasifikasi, regresi, dan clustering yang diberikan terkait data NBA.

Steps

Business Understanding

Data Understanding

Data Preprocessing

Joining Data

Data Modelling & Parameter Tuning

Evaluation



Data Understanding

Data Understanding

Datasets

Topik NBA Stats memiliki 6 datasets, diantaranya mencakup informasi mengenai MVPs, Nicknames, Player MVP Stats, Players, Salaries, dan Teams dari liga NBA.



Data Understanding (cont.)

- Pos: Position
- Age: Age
- Tm: Team
- G: Games Played
- GS: Games Started
- MP: Minutes Played
- FG: Field Goal Made
- FGA: Field Goal Attempts
- FG%: Field Goal Percentage
- 3P: 3-Point Made
- 3PA: 3-Point Attempts
- 3P%: 3-Point Percentage
- 2P: 2-Point Made
- 2PA: 2-Point Attempts
- 2P%: 2-Point Percentage
- eFG%: Effective Field Goal Percentage
- FT: Free Throw Made
- FTA: Free Throw Attempts
- FT%: Free Throw Percentage
- ORB: Offensive Rebounds
- DRB: Defensive Rebounds
- TRB: Total Rebounds
- AST: Assists
- STL: Steals
- BLK: Blocks
- TOV: Turnovers
- PF: Personal Fouls
- PTS: Points Made
- Year: Season Year
- W: Win
- L: Lose
- W/L%: Win/Lose Percentage



Data Understanding (cont.)

MVPs

```
[12] mvps_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 474 entries, 0 to 473
Data columns (total 21 columns):
 #   Column      Non-Null Count  Dtype  
 ---  --          --          --      
 0   Rank        474 non-null    int64  
 1   Player       474 non-null    object 
 2   Age          474 non-null    int64  
 3   Tm          474 non-null    object 
 4   First        474 non-null    int64  
 5   Pts Won     474 non-null    int64  
 6   Pts Max     474 non-null    int64  
 7   Share        474 non-null    float64
 8   G            474 non-null    int64  
 9   MP           474 non-null    float64
 10  PTS          474 non-null    float64
 11  TRB          474 non-null    float64
 12  AST          474 non-null    float64
 13  STL          474 non-null    float64
 14  BLK          474 non-null    float64
 15  FG%          474 non-null    float64
 16  3P%          474 non-null    float64
 17  FT%          474 non-null    float64
 18  WS           474 non-null    float64
 19  WS/48        474 non-null    float64
 20  Year         474 non-null    int64  
dtypes: float64(12), int64(7), object(2)
memory usage: 77.9+ KB
```

Nicknames

```
[18] nicknames_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 40 entries, 0 to 39
Data columns (total 2 columns):
 #   Column      Non-Null Count  Dtype  
 ---  --          --          --      
 0   Abbreviation 40 non-null    object 
 1   Name         40 non-null    object 
dtypes: object(2)
memory usage: 768.0+ bytes
```

Teams

```
[80] teams_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 906 entries, 0 to 1032
Data columns (total 9 columns):
 #   Column      Non-Null Count  Dtype  
 ---  --          --          --      
 0   W           906 non-null    int64  
 1   L           906 non-null    int64  
 2   W/L%        906 non-null    float64
 3   GB          738 non-null    float64
 4   PS/G        906 non-null    float64
 5   PA/G        906 non-null    float64
 6   SRS          906 non-null    float64
 7   Year         906 non-null    int64  
 8   Team         906 non-null    object 
dtypes: float64(5), int64(3), object(1)
memory usage: 70.8+ KB
```

Data Understanding (cont.)

Salaries

```
[39] salaries_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 37420 entries, 0 to 37419
Data columns (total 4 columns):
 #   Column    Non-Null Count  Dtype  
--- 
 0   Name       37420 non-null   object  
 1   Year        37420 non-null   int64  
 2   Salaries    9346 non-null   float64 
 3   Rank        37420 non-null   int64  
dtypes: float64(1), int64(2), object(1)
memory usage: 1.1+ MB
```

Player MVP Stats

```
[23] player_mvp_stats_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 14092 entries, 0 to 14091
Data columns (total 41 columns):
 #   Column    Non-Null Count  Dtype  
--- 
 0   Player     14092 non-null   object  
 1   Pos        14092 non-null   object  
 2   Age         14092 non-null   int64  
 3   Tm         14092 non-null   object  
 4   G          14092 non-null   int64  
 5   GS         14092 non-null   int64  
 6   MP         14092 non-null   float64 
 7   FG         14092 non-null   float64 
 8   FGA        14092 non-null   float64 
 9   FG%        14082 non-null   float64 
 10  3P        14092 non-null   float64 
 11  3PA       14092 non-null   float64 
 12  3P%       12050 non-null   float64 
 13  2P        14092 non-null   float64 
 14  2PA       14092 non-null   float64 
 15  2P%       14008 non-null   float64 
 16  eFG%      14082 non-null   float64 
 17  FT         14092 non-null   float64 
 18  FTA        14092 non-null   float64 
 19  FT%       13630 non-null   float64 
 20  ORB        14092 non-null   float64 
 21  DRB        14092 non-null   float64 
 22  TRB        14092 non-null   float64 
 23  AST        14092 non-null   float64 
 24  STL        14092 non-null   float64 
 25  BLK        14092 non-null   float64 
 26  TOV        14092 non-null   float64 
 27  PF         14092 non-null   float64 
 28  PTS        14092 non-null   float64 
 29  Year       14092 non-null   int64  
 30  Pts Won    14092 non-null   int64  
 31  Pts Max    14092 non-null   int64  
 32  Share      14092 non-null   float64 
 33  Team       14092 non-null   object  
 34  W          14092 non-null   int64  
 35  L          14092 non-null   int64  
 36  W/L%      14092 non-null   float64 
 37  GB         14092 non-null   float64 
 38  PS/G      14092 non-null   float64 
 39  PG/G      14092 non-null   float64 
 40  SRS        14092 non-null   float64 
dtypes: float64(29), int64(8), object(4)
memory usage: 4.4+ MB
```

Players

```
[30] players_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 18044 entries, 0 to 18043
Data columns (total 31 columns):
 #   Column    Non-Null Count  Dtype  
--- 
 0   Rk         18044 non-null   object  
 1   Player     18044 non-null   object  
 2   Pos        18044 non-null   object  
 3   Age         18044 non-null   object  
 4   Tm         18044 non-null   object  
 5   G          18044 non-null   object  
 6   GS         18044 non-null   object  
 7   MP         18044 non-null   object  
 8   FG         18044 non-null   object  
 9   FGA        18044 non-null   object  
 10  FG%        18044 non-null   object  
 11  3P         18044 non-null   object  
 12  3PA       18044 non-null   object  
 13  3P%       18044 non-null   object  
 14  2P         18044 non-null   object  
 15  2PA       18044 non-null   object  
 16  2P%       18044 non-null   object  
 17  eFG%      18044 non-null   object  
 18  FT         18044 non-null   object  
 19  FTA        18044 non-null   object  
 20  FT%       18044 non-null   object  
 21  ORB        18044 non-null   object  
 22  DRB        18044 non-null   object  
 23  TRB        18044 non-null   object  
 24  AST        18044 non-null   object  
 25  STL        18044 non-null   object  
 26  BLK        18044 non-null   object  
 27  TOV        18044 non-null   object  
 28  PF         18044 non-null   object  
 29  PTS        18044 non-null   object  
 30  Year       18044 non-null   int64  
dtypes: int64(1), object(30)
```



Problem Set 1

Exploratory Data Analysis

Pengerjaan Problem Set 1 - EDA

- a. Apakah yang menjadi faktor utama (secara statistik) seorang pemain menjadi MVP?

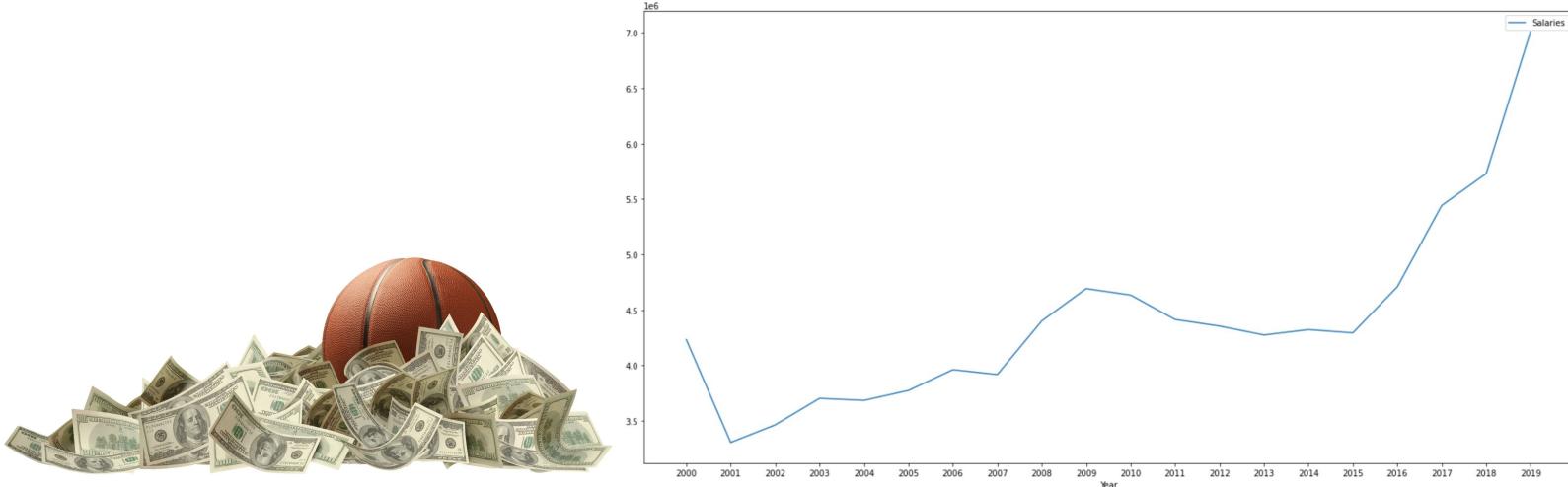
Asumsi kami untuk player yang menjadi MVP adalah ia banyak memberikan poin pada suatu game. Maka kami mencari korelasi tertinggi dengan kolom PTS (Points Made) pada dataset player_mvp_stats. Tiga kolom yang paling berkorelasi tinggi adalah MP, FG, dan FGA. FG (Field Goal) dan FGA (Field Goal Attempts) sudah pasti berkorelasi tinggi dengan PTS. Maka, yang paling cocok untuk menjadi faktor utama seorang player menjadi MVP adalah **Minutes Played**.

TOV	-0.01	0.41	0.67	0.81	0.82	0.83	0.12	0.32	0.34
PF	0.03	0.50	0.59	0.70	0.59	0.56	0.22	0.09	0.08
PTS	0.02	0.48	0.74	0.89	0.99	0.98	0.16	0.50	0.50
W	0.18	0.15	0.06	-0.02	0.02	-0.02	0.05	0.04	0.01
L	-0.18	-0.04	-0.02	0.02	-0.01	0.02	-0.04	-0.06	-0.04
W/L%	0.08	0.08	0.03	-0.02	0.02	-0.01	0.03	0.05	0.04
GB	-0.15	-0.06	-0.03	0.02	-0.01	0.02	-0.03	-0.07	-0.05
PS/G	-0.05	-0.02	-0.02	-0.05	0.05	0.01	0.06	0.19	0.18
PA/G	-0.17	-0.09	-0.05	-0.04	0.04	0.03	0.02	0.15	0.16
SRS	0.18	0.11	0.04	-0.02	0.02	-0.02	0.05	0.05	0.03
	Age	G	GS	MP	FG	FGA	FG%	3P	3PA

Pengerjaan Problem Set 1 - EDA (cont.)

b. Apakah terdapat kenaikan rata-rata gaji pemain dari tahun 2000 - 2019?

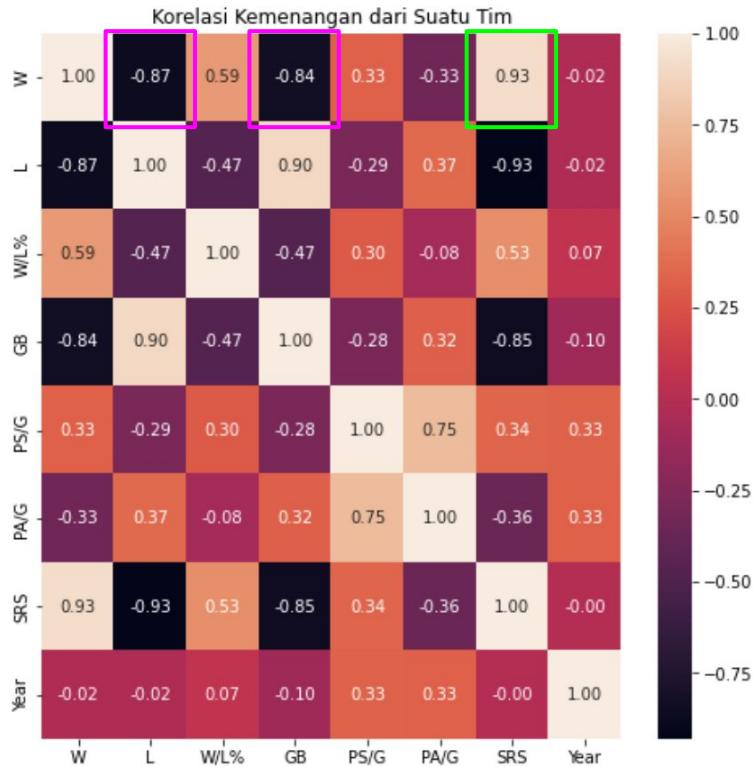
Kami melakukan visualisasi rata-rata gaji semua pemain tiap tahunnya dari data Salaries. Hasil tersebut menunjukkan bahwa terdapat beberapa tahun yang mengalami penurunan rerata gaji pemain, yaitu 2001, 2004, 2007, 2010 - 2015. Namun, bila dibandingkan gaji tahun **2000** dengan tahun **2019**, terdapat kenaikan rata-rata gaji pemain yang cukup signifikan.



Pengerjaan Problem Set 1 - EDA (cont.)

- c. Apakah kemenangan dari suatu tim dapat dikorelasikan dengan suatu variabel tertentu?

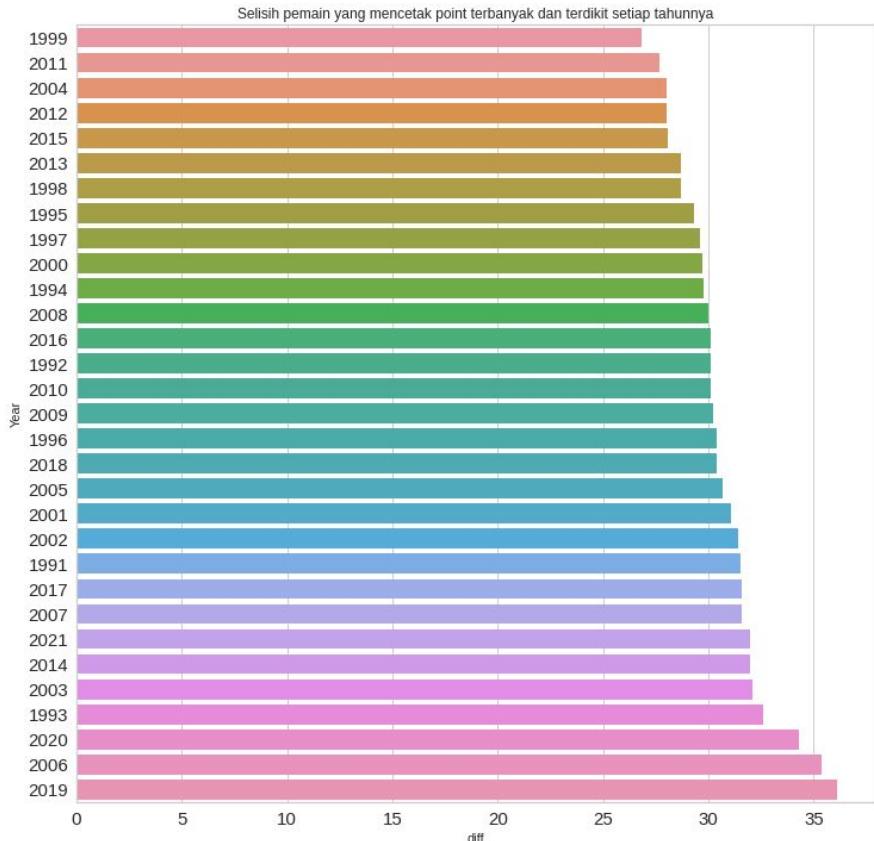
Menggunakan heatmap correlation, kami menemukan korelasi antara kolom **W** (**Win**) dengan variabel lainnya yang dilihat berdasarkan angka tertinggi (korelasi positif) maupun angka terendah (korelasi negatif). Dapat disimpulkan bahwa kemenangan suatu tim berkorelasi positif dengan **SRS (Simple Rating System)** sebesar 0.93 serta berkorelasi negatif dengan **L (Lose)** dan **GB (Games Behind)** berturut-turut sebesar -0.87 dan -0.84.



Pengerjaan Problem Set 1 - EDA (cont.)

- d. Tahun berapakah yang merupakan tahun yang paling kompetitif untuk liga NBA?

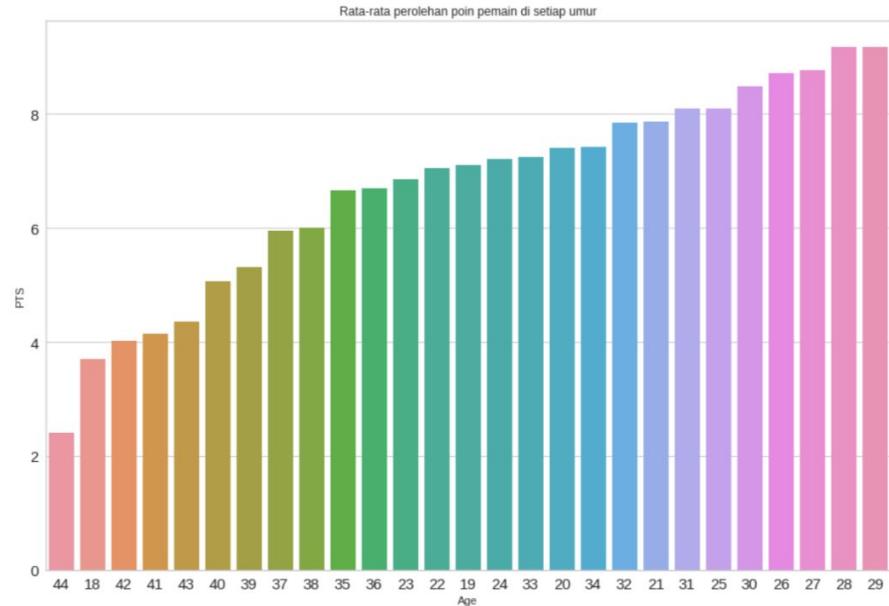
Dari hasil visualisasi di samping, didapatkan tahun yang paling kompetitif berdasarkan selisih terkecil antara pemain yang mencetak poin tertinggi dan terendah setiap tahunnya adalah **tahun 1999**. Ini berarti pada tahun 1999, range perolehan poin setiap pemain adalah yang paling kecil dibanding tahun lainnya sehingga tingkat kompetitif pemain tergolong tinggi pada tahun tersebut.



Pengerjaan Problem Set 1 - EDA (cont.)

- e. Pada kisaran umur berapa pemain paling banyak memberikan poin?

Kami melakukan visualisasi untuk dapat membandingkan rata-rata poin dengan masing-masing umur pemain. Dapat dilihat bahwa **umur 28 dan 29 menjadi umur yang paling banyak memberikan point.**

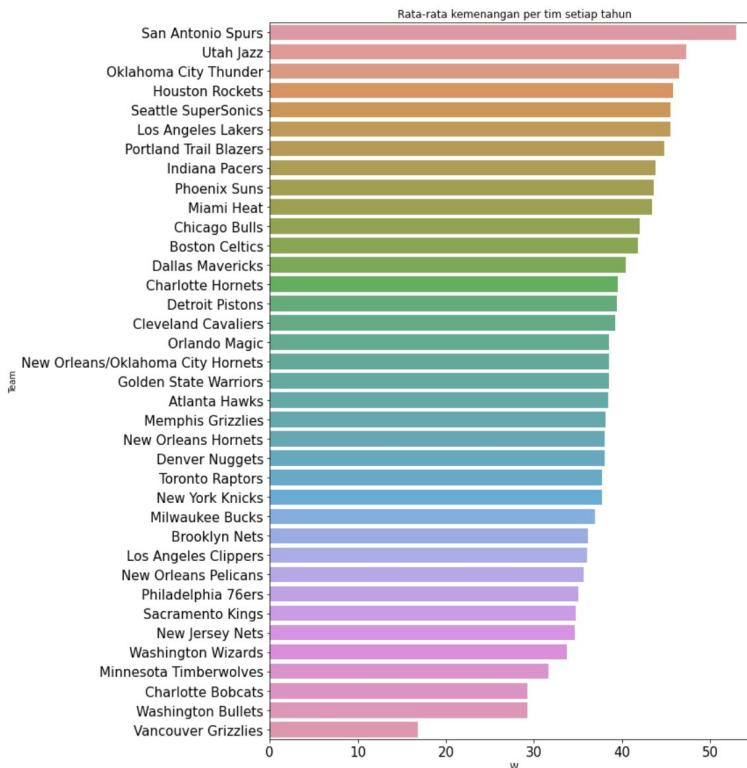


Pengerjaan Problem Set 1 - EDA (cont.)

- f. Tim manakah yang merupakan tim 'terbaik' berdasarkan rata-rata kemenangan per tim setiap tahunnya?

Kami telah menghitung rata-rata kemenangan per tim setiap tahun yang ada pada dataset Teams, lalu memvisualisasikan hasilnya dengan menggunakan bar chart.

Berdasarkan visualisasi di samping, tim **San Antonio Spurs** adalah tim yang 'terbaik' dengan rata-rata kemenangan lebih dari 50 setiap tahunnya.





Preprocessing

Preprocessing – Data anomali

Players

```
players_df[pd.to_numeric(players_df["PF"], errors='coerce').isnull()]
```

	Rk	Player	Pos	Age	Tm	G	GS	MP	FG	FGA	...	ORB	DRB	TRB	AST	STL	BLK	TOV	PF	PTS	Year
47	Rk	Player	Pos	Age	Tm	G	GS	MP	FG	FGA	...	ORB	DRB	TRB	AST	STL	BLK	TOV	PF	PTS	1991
70	Rk	Player	Pos	Age	Tm	G	GS	MP	FG	FGA	...	ORB	DRB	TRB	AST	STL	BLK	TOV	PF	PTS	1991
93	Rk	Player	Pos	Age	Tm	G	GS	MP	FG	FGA	...	ORB	DRB	TRB	AST	STL	BLK	TOV	PF	PTS	1991
118	Rk	Player	Pos	Age	Tm	G	GS	MP	FG	FGA	...	ORB	DRB	TRB	AST	STL	BLK	TOV	PF	PTS	1991
145	Rk	Player	Pos	Age	Tm	G	GS	MP	FG	FGA	...	ORB	DRB	TRB	AST	STL	BLK	TOV	PF	PTS	1991
...	
17908	Rk	Player	Pos	Age	Tm	G	GS	MP	FG	FGA	...	ORB	DRB	TRB	AST	STL	BLK	TOV	PF	PTS	2021
17931	Rk	Player	Pos	Age	Tm	G	GS	MP	FG	FGA	...	ORB	DRB	TRB	AST	STL	BLK	TOV	PF	PTS	2021
17960	Rk	Player	Pos	Age	Tm	G	GS	MP	FG	FGA	...	ORB	DRB	TRB	AST	STL	BLK	TOV	PF	PTS	2021
17989	Rk	Player	Pos	Age	Tm	G	GS	MP	FG	FGA	...	ORB	DRB	TRB	AST	STL	BLK	TOV	PF	PTS	2021
18017	Rk	Player	Pos	Age	Tm	G	GS	MP	FG	FGA	...	ORB	DRB	TRB	AST	STL	BLK	TOV	PF	PTS	2021

658 rows x 31 columns

Handling:

- Men-drop baris-baris yang bersangkutan kemudian meng-convert kolom menjadi tipe yang seharusnya.

Ditemukan label kolom pada beberapa row di dataset Players yang mengakibatkan setiap kolom pada dataset ini bertipe object

Preprocessing (cont.) – Data anomali

Teams

	W	L	W/L%	GB	PS/G	PA/G	SRS	Year	Team	🔗
6	Central Division	1991	Central Division							
14	Midwest Division	1991	Midwest Division							
22	Pacific Division	1991	Pacific Division							
37	Central Division	1992	Central Division							
45	Midwest Division	1992	Midwest Division							
...
1003	Central Division	2021	Central Division							
1009	Southeast Division	2021	Southeast Division							
1015	Northwest Division	2021	Northwest Division							
1021	Pacific Division	2021	Pacific Division							
1027	Southwest Division	2021	Southwest Division							

127 rows x 9 columns

Handling:

- Men-drop baris-baris yang bersangkutan kemudian meng-convert kolom menjadi tipe yang seharusnya.

Ditemukan hal yang serupa pada dataset Teams. Nilai pada kolom team ter-duplicate ke kolom-kolom sebelahnya yang mengakibatkan setiap kolom lainnya pada dataset ini bertipe object.



Preprocessing (cont.) – Penyesuaian kolom numerik

MVPs

```
[ ] mvps_df[pd.to_numeric(mvps_df["Rank"], errors='coerce').isnull()]
```

Rank	Player	Age	Tm	First	Pts Won	Pts Max	Share	G	MP	...	TRB	AST	STL	BLK	FG%	3P%	FT%	WS	WS/48	Year	
8	9T	Larry Bird	34	BOS	0	25	960	26.0	60	38.0	...	8.5	7.2	1.8	1.0	454.00	389.0	891.0	6.6	0.14	1991
9	9T	Terry Porter	27	POR	0	25	960	26.0	81	32.9	...	3.5	8.0	2.0	0.1	515.00	415.0	823.0	13.0	235.00	1991
18	19T	Tim Hardaway	24	GSW	0	1	960	1.0	82	39.2	...	4.0	9.7	2.6	0.1	476.00	385.0	803.0	9.9	148.00	1991
19	19T	Kevin McHale	33	BOS	0	1	960	1.0	68	30.4	...	7.1	1.9	0.4	2.1	553.00	405.0	829.0	7.9	182.00	1991
31	12T	Charles Barkley	28	PHI	0	18	960	19.0	75	38.4	...	11.1	4.1	1.8	0.6	552.00	234.0	695.0	12.3	205.00	1992
...	
445	11T	Rudy Gobert	26	UTA	0	1	1010	1.0	81	31.8	...	12.9	2.0	0.8	2.3	669.00	0.0	636.0	14.4	268.00	2019
446	11T	LeBron James	34	LAL	0	1	1010	1.0	55	35.2	...	8.5	8.3	1.3	0.6	0.51	339.0	665.0	7.2	179.00	2019
471	13T	James Harden	31	TOT	0	1	1010	1.0	44	36.6	...	7.9	10.8	1.2	0.8	466.00	362.0	861.0	7.0	208.00	2021
472	13T	LeBron James	36	LAL	0	1	1010	1.0	45	33.4	...	7.7	7.8	1.1	0.6	513.00	365.0	698.0	5.6	179.00	2021
473	13T	Kawhi Leonard	29	LAC	0	1	1010	1.0	52	34.1	...	6.5	5.2	1.6	0.4	512.00	398.0	885.0	8.8	238.00	2021

122 rows x 21 columns

Handling:

- Menghapus akhiran T pada row yang bersangkutan dan meng-convert kolom tersebut menjadi bertipe numerik

Pada dataset MVPs, ditemukan bahwa player yang memiliki rank yang sama akan memiliki akhiran 'T' di kolom Rank. Hal ini mengakibatkan kolom tersebut menjadi bertipe object dan dapat menyulitkan proses analisis maupun modeling nantinya.

Preprocessing (cont.)

- Checking & Handling Missing Values

```
[ ] null_check(mvps_df)
```

Tidak ditemukan missing value pada dataset

```
[ ] null_check(nicknames_df)
```

Tidak ditemukan missing value pada dataset

```
[ ] null_check(player_mvp_stats_df)
```

	Total	Percent
eFG%	50	0.003548
FG%	50	0.003548
2P%	84	0.005961
FT%	462	0.032785
3P%	2042	0.144905

```
[ ] null_check(players_df)
```

Tidak ditemukan missing value pada dataset

```
[ ] null_check(salaries_df)
```

	Total	Percent
Salaries	28074	0.750241

```
[ ] null_check(teams_df)
```

	Total	Percent
GB	168	0.18543

handling



```
[ ] player_mvp_stats_df.dropna(inplace = True)  
null_check(player_mvp_stats_df)
```

Tidak ditemukan missing value pada dataset

```
[ ] salaries_df.dropna(inplace = True)  
null_check(salaries_df)
```

Tidak ditemukan missing value pada dataset

```
[ ] teams_df['GB'] = teams_df['GB'].fillna(0)  
null_check(teams_df)
```

Tidak ditemukan missing value pada dataset

- Pada datasets MVPs, Nicknames, dan Players tidak ditemukan missing values.
- Pada datasets **Player MVP Stats** dan **Salaries**, **missing values-nya di-drop**.
- Pada dataset Teams, missing value pada **kolom GB diisi dengan nilai 0** dengan asumsi tim tersebut berada di posisi pertama sehingga tidak memiliki nilai 'Games Back'.

Preprocessing (cont.)

- Checking & Handling Duplicate Values

```
Jumlah Nilai Duplikat pada mvps: 0
Jumlah Nilai Duplikat pada nicknames: 0
Jumlah Nilai Duplikat pada player_mvp_stats: 0
Jumlah Nilai Duplikat pada players: 0
Jumlah Nilai Duplikat pada salaries: 0
Jumlah Nilai Duplikat pada teams: 0
```

Tidak ditemukan duplicate values.

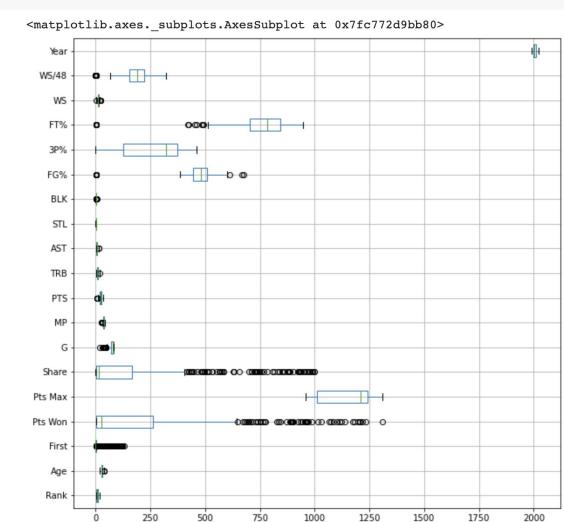


Preprocessing (cont.) - Checking & Handling Outliers

MVPs

Jumlah outlier pada dataset mvps di masing-masing fitur adalah

```
Rank      0
Age       3
First     83
Pts Won   57
Pts Max   0
Share    71
G        16
MP       9
PTS      6
TRB      1
AST      2
STL      0
BLK      21
FG%      50
3P%      0
FT%      61
WS       6
WS/48    46
Year     0
dtype: int64
```



Handling:

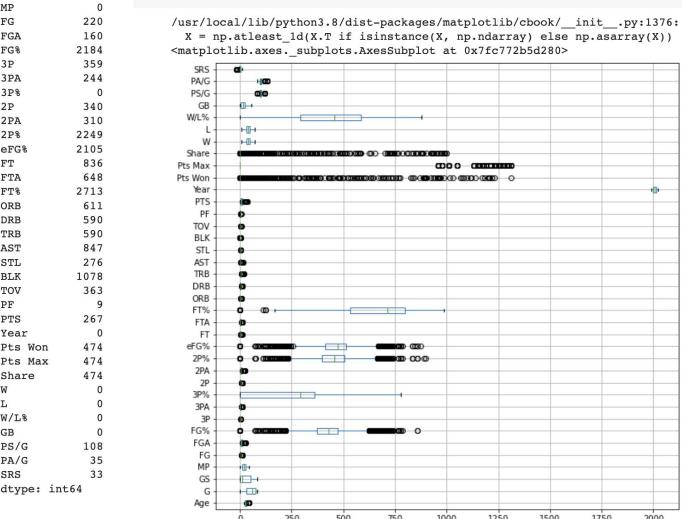
- Men-drop baris yang mengandung outlier pada kolom FT%, Share, Pts Won, First.
- Membiarkan outlier pada kolom lain karena diasumsikan terdistribusi secara normal.

```
mvps_df shape before and after drop outlier
(474, 21)
(322, 21)
```

Player MVP Stats

Jumlah outlier pada dataset player_mvp_stats di masing-masing fitur adalah

```
Age      11
G       0
GS      0
MP      0
FG     220
FGA    160
FG%    2184
3P     359
3PA    244
3P%    0
ZP     340
ZPA    310
ZP%    2249
eFG%   2105
FT     836
FTA    648
FT%    2713
ORB    611
DRB    590
TRB    590
AST    847
STL    276
BLK    1078
TOV    363
PF     9
PTS    267
PTS%   0
Year   0
Pts Won 474
Pts Max 474
Share  474
W      0
L      0
W/L%   0
GB     0
PS/G   108
PA/G   35
SRS    33
dtype: int64
```



Handling:

- Men-drop baris yang mengandung outlier pada kolom Share, Pts Won, eFG%, 2P%, FG% .
- Membiarkan outlier pada kolom lain karena diasumsikan terdistribusi secara normal.

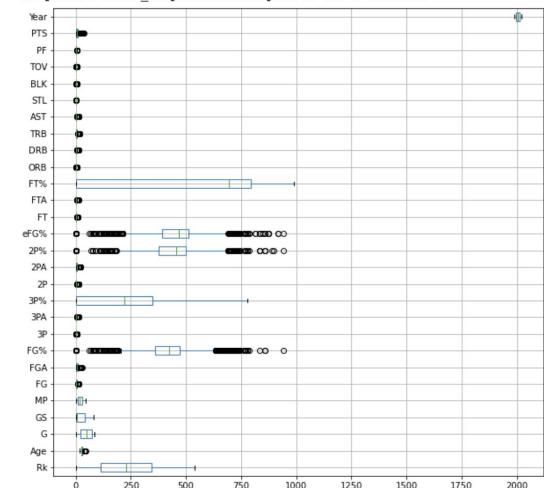
```
player_mvp_stats_df shape before and after drop outlier
(11817, 41)
(7790, 41)
```

Preprocessing (cont.) - Checking & Handling Outliers

Players

Jumlah outlier pada dataset players di masing-masing fitur adalah

```
Rk          0
Age         30
G           0
GS          0
MP          0
FG          314
FGA         267
FG%         2918
3P          416
3PA         340
3P%         0
2P          458
2PA         453
2P%         2979
eFG%        2844
FT          857
FTA         896
FT%         0
ORB         689
DRB         674
TRB         696
AST         1015
STL         313
BLK         1211
TOV         413
PF          24
PTS         352
Year        0
Rk          0
dtype: int64
```



Handling:

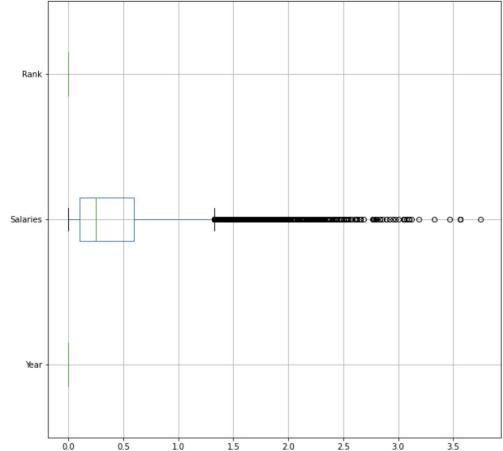
- Men-drop baris yang mengandung outlier pada kolom eFG%, 2P%, FG%.
- Membiarkan outlier pada kolom lain karena diasumsikan terdistribusi secara normal.

```
players_df shape before and after drop outlier
(17386, 31)
(11607, 31)
```

Salaries

Jumlah outlier pada dataset salaries di masing-masing fitur adalah

```
Year          0
Salaries      705
Rank          0
dtype: int64
```



Handling:

- Men-drop baris yang mengandung outlier pada kolom Salaries .

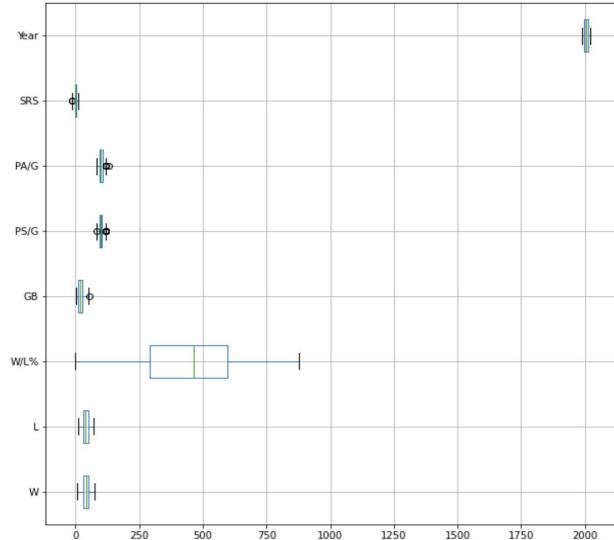
```
salaries_df shape before and after drop outlier
(9346, 4)
(8641, 4)
```

Preprocessing (cont.) - Checking & Handling Outliers

Teams

Jumlah outlier pada dataset teams di masing-masing fitur adalah

```
W      0
L      0
W/L%   0
GB     1
PS/G   9
PA/G   4
SRS    2
Year   0
dtype: int64
```



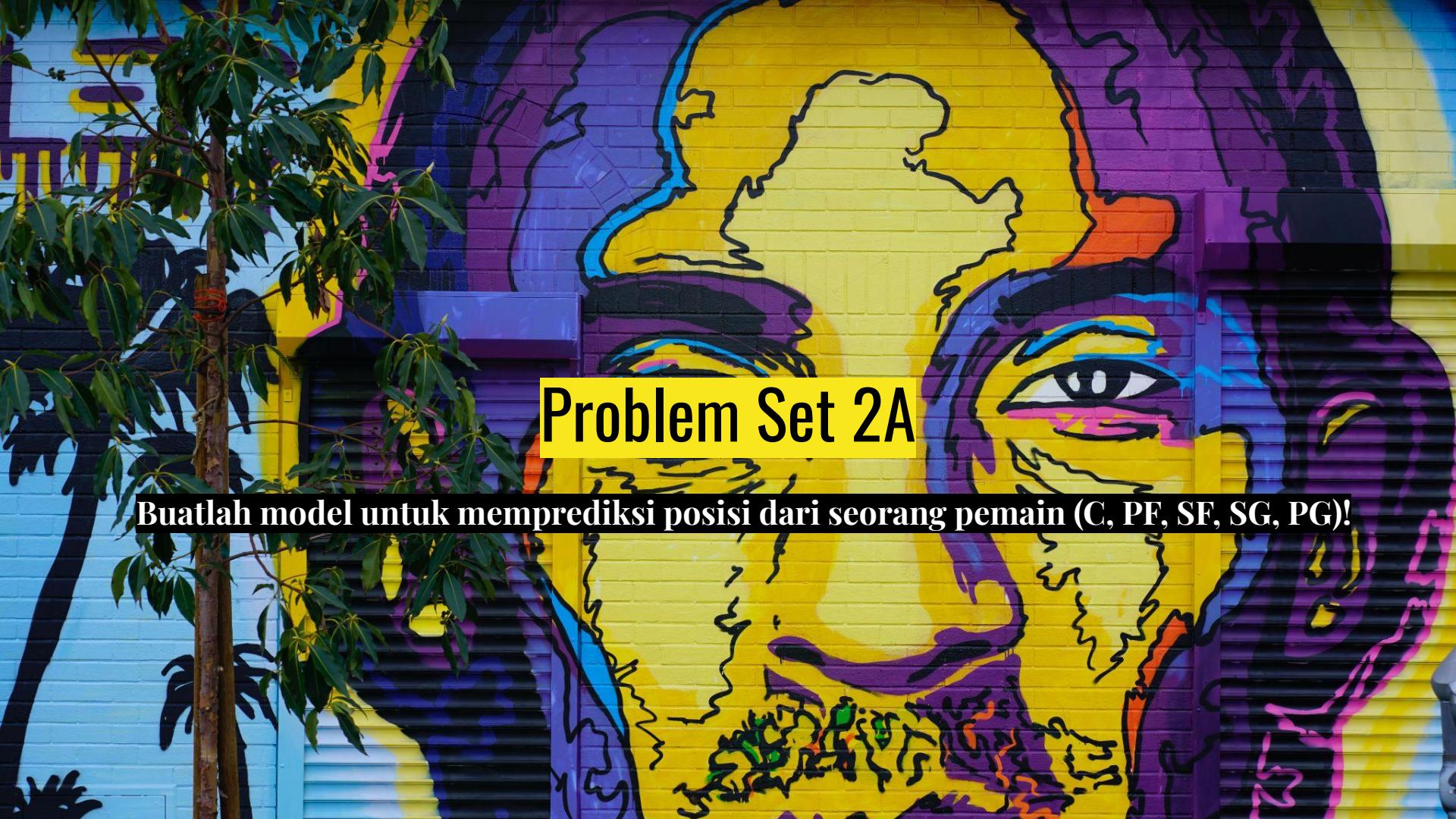
Handling:

- Membiarakan outlier pada semua kolom karena diasumsikan terdistribusi secara normal.

Nicknames

Dataset Nicknames hanya terdiri atas kolom kategorikal sehingga tidak mungkin memiliki outlier.





Problem Set 2A

Buatlah model untuk memprediksi posisi dari seorang pemain (C, PF, SF, SG, PG)!

Pengerjaan Problem Set 2A

A. Buatlah model untuk memprediksi posisi dari seorang pemain (C, PF, SF, SG, PG)!

Permasalahan di atas dapat diselesaikan menggunakan pemodelan **klasifikasi**. Berikut langkah-langkah yang perlu dilakukan:

- Preparation
- Modeling
 - Softmax Regression
 - Decision Tree
 - Random Forest
 - K-nearest Neighbors
- Conclusion

Pengerjaan Problem Set 2A - Preparation

- **Preparation**

- Menggunakan dataset Players yang sudah diproses lalu hanya memilih yang posisinya C, PF, SF, SG, atau PG.
- Men-drop kolom-kolom yang sekiranya redundan dan tidak relevan terhadap penentuan posisi pemain.
- Menyeleksi secara iteratif fitur-fitur yang akan di-keep menggunakan *feature selection*.
- Splitting data menjadi training dan testing.
- Scaling data fitur untuk model yang algoritmanya berbasis *distance* maupun *gradient descent*.

Pengerjaan Problem Set 2A - Modeling

- **Modeling**
 - Menerapkan cross validation pada setiap model untuk mendapatkan model terbaik.
 - Metrik yang digunakan yaitu Micro F1-Score.
 - CV sebanyak 5-fold terhadap data training.
 - Percobaan dilakukan pada model-model berikut:
 - Softmax Regression
 - Decision Tree
 - Random Forest
 - K-nearest Neighbors

Pengerjaan Problem Set 2A - Modeling (cont.)

- **Softmax Regression**

Hasil evaluasi:

```
---Softmax Regression Scoring---
Fold #1 Micro F1-Score: 0.619541
Fold #2 Micro F1-Score: 0.647926
Fold #3 Micro F1-Score: 0.621725
Fold #4 Micro F1-Score: 0.632642
Fold #5 Micro F1-Score: 0.615721
Average Micro F1-Score: 0.627511
```

- **Decision Tree**

Hasil evaluasi:

```
---Decision Tree Scoring---
Fold #1 Micro F1-Score: 0.568231
Fold #2 Micro F1-Score: 0.599891
Fold #3 Micro F1-Score: 0.558952
Fold #4 Micro F1-Score: 0.593886
Fold #5 Micro F1-Score: 0.584607
Average Micro F1-Score: 0.581114
```

Pengerjaan Problem Set 2A - Modeling (cont.)

- **Random Forest**

Hasil evaluasi:

```
---Random Forest Scoring---
Fold #1 Micro F1-Score: 0.634825
Fold #2 Micro F1-Score: 0.664847
Fold #3 Micro F1-Score: 0.619541
Fold #4 Micro F1-Score: 0.632096
Fold #5 Micro F1-Score: 0.617358
Average Micro F1-Score: 0.633734
```

- **K-nearest Neighbors**

Hasil evaluasi:

```
---K-nearest Neighbors Scoring---
Fold #1 Micro F1-Score: 0.603712
Fold #2 Micro F1-Score: 0.610808
Fold #3 Micro F1-Score: 0.603166
Fold #4 Micro F1-Score: 0.600983
Fold #5 Micro F1-Score: 0.588428
Average Micro F1-Score: 0.601419
```

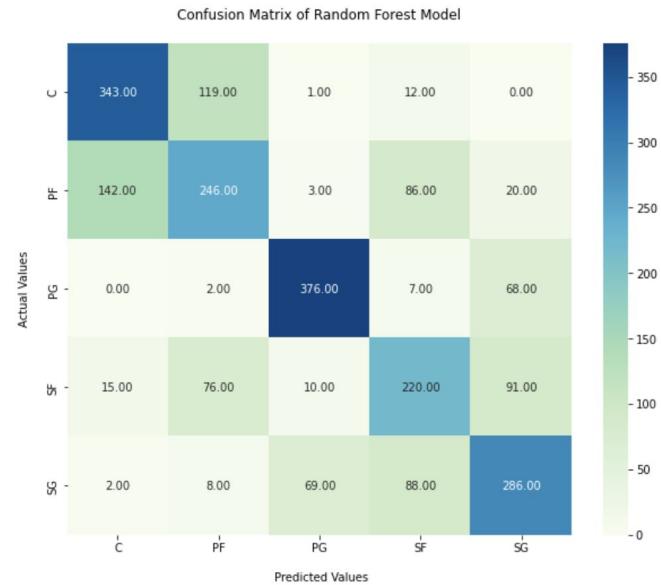
Pengerjaan Problem Set 2A - Modeling (cont.)

- Evaluasi Model Terbaik: Random Forest**

Dari keempat model tadi, dapat disimpulkan bahwa Random Forest memiliki rerata skor F1 mikro tertinggi dibanding ketiga model lainnya. Dari situ, kita dapat menerapkan model Random Forest tersebut pada data testing yang sudah di-split di awal.

Hasil Evaluasi berdasarkan classification report				
	precision	recall	f1-score	support
C	0.68	0.72	0.70	475
PF	0.55	0.49	0.52	497
PG	0.82	0.83	0.82	453
SF	0.53	0.53	0.53	412
SG	0.62	0.63	0.62	453
accuracy			0.64	2290
macro avg	0.64	0.64	0.64	2290
weighted avg	0.64	0.64	0.64	2290

F1 Macro Average: 0.6404250395167338
 F1 Micro Average: 0.6423580786026201
 Precision Macro Average: 0.6391270011575603
 Precision Micro Average: 0.6423580786026201
 Recall Macro Average: 0.6424848636034562
 Recall Micro Average: 0.6423580786026201



Pengerjaan Problem Set 2A - Conclusion

- **Conclusion**

Dari keempat model klasifikasi yang dibuat, hasil evaluasi semua model berkisar antara 58% hingga 65% dengan model terbaiknya, Random Forest, memiliki tingkat akurasi sebesar ~65%.

Hal tersebut berarti model Random Forest memiliki probabilitas sebesar 65% dalam memprediksi posisi pemain NBA dengan tepat berdasarkan data yang tersedia. Angka tersebut memang masih belum terlalu memuaskan. Hal itu bisa saja disebabkan oleh data yang memang masih kurang baik.

Menurut kami, Random Forest mampu memberikan hasil yang paling baik karena Random Forest adalah model ensemble dan juga didesain untuk meningkatkan keakuratan serta memecahkan masalah overfitting yang ada pada Decision Tree. Selain itu, Random Forest juga mampu menangani dataset berdimensi tinggi dengan baik, walaupun memang waktu komputasi yang diperlukan sedikit lebih lama jika dibandingkan dengan model lainnya.

Parameter Random Forest

```
max_depth = 12  
min_samples_leaf = 5  
min_samples_split = 10
```

*nilai parameter yang dihasilkan dari hyperparameter tuning mungkin saja sedikit berbeda setiap kali program dijalankan.

Problem Set 2B

Buatlah model untuk memprediksi gaji dari seorang pemain!

Pengerjaan Problem Set 2B

B. Buatlah model untuk memprediksi gaji dari seorang pemain!

Permasalahan di atas dapat diselesaikan menggunakan pemodelan **regresi**. Berikut langkah-langkah yang perlu dilakukan:

- Preprocessing
- Preparation
- Modeling
 - Decision Tree
 - Extra Tree
 - Hist Gradient Boosting
 - Random Forest
 - Lasso
- Conclusion

Pengerjaan Problem Set 2B - Preprocessing

- Modifikasi nama kolom “Players” menjadi “Name” pada data frame players agar sesuai dengan yang ada di dataframe salaries
- Join 2 data frame yaitu players dan salaries berdasarkan kolom “Name” dan “Year”
- Cek null value, duplikasi, dan outliers dari hasil join kedua dataframe
- Drop kolom-kolom yang tidak relevan dengan fitur target
- 1-hot-encoding pada fitur kategorikal

Pengerjaan Problem Set 2B - Preparation

- Memisahkan fitur target dari dataframe
- Melakukan feature selection untuk mendapatkan kolom-kolom paling signifikan terhadap fitur target
- Splitting data menjadi training dan testing.

Pengerjaan Problem Set 2B - Modeling

Menerapkan beberapa regression model dengan metode:

- Decision Tree
- Ensemble Learning seperti bagging (Random Forest dan Extra Tree) dan boosting (Hist Gradient)
- Linear seperti Lasso dengan regulasi L₁

yang menggunakan MAE, MSE, RMSE, dan R² sebagai evaluation metrics

Pengerjaan Problem Set 2B - Modeling (cont.)

- Decision Tree

MAE: 845799.7818791947

MSE: 2510273160654.931

RMSE: 1584384.1581683815

R_squared: 0.7688266642582854

- Random Forest

MAE: 628277.6330201342

MSE: 1254541771887.7852

RMSE: 1120063.289233151

R_squared: 0.8844681085786875

Pengerjaan Problem Set 2B - Modeling (cont.)

- Extra Tree
- Hist Gradient Boosting

MAE: 627254.8217953021

MSE: 1241833534776.2283

RMSE: 1114375.849871231

R_squared: 0.8856384216786805

MAE: 650067.9790658664

MSE: 1267681119773.8352

RMSE: 1125913.4601619411

R_squared: 0.8832580940958427

Pengerjaan Problem Set 2B - Modeling (cont.)

- Lasso

```
R-squared Predict 1: 0.7654593851144402
R-squared Predict 2: 0.765459984767993
R-squared Predict 3: 0.7654608808883581
R-squared Predict 4: 0.7654617736673275
R-squared Predict 5: 0.7654626619437672
RMSE 1: 1595881.5400981698
RMSE 2: 1595879.4999895785
RMSE 3: 1595876.4512529164
RMSE 4: 1595873.413878395
RMSE 5: 1595870.391816451
```

Pengerjaan Problem Set 2B - Conclusion

Dari hasil yang didapatkan, dapat disimpulkan regression model dengan ensemble learning menghasilkan metrics yang paling baik. Contohnya adalah Extra Tree yang menghasilkan metrics berikut:

```
MAE: 627254.8217953021
MSE: 1241833534776.2283
RMSE: 1114375.849871231
R_squared: 0.8856384216786805
```

Dari nilai R_squared yang didapat, model bisa dibilang cukup baik dalam menjelaskan variansi yang ada di dalam dataset. Walaupun metrics lainnya seperti RMSE masih belum optimal, yang salah satunya mungkin disebabkan oleh skala fitur target yang besar pula (salaries).



Problem Set 2C

Buatlah cluster dari dataset dan jelaskan karakteristik dari cluster-cluster yang terbentuk!

Pengerjaan Problem Set 2C

C. Buatlah cluster dari dataset dan jelaskan karakteristik dari cluster-cluster yang terbentuk!

Permasalahan di atas dapat diselesaikan menggunakan pemodelan **clustering pada dataset Players**.

Berikut langkah-langkah yang perlu dilakukan:

- Preparation
 - Standardscaler
 - PCA
- K-Means Clustering
 - Visualisasi hasil clustering menggunakan Scatter Plot
 - Membandingkan persebaran data tiap cluster menggunakan KDE Plot
- Conclusion

Pengerjaan Problem Set 2C - Preparation

- **Preparation**
 - Men-drop kolom categorical Player, Tm, PF, Pos.
 - Melakukan scaling dengan Standardscaler.
 - Melakukan dimensionality reduction dengan PCA menjadi 3 Principle Component.

Pengerjaan Problem Set 2C - Silhouette Coefficient

- K-Means
 - Menentukan nilai K dengan mencari Silhouette Coefficient yang terbaik

Silhouette Coefficient dari nilai K = 2 adalah 0.410

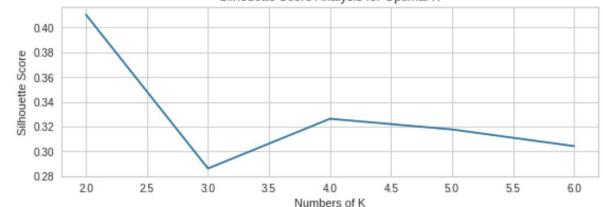
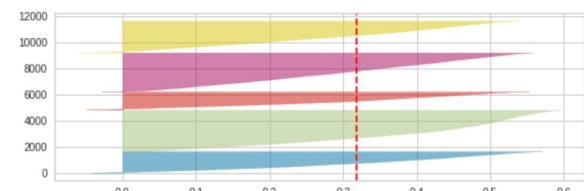
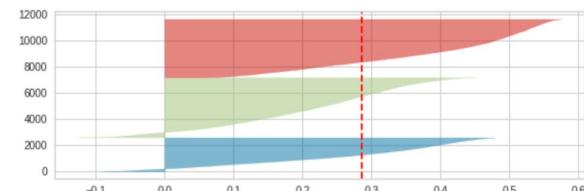
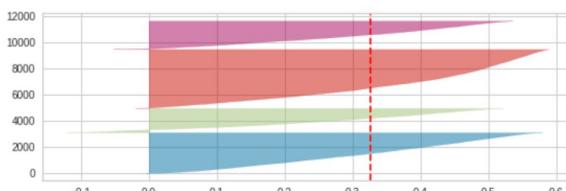
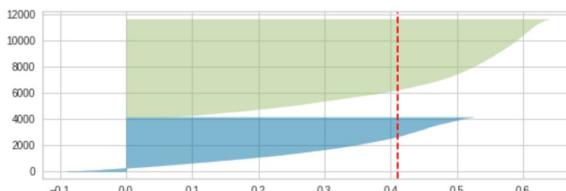
Silhouette Coefficient dari nilai K = 3 adalah 0.286

Silhouette Coefficient dari nilai K = 4 adalah 0.326

Silhouette Coefficient dari nilai K = 5 adalah 0.318

Silhouette Coefficient dari nilai K = 6 adalah 0.304

Memilih K = 2 karena nilai Silhouette Coefficient nya mendekati 1



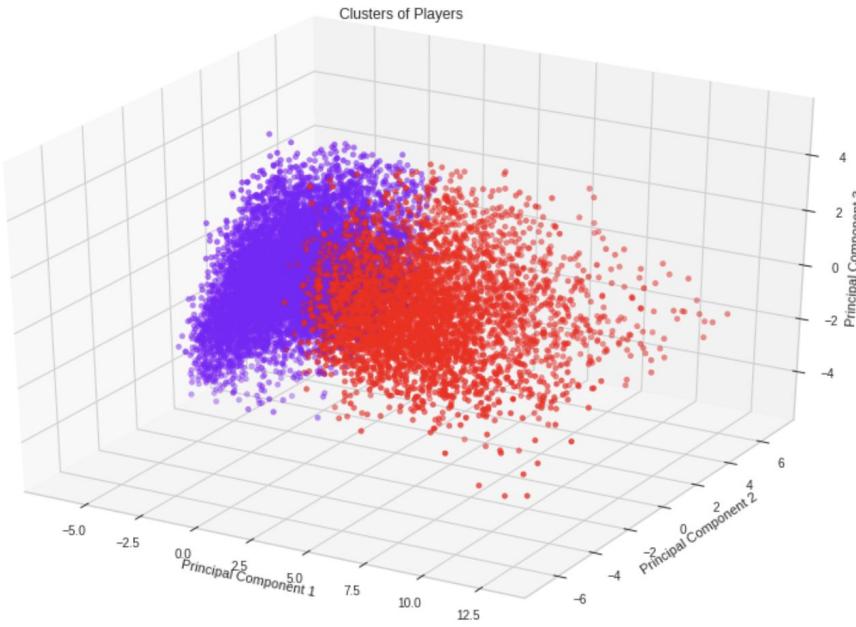
Pengerjaan Problem Set 2C - K-Means

Melakukan K-Means dengan 2 cluster:

	0	1	2	Clusters
0	-3.439290	1.539629	-0.024495	0
1	1.624011	-0.771576	-2.234013	1
2	-1.227484	2.238950	0.983478	0
3	9.114658	-5.481761	-2.484928	1
4	2.691045	0.623136	-1.046619	1
...
11602	6.486036	0.402601	2.265058	1
11603	1.944520	-1.365926	1.700622	1
11604	1.993184	-1.231549	1.329819	1
11605	8.146556	-4.368775	-0.527051	1
11606	1.508997	2.596342	2.560211	1

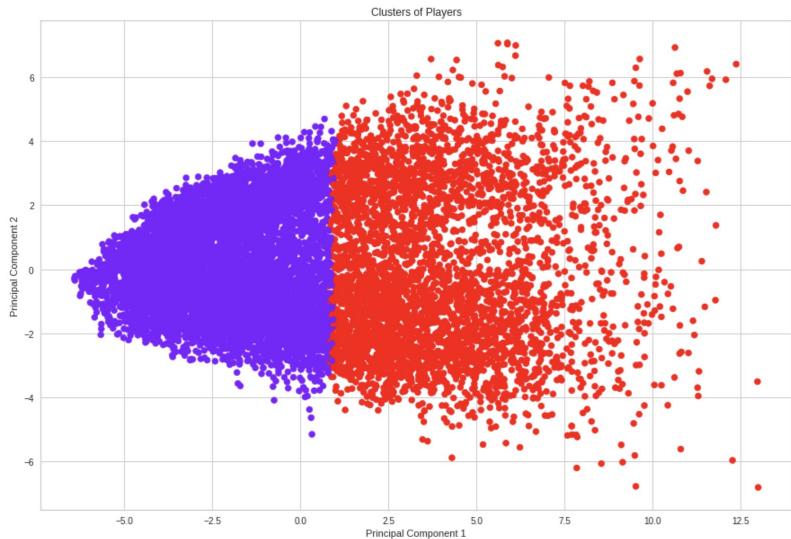
11607 rows × 4 columns

Scatterplot 3D:

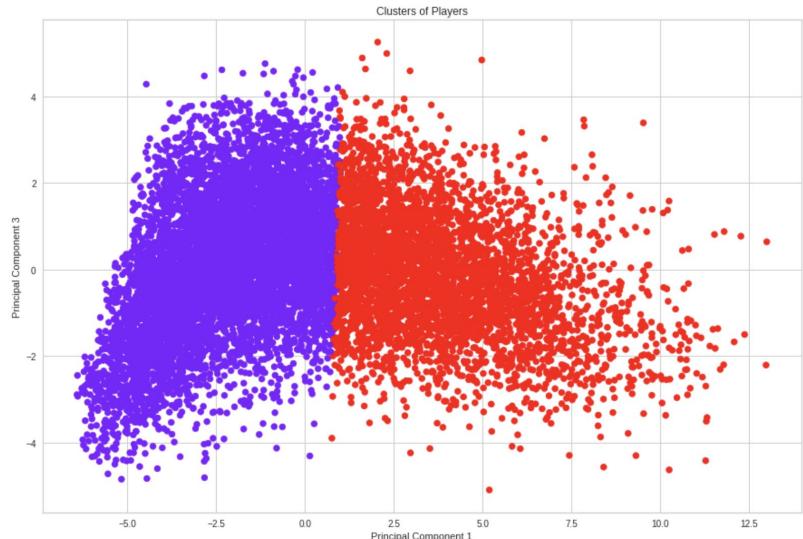


Pengerjaan Problem Set 2C - K-Means (cont.)

Scatterplot 2D:

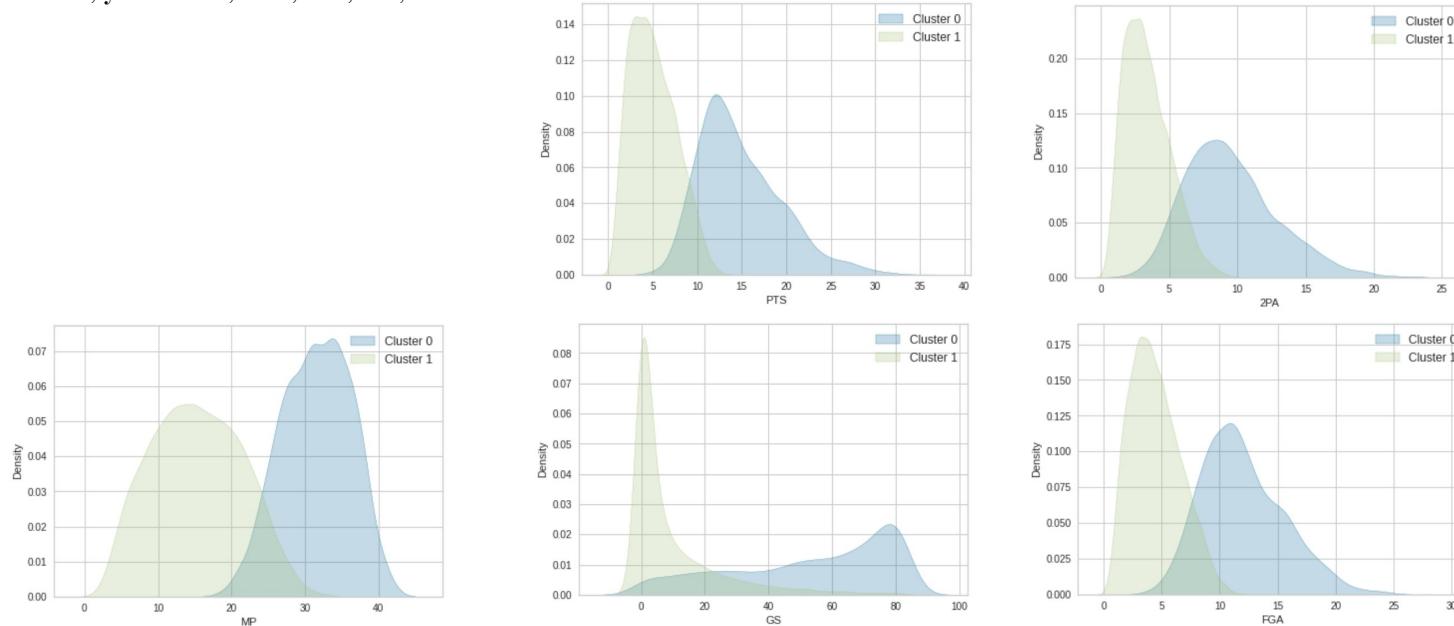


Scatterplot 2D sudut pandang lain:



Pengerjaan Problem Set 2C - Analisis karakteristik

Analisis karakteristik kedua cluster dengan visualisasi KDE Plot. Visualisasi KDE Plot dibawah dilakukan untuk mengomparasi kedua cluster dari semua kolom yang ada setelah melewati proses clustering. Terdapat 5 kolom yang cukup terlihat jelas berbeda antara kedua cluster, yaitu PTS, 2PA, MP, GS, FGA



Pengerjaan Problem Set 2C - Conclusion

- **Conclusion**

Pada hasil clustering, terbentuk 2 cluster berdasarkan hasil analisis Silhouette Score. Kedua cluster tersebut tidak terlalu memiliki karakteristik yang unik. Namun, berdasarkan visualisasi KDE plot untuk membandingkan persebaran data setiap kolom pada kedua cluster, kami berhasil menemukan beberapa kolom yang dapat dijadikan karakteristik kedua Cluster. Berikut karakteristik kedua cluster yang kami temukan:

- Cluster o didominasi oleh pemain yang lebih banyak **mencetak poin** dibanding cluster 1.
- Cluster o didominasi oleh pemain yang lebih banyak memberikan **2-point score** dibanding cluster 1
- Cluster o didominasi oleh pemain yang lebih banyak melakukan **field goal attempt** dibanding cluster 1
- Cluster o didominasi oleh pemain yang **waktu bermainnya** lebih banyak dibanding cluster 1.

Dari temuan kami diatas, kesimpulan yang kami ambil adalah cluster o didominasi oleh pemain papan atas sedangkan cluster 1 didominasi oleh pemain papan menengah ke bawah.

A photograph of a basketball court in a tropical setting. In the foreground, a weathered basketball hoop stands on a dark, textured court surface. To its right, a soccer goal is partially visible. The court is situated in a large, open grassy field. In the background, several tall palm trees stand against a backdrop of dense, green hills and mountains under a cloudy sky.

Terima Kasih