

# Cycling Segments

Proyek Akhir



# Kelompok kafka



2206029191

**Muhammad Daffa'I  
Rafi Prasetyo**



2206081976

**Ramadhan Andika  
Putra**



2206081963

**Shirin Zarqaa  
Rabbaanii Arham**



2206825776

**Vina Myrnauli  
Abigail Siallagan**

# Data Preprocessing

Step 1

```
[500] df.info()  
  
-> <class 'pandas.core.frame.DataFrame'>  
RangeIndex: 6292 entries, 0 to 6291  
Data columns (total 17 columns):  
 #   Column           Non-Null Count  Dtype     
---  --  
 0   user_age_group  6292 non-null    object    
 1   user_id         6292 non-null    int64     
 2   attempt_date    6292 non-null    object    
 3   gender          6292 non-null    object    
 4   smt_rank         6292 non-null    int64     
 5   smt_avg_spd     6292 non-null    float64  
 6   smt_finish_seconds 6292 non-null    int64     
 7   smt_name         6292 non-null    object    
 8   user_weight_category 5932 non-null    object    
 9   act_title        6292 non-null    object    
 10  act_avg_spd     6292 non-null    float64  
 11  act_max_spd     6292 non-null    float64  
 12  act_total_km     6292 non-null    float64  
 13  act_moving_seconds 6292 non-null    int64     
 14  act_total_seconds 6292 non-null    int64     
 15  has_hr_data      6292 non-null    int64     
 16  id               6292 non-null    int64     
dtypes: float64(4), int64(7), object(6)  
memory usage: 835.8+ KB
```

Membuat dan membaca infomasi  
dataframe

Step 2

```
df_clean['user_weight_category'] = fill_null_mode(df_clean['user_weight_category'])  
  
df_clean.isnull().sum()
```

	0
user_age_group	0
user_id	0
attempt_date	0
gender	0
smt_rank	0
smt_avg_spd	0
smt_finish_seconds	0
smt_name	0
user_weight_category	0
act_title	0
act_avg_spd	0
act_max_spd	0
act_total_km	0
act_moving_seconds	0
act_total_seconds	0
has_hr_data	0
id	0

informasi  
missing value  
sebelum di  
handle

Handle Missing Values

Step 3

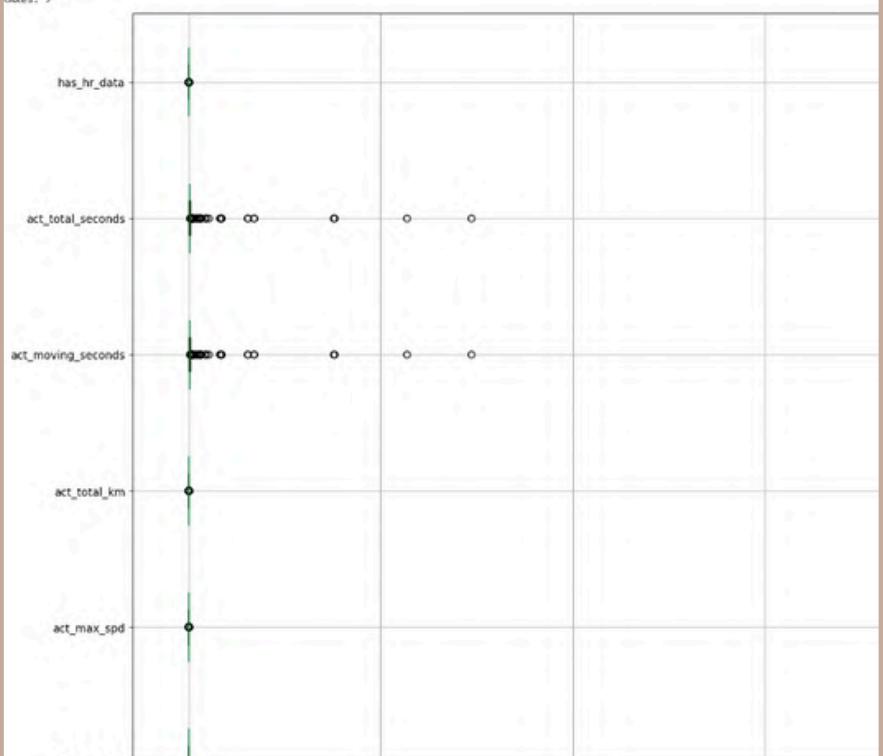
## Handle duplicate data

```
] duplicate_rows = df_clean[df_clean.duplicate_rows]  
duplicate_total_rows, duplicate_total_percent = len(duplicate_rows), (len(duplicate_rows) / len(df_clean)) * 100  
print('jumlah data duplicate: ', duplicate_total_rows)  
  
jumlah data duplicate: 0  
  
] df_clean.drop(['id', 'user_id'])
```

Handle Duplicate Data

Step 4

```
hitung_outlier(df_clean)  
  
smt_rank smt_avg_spd smt_finish_seconds act_avg_spd act_max_spd act_total_km act_moving_seconds act_total_seconds  
Outliers count 163.000000 53.000000 288.000000 94.000000 375.000000 308.000000 296.000000 296.000000  
Outliers percentage (%) 2.590591 0.842339 4.577241 1.493961 5.959949 4.895105 4.704387 4.704387  
  
df_clean.boxplot(vert=False, figsize=(20,20))  
Axes: >
```



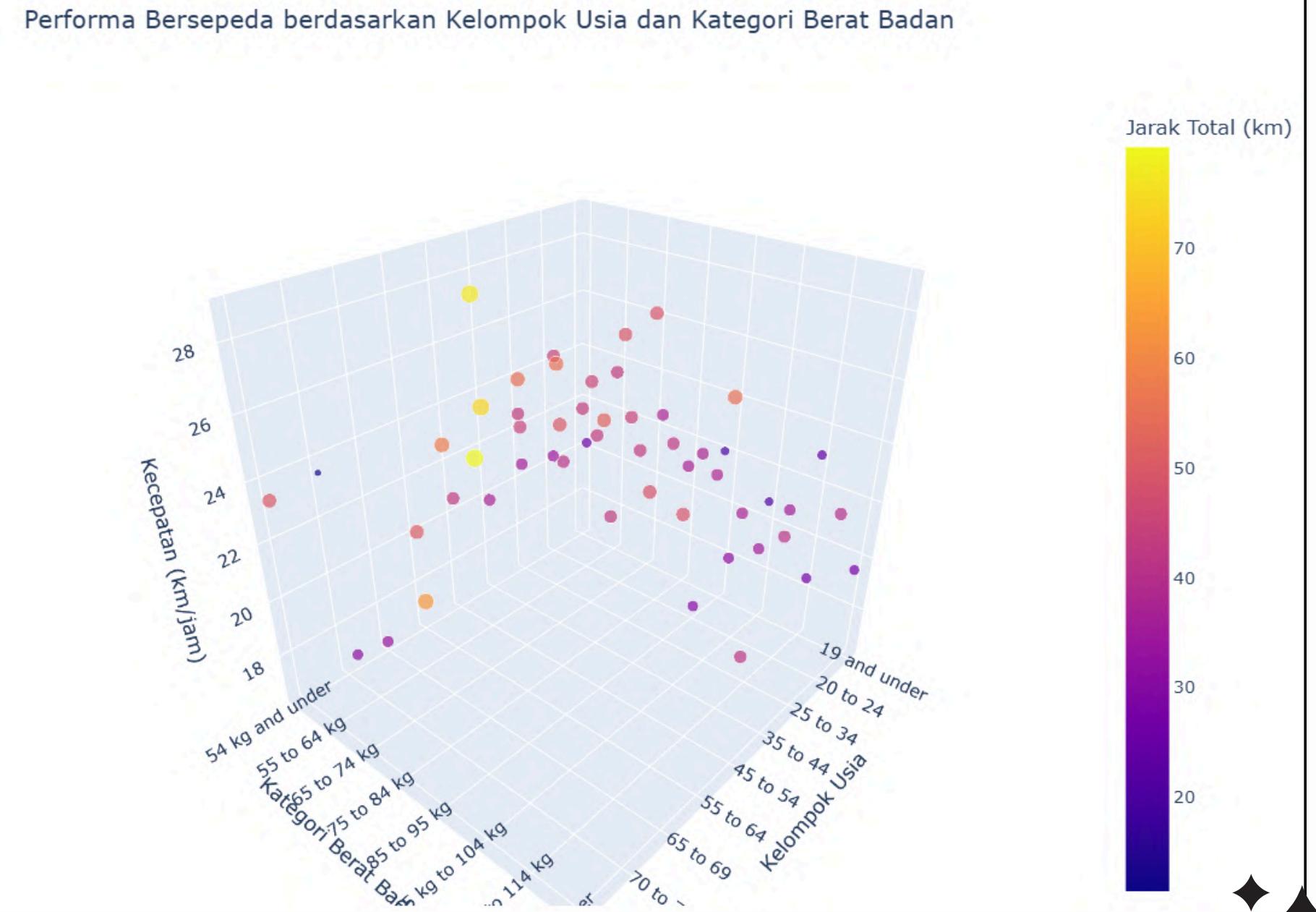
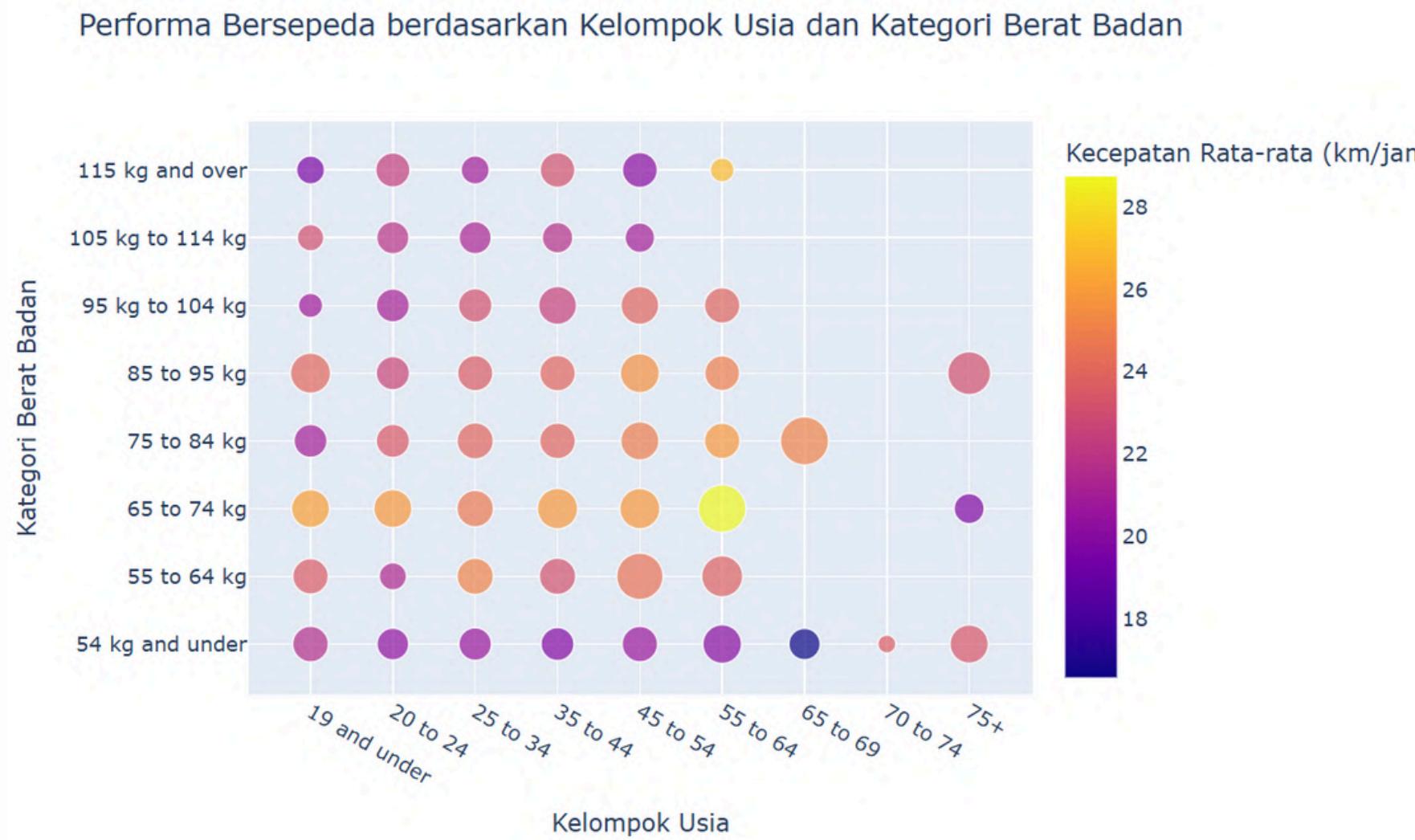
Handle Outlier

# Exploratory Data Analysis



01

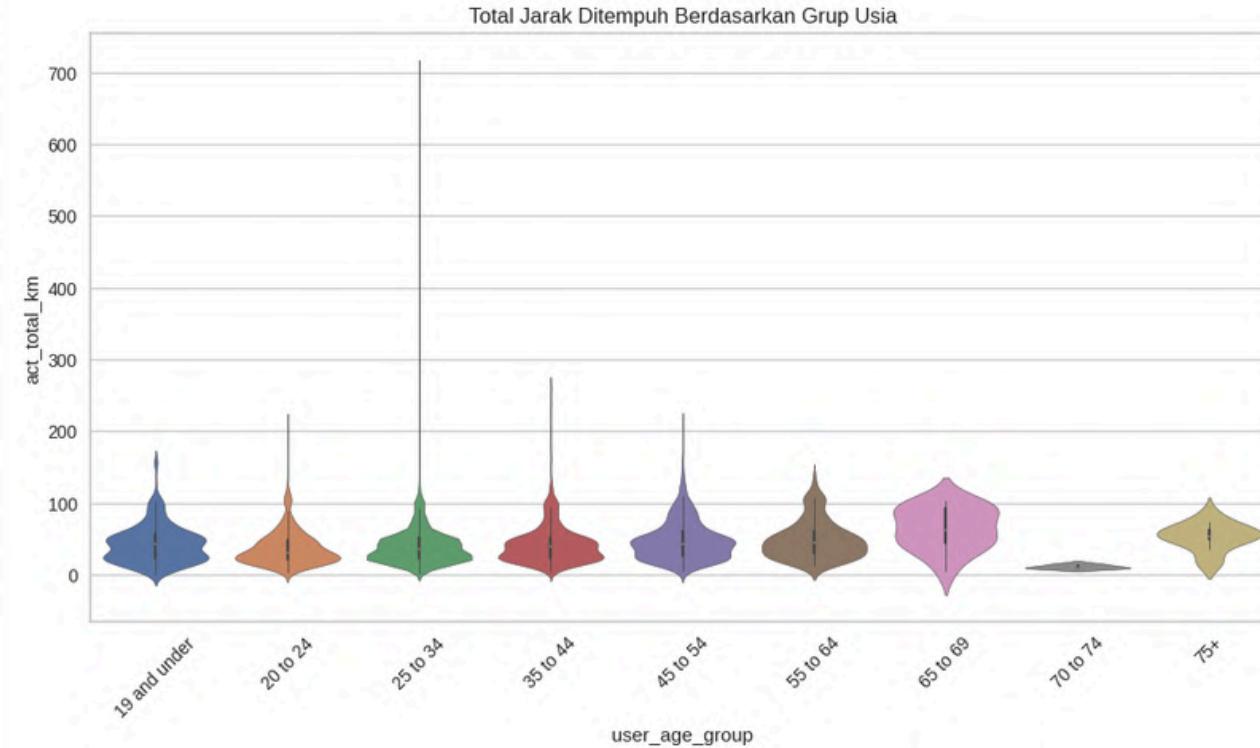
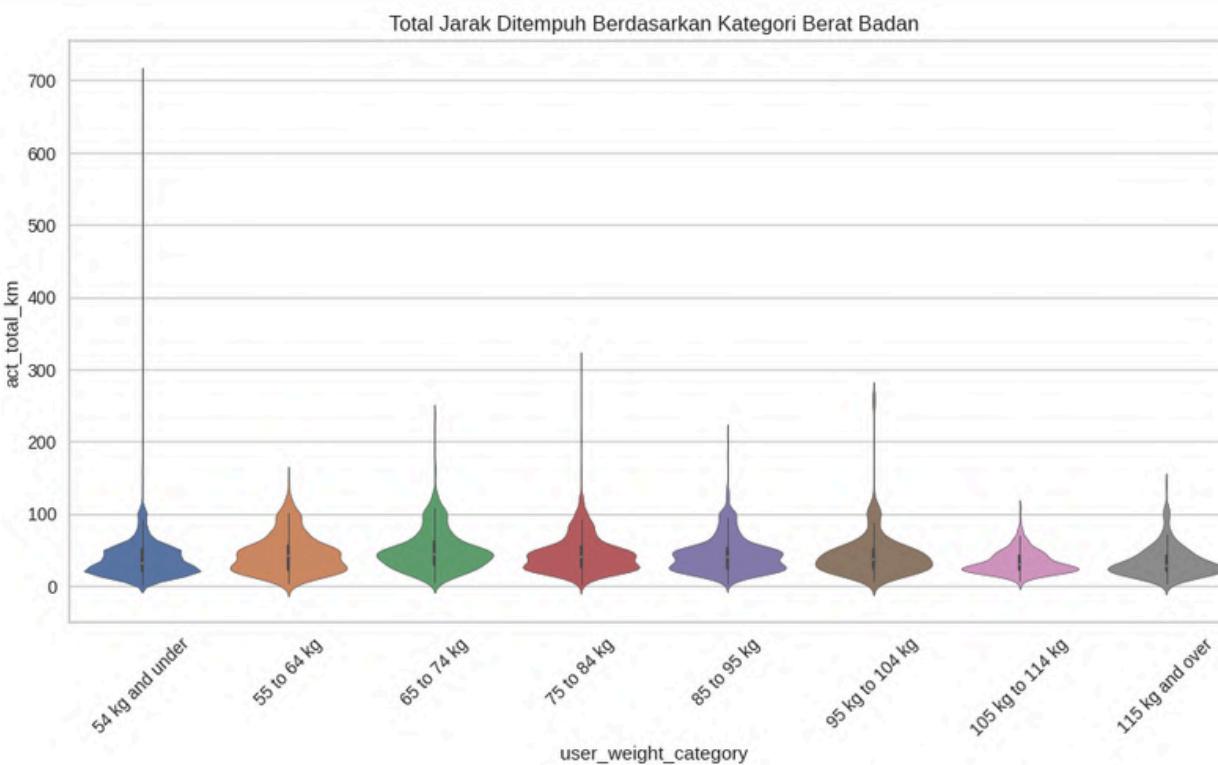
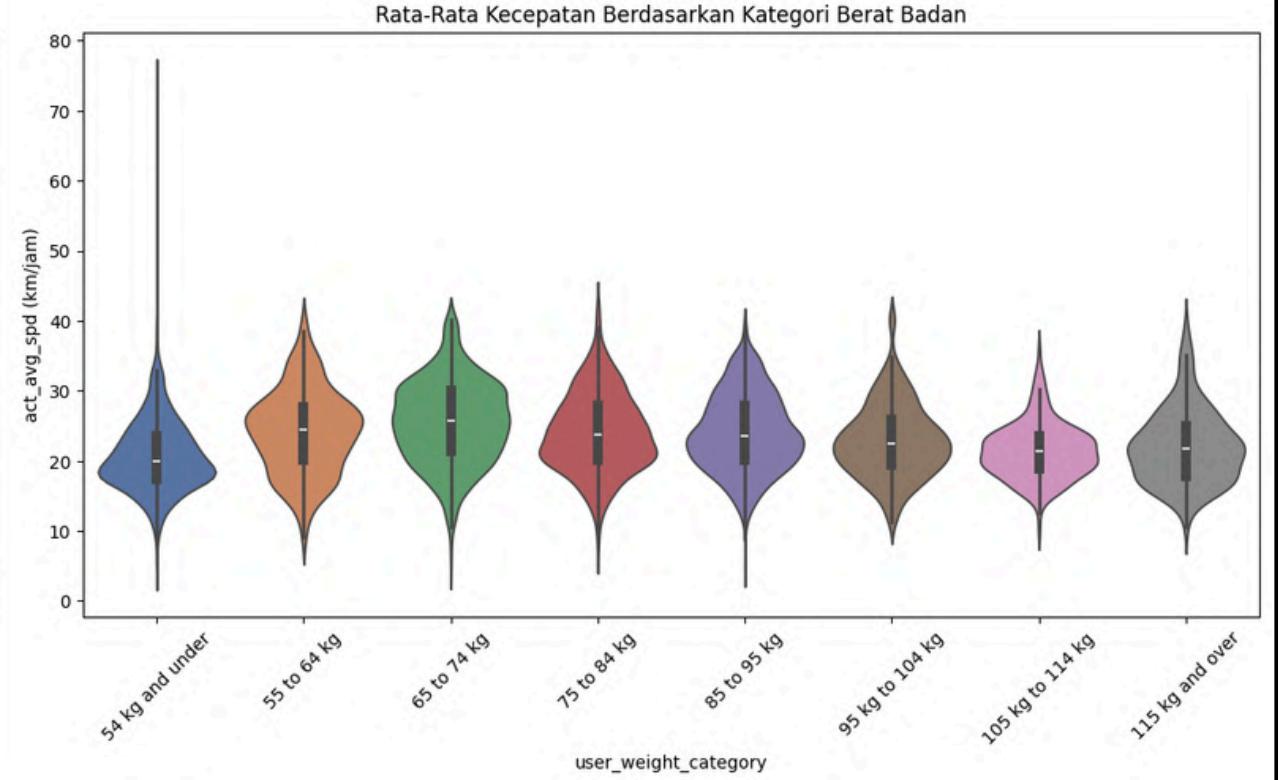
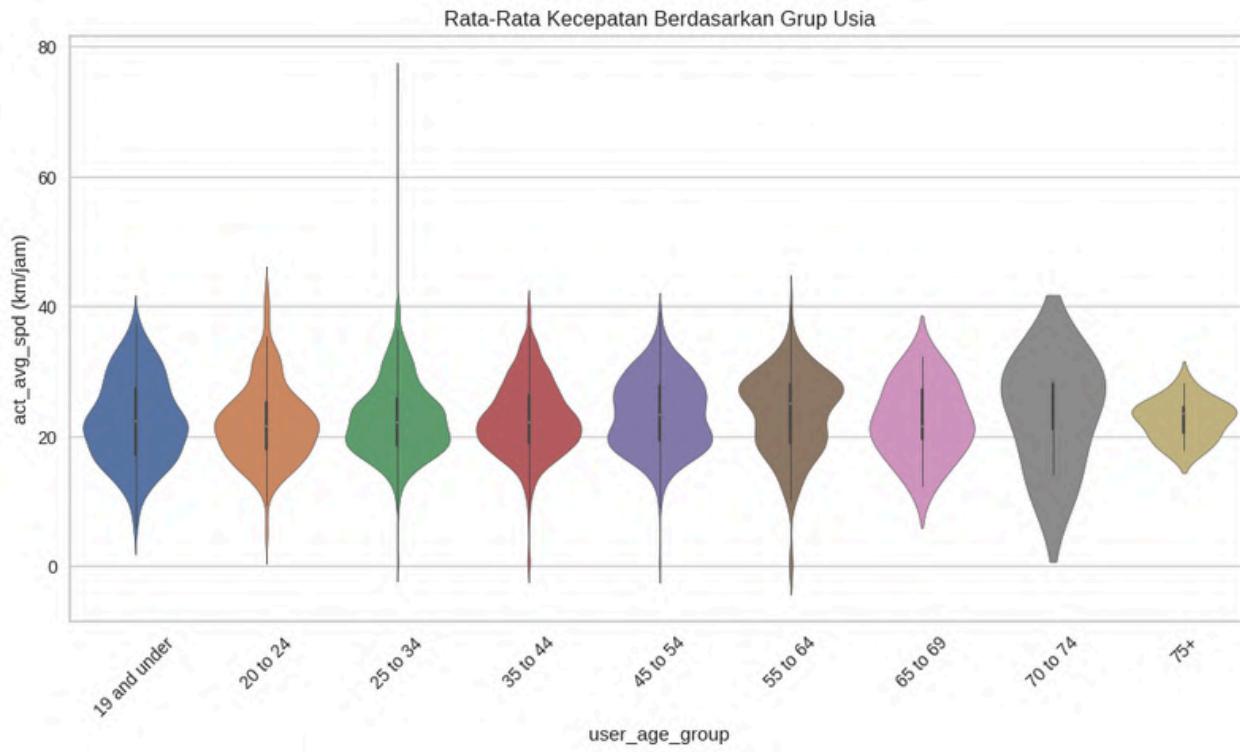
Apakah umur dan berat badan mempengaruhi performa bersepeda?



# 01

Apakah umur dan berat badan mempengaruhi performa bersepeda?

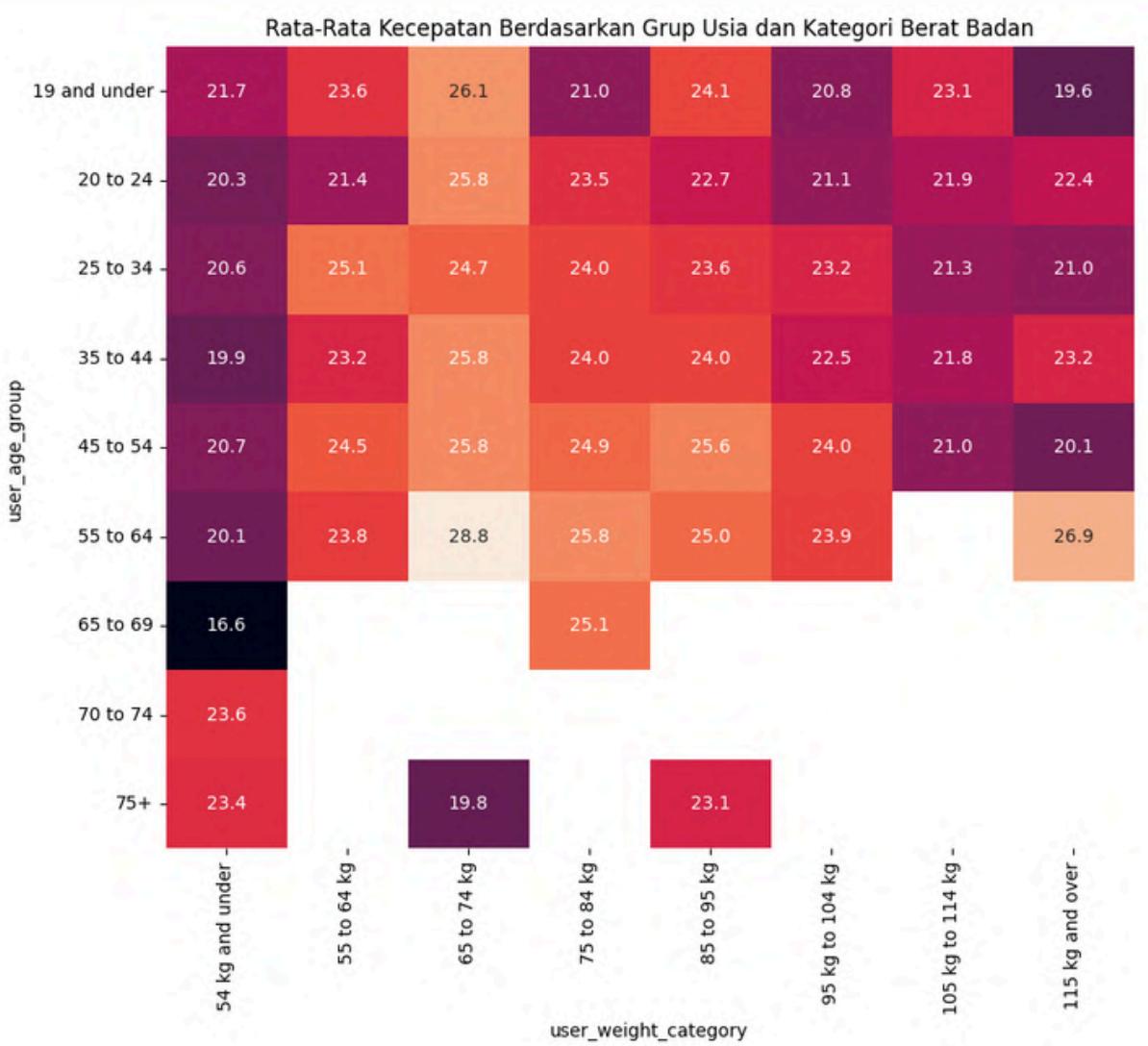
Rata-rata kecepatan ►



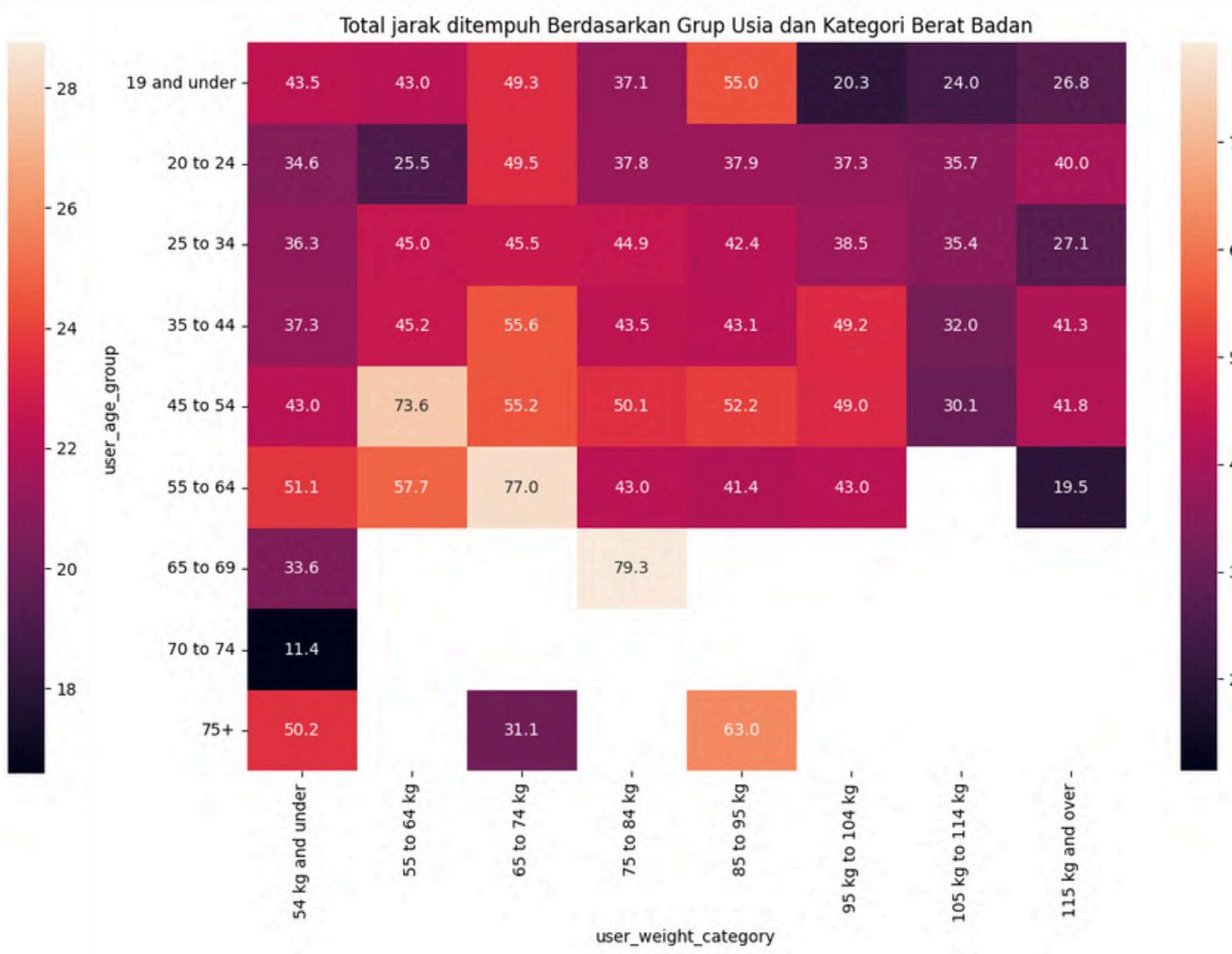
◀ Total jarak ditempuh

# 01

Apakah umur dan berat badan mempengaruhi performa bersepeda?



Rata-rata kecepatan



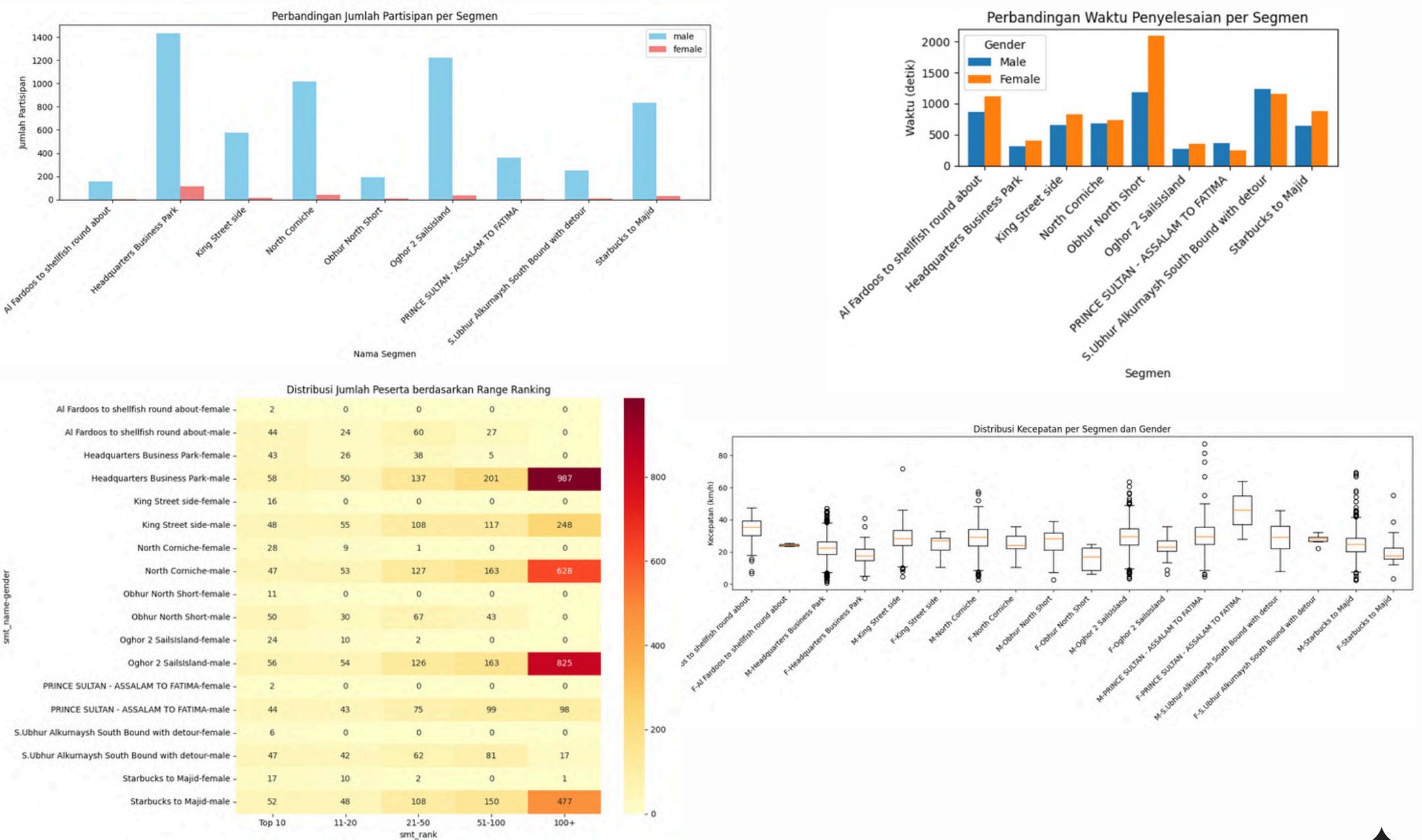
Total jarak ditempuh



Umur dan berat badan **terbukti berpengaruh** terhadap performa bersepeda. Kelompok usia menengah (25-69 tahun) dengan berat badan sedang (65-84 kg) cenderung memiliki kecepatan dan jarak tempuh terbaik, sementara usia terlalu muda atau tua serta berat badan terlalu ringan atau berat menurunkan performa. Namun, beberapa individu di luar kelompok ini juga dapat mencapai hasil lebih baik, menunjukkan bahwa mungkin ada faktor lain yang juga penting dalam mempengaruhi performa bersepeda.

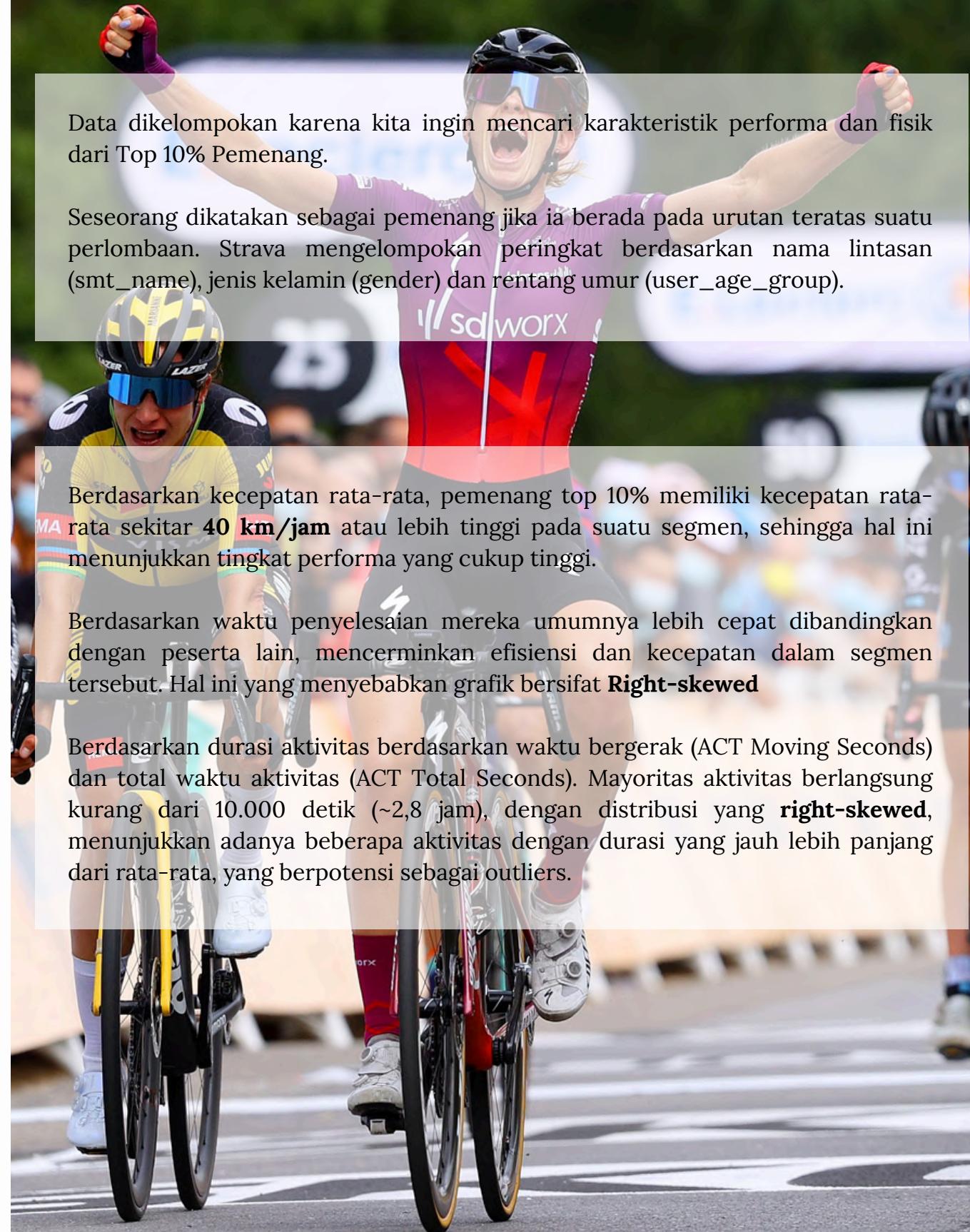
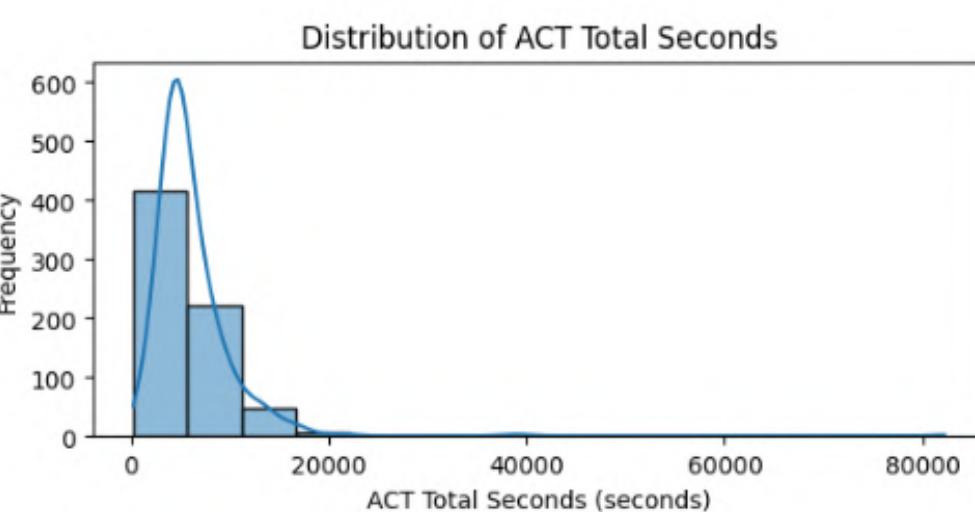
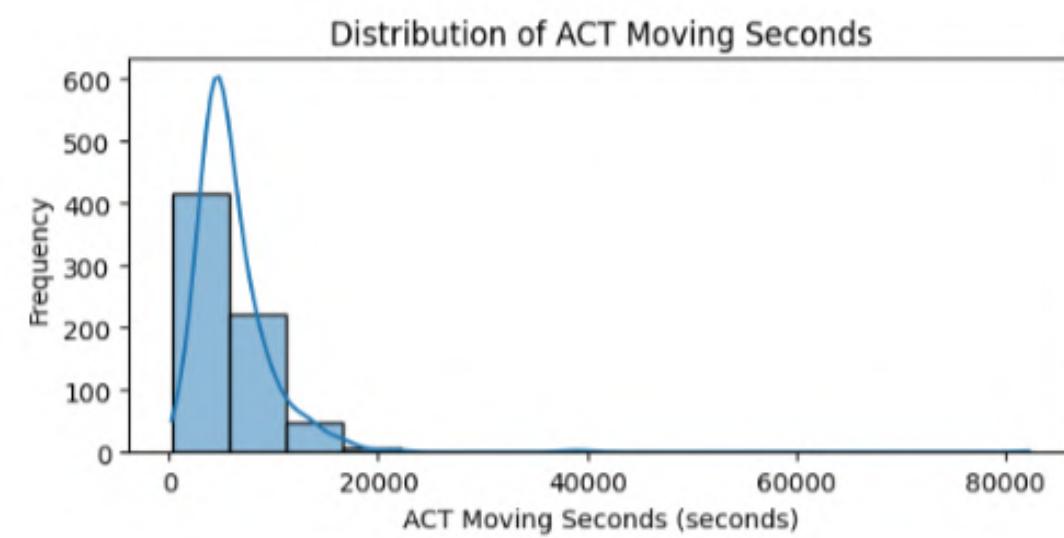
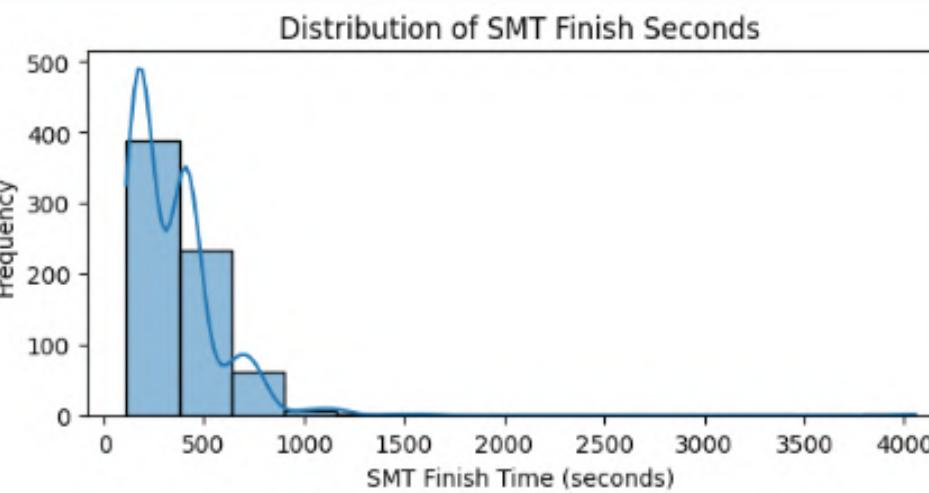
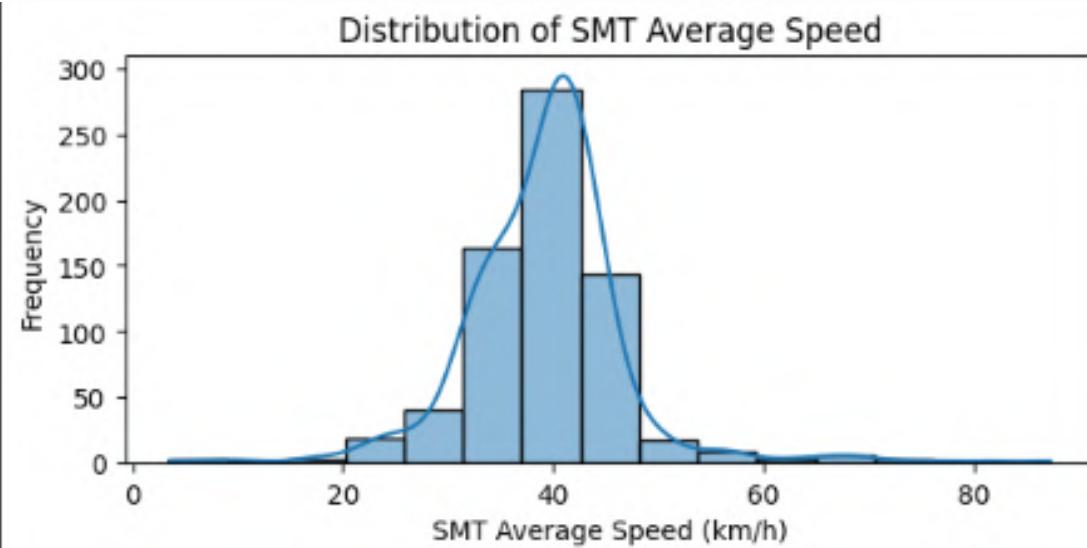
# 02

Apakah terdapat perbedaan performa bersepeda antara laki-laki dan perempuan pada segmen yang sama?



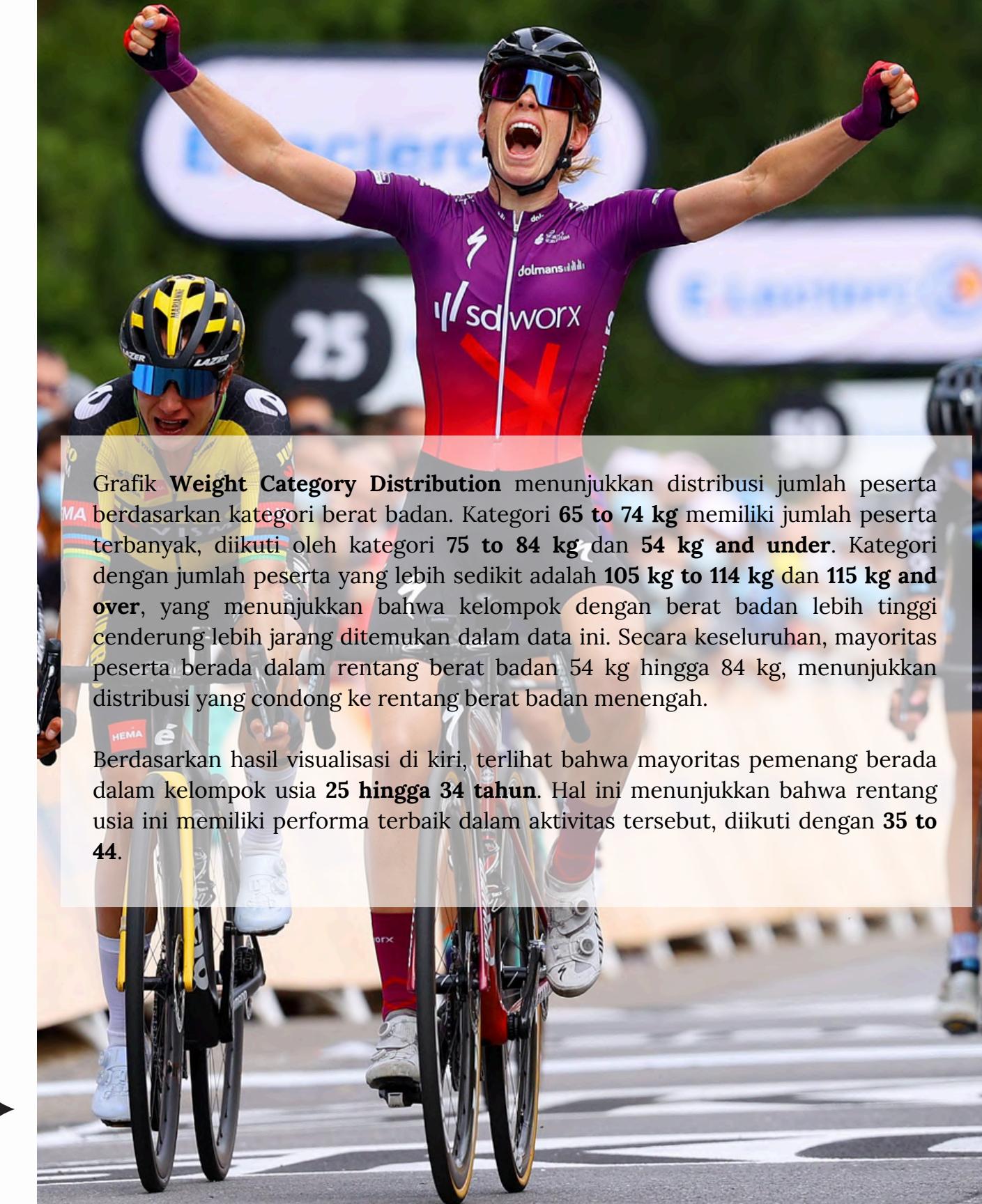
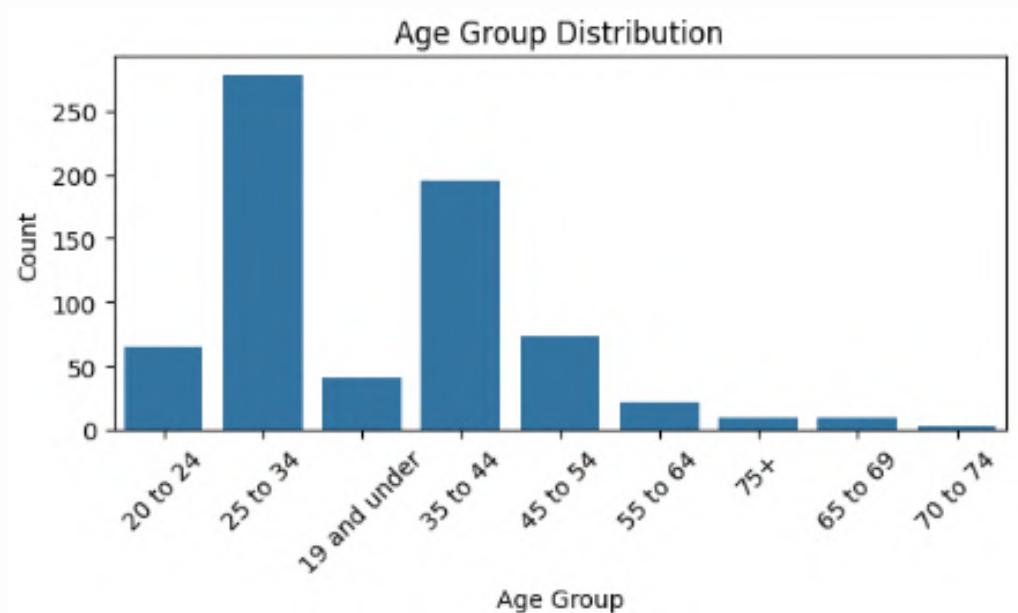
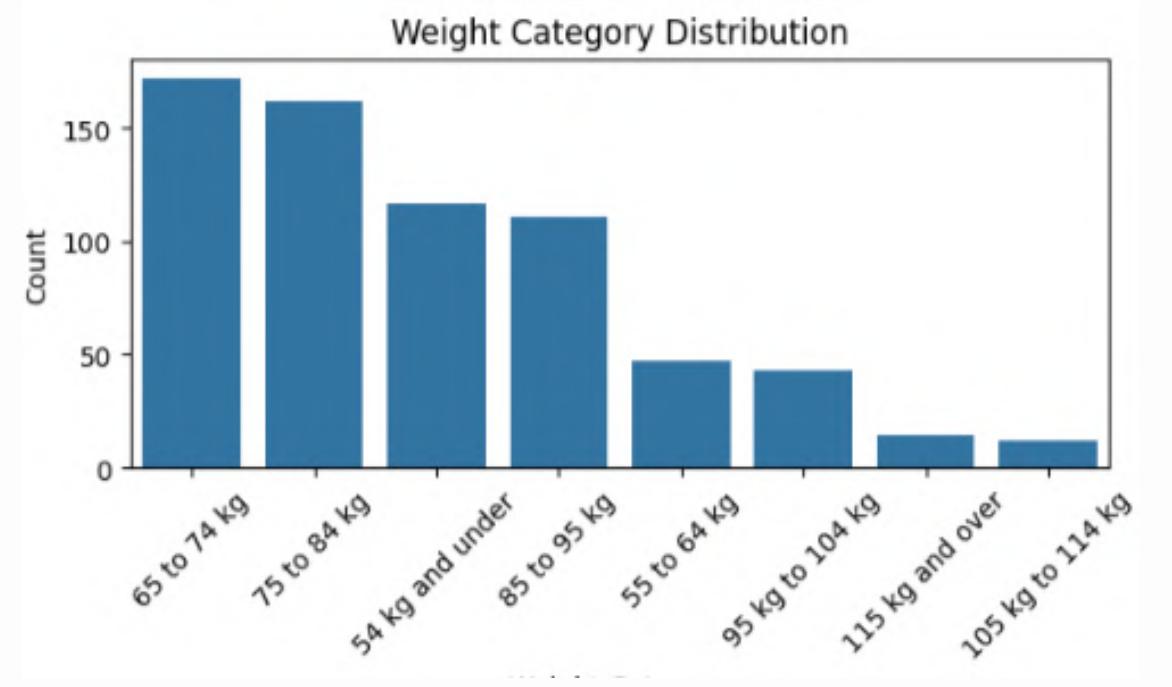
# 03

## Bagaimana karakteristik performa dan fisik top 10% pemenang?



# 03

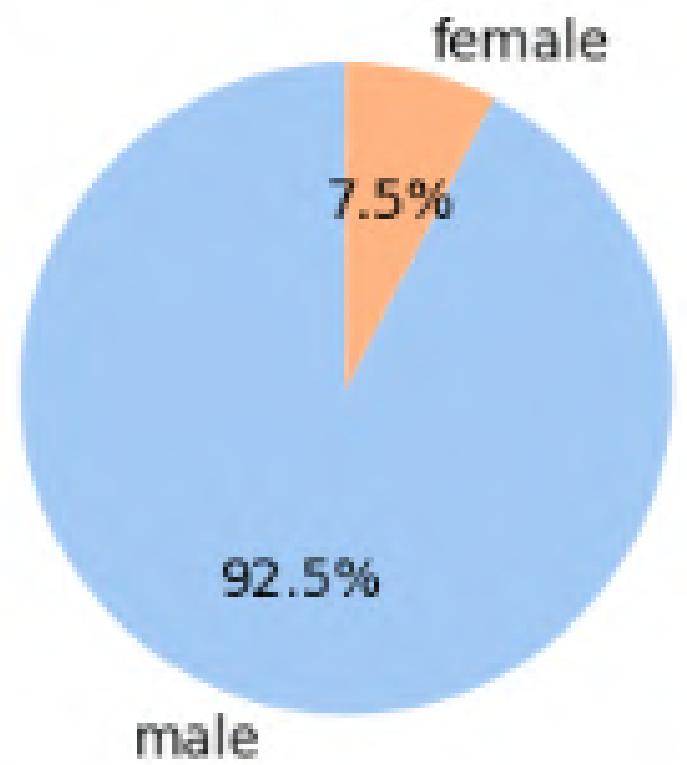
Bagaimana karakteristik performa dan fisik top 10% pemenang?



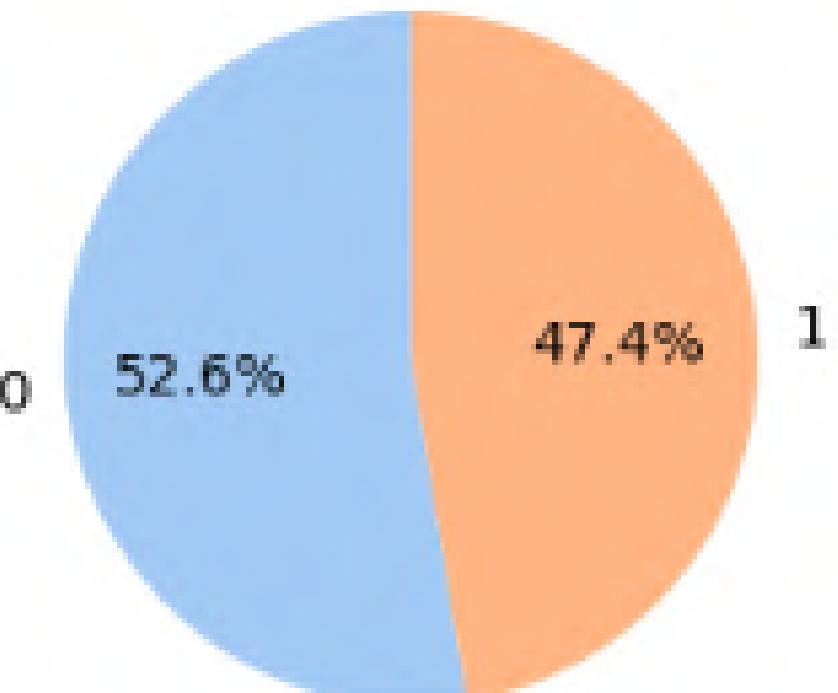
# 03

Bagaimana karakteristik performa dan fisik top 10% pemenang?

Gender Distribution

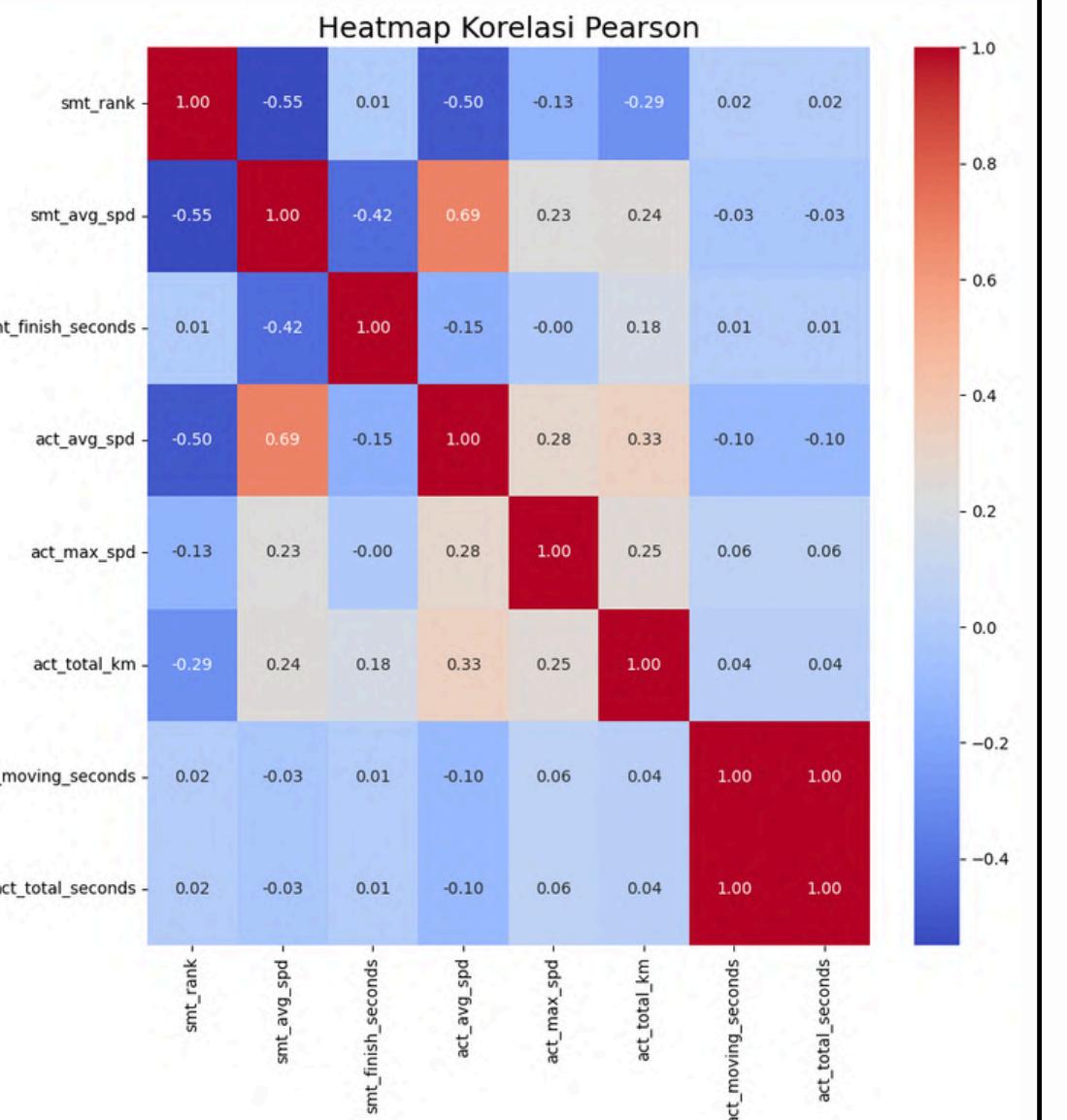
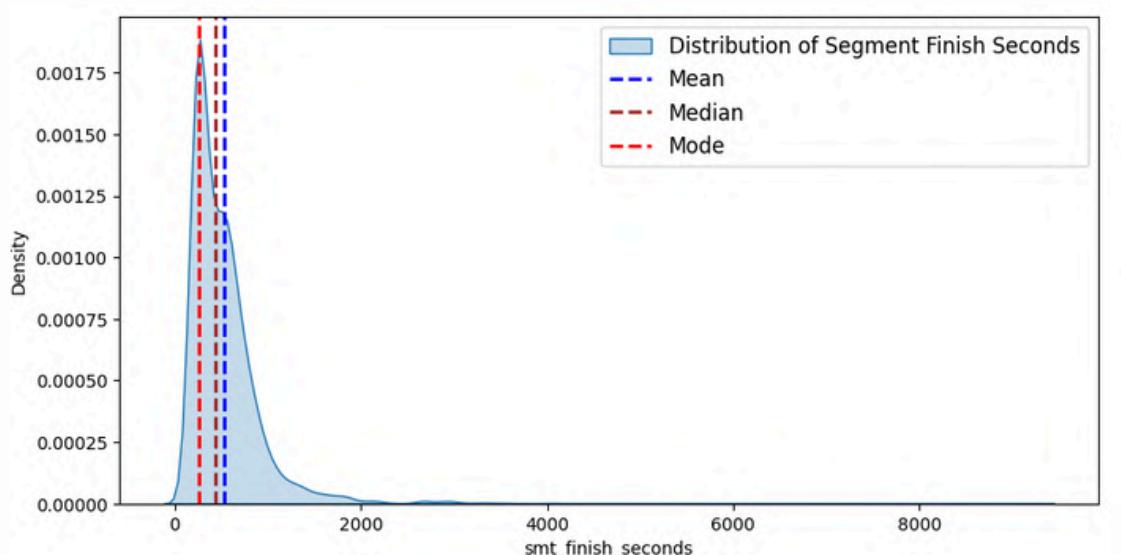
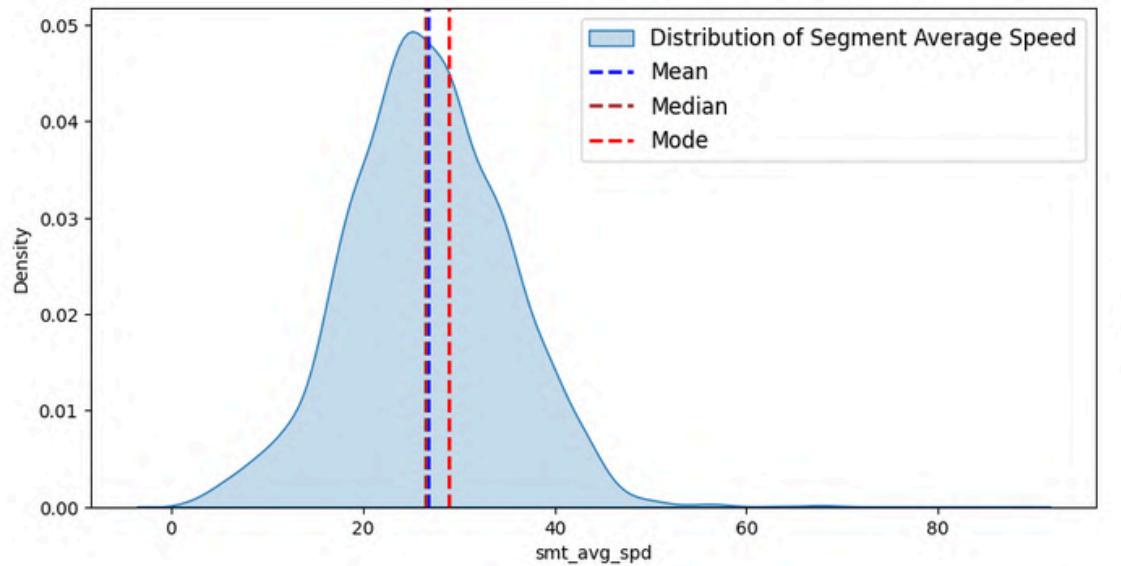


Has Heart Rate Data



# 04

Segmen apa yang memiliki leaderboard perempuan dan laki-laki dengan performa paling unggul?



**Road Code Ranking**

**STG Stage Races**



**RANKING**

Rank	Cyclist	Team	Points
1	R. Evenepoel	SOQ	2799
2	A. Vlasov	BOH	2658
3	J. Ayuso	UAD	2636
4	M. Jorgenson	TVL	2382
5	P. Roglič	BOH	2155
6	C. Rodríguez	IGD	2137
7	J. Vingegaard	TVL	2103
8	B. McNulty	UAD	1961

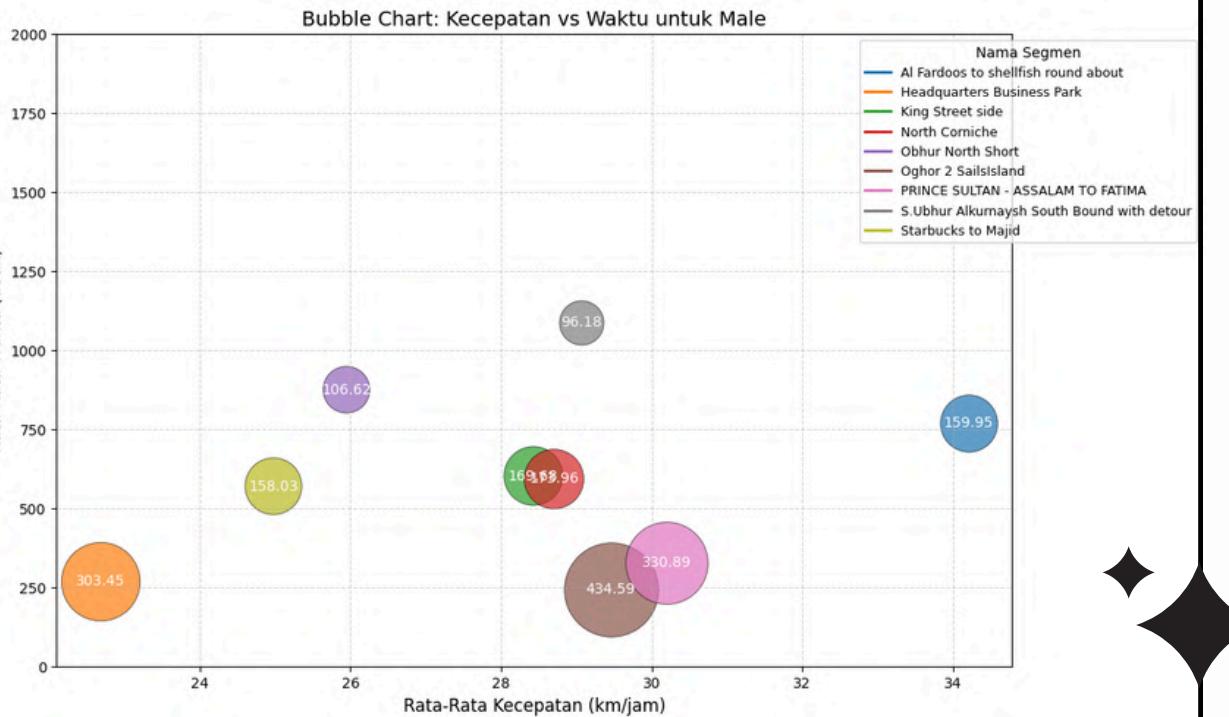
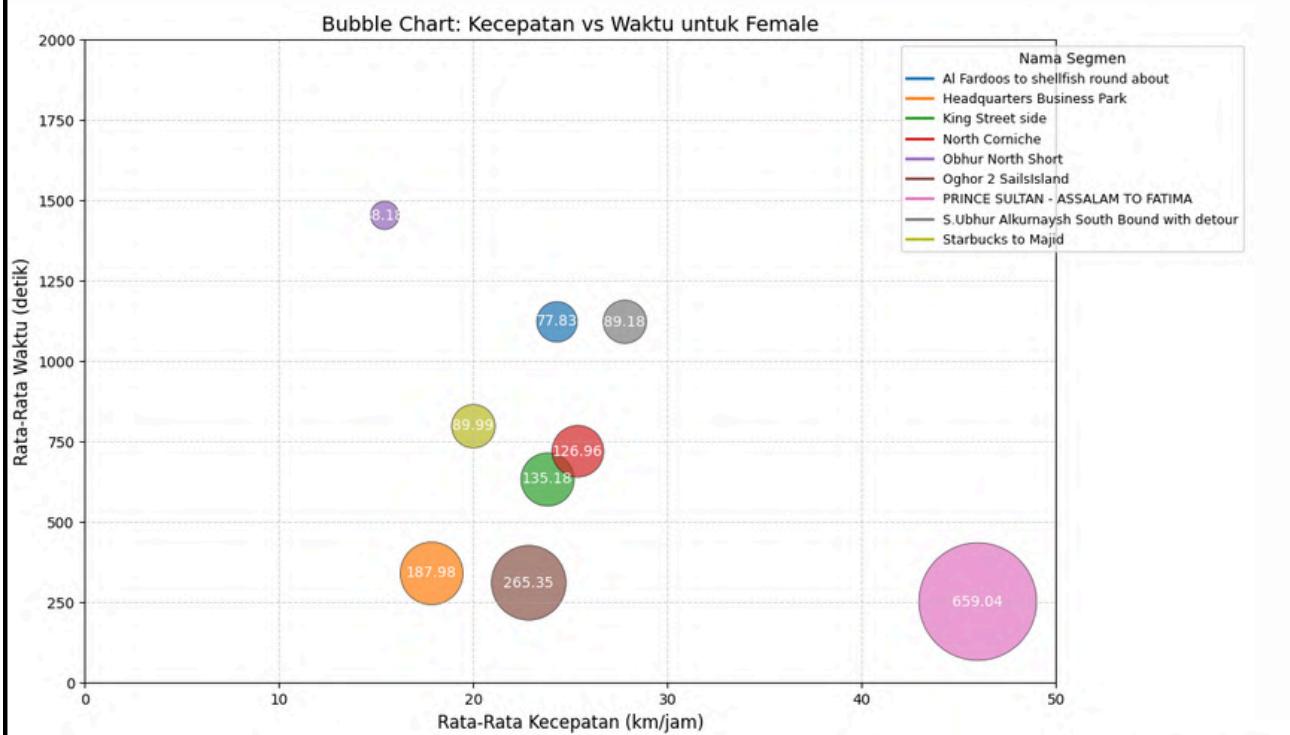
Ranking as of June 10, 2024

**Road Code**

Remco Evenepoel

# 04

Segmen apa yang memiliki leaderboard perempuan dan laki-laki dengan performa paling unggul?



**Road Code Ranking**  
Stage Races

STG

RANKING

POINTS

- 1 R. Evenepoel SOQ 2799
- 2 A. Vlasov BOH 2658
- 3 J. Ayuso UAD 2636
- 4 M. Jorgenson TVL 2382
- 5 P. Roglič BOH 2155
- 6 C. Rodríguez IGD 2137
- J. Vingegaard 2103

• Segmen "PRINCE SULTAN - ASSALAM TO FATIMA" menunjukkan kecepatan rata-rata tertinggi dan waktu penyelesaian yang terbaik untuk peserta perempuan.

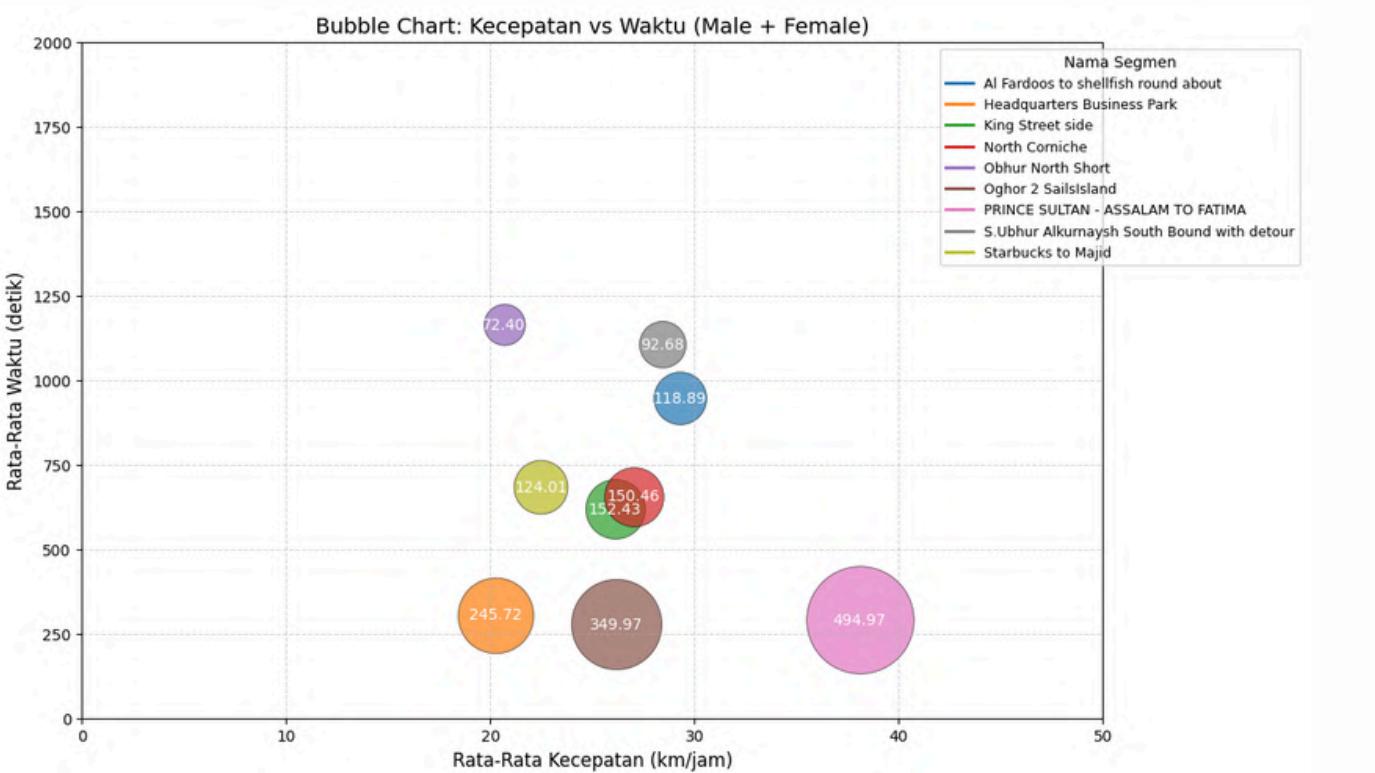
• Segmen "Oghor 2 SailsIsland" menunjukkan kombinasi waktu tempuh yang relatif kecil dengan kecepatan rata-rata yang konsisten lebih tinggi untuk peserta laki-laki..

Road Code

Evenepoel

# 04

Segmen apa yang memiliki leaderboard perempuan dan laki-laki dengan performa paling unggul?



- Untuk kedua gender, segmen "Segmen Prince Sultan - Assalam to Fatima" adalah yang paling baik.
- Insight yang didapati:
  - Kecepatan rata-rata yang tinggi mempercepat waktu penyelesaian segmen, terlihat dari korelasi negatif antara smt\_avg\_spd dan smt\_finish\_seconds. Hal ini juga membantu peserta meraih peringkat lebih baik di leaderboard.
  - Peserta laki-laki mendominasi leaderboard karena variasi performa yang lebih luas, sementara perempuan unggul di segmen tertentu seperti "PRINCE SULTAN - ASSALAM TO FATIMA."

**Road Code Ranking**  
Stage Races

STG

RANKING POINTS

Rank	Rider Name	Team	Points
1	R. Evenepoel	SOQ	2799
2	A. Vlasov	BOH	2658
3	J. Ayuso	UAD	2636
4	M. Jorgenson	TVL	2382
5	P. Roglič	BOH	2155
6	C. Rodríguez	IGD	2137
7	J. Vingegaard	TVL	2103
8	B. McNulty	UAD	1961

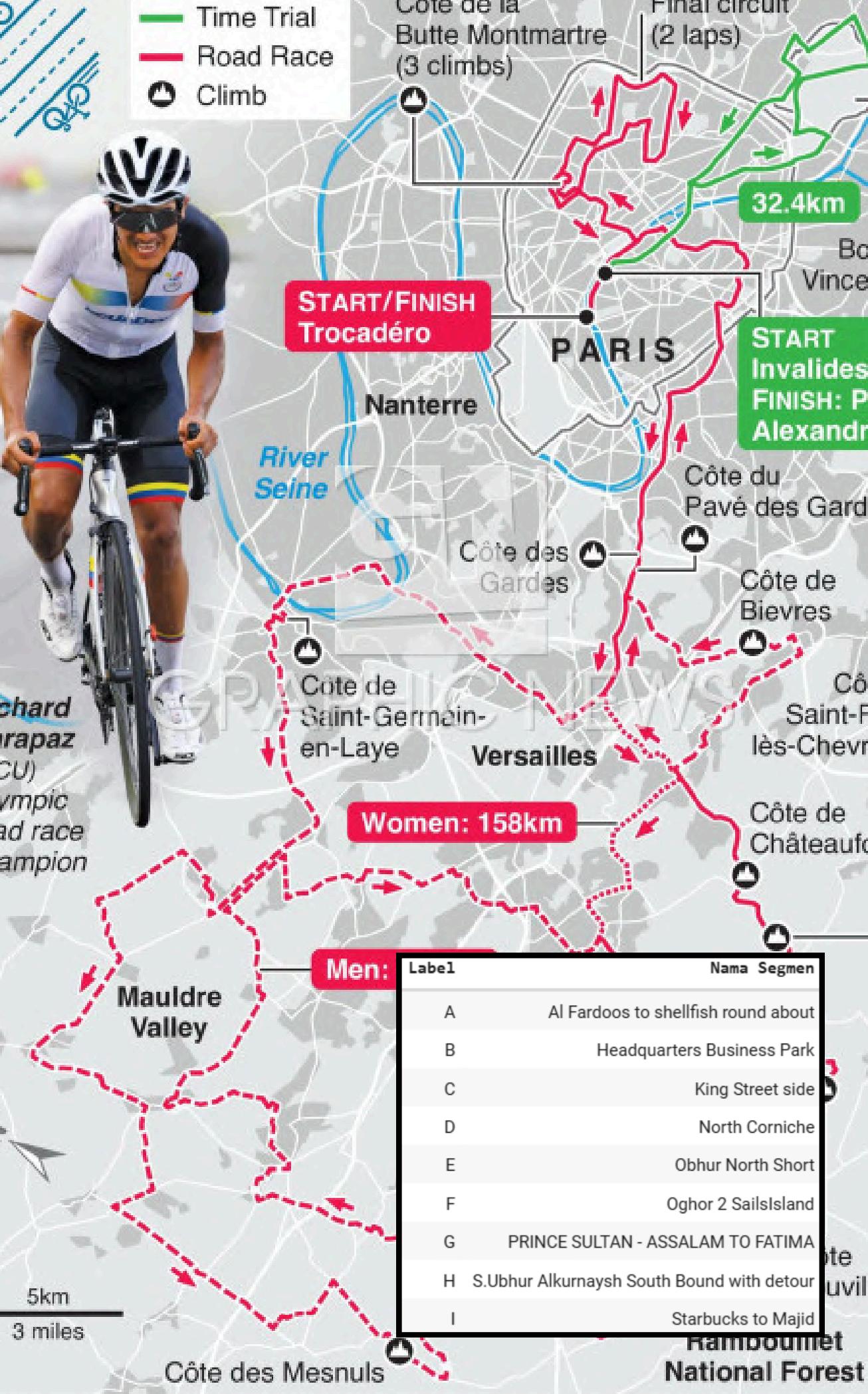
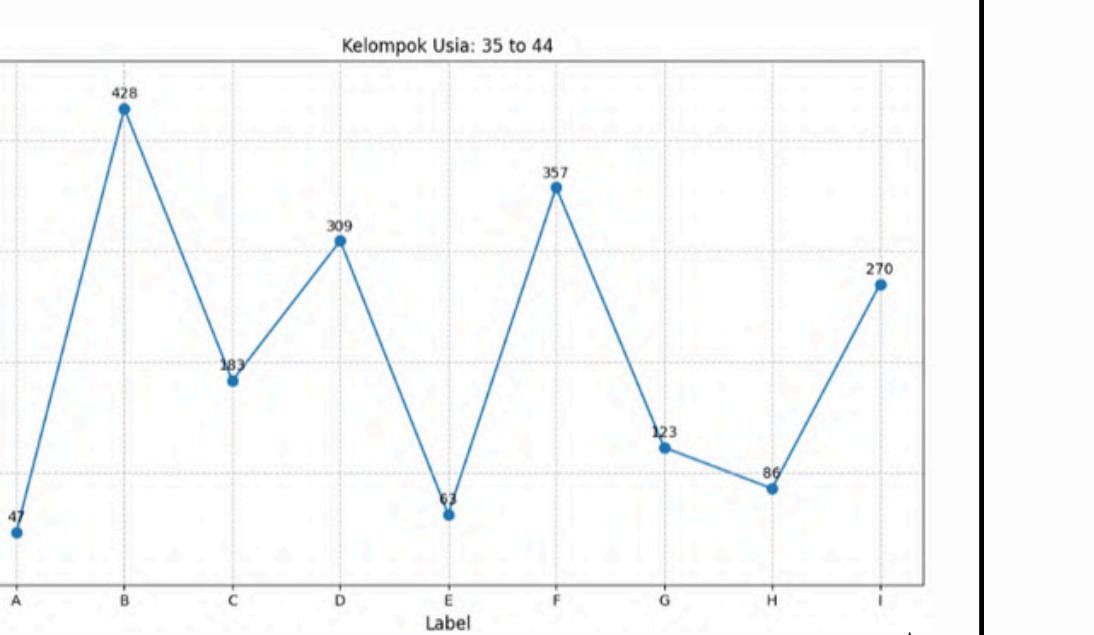
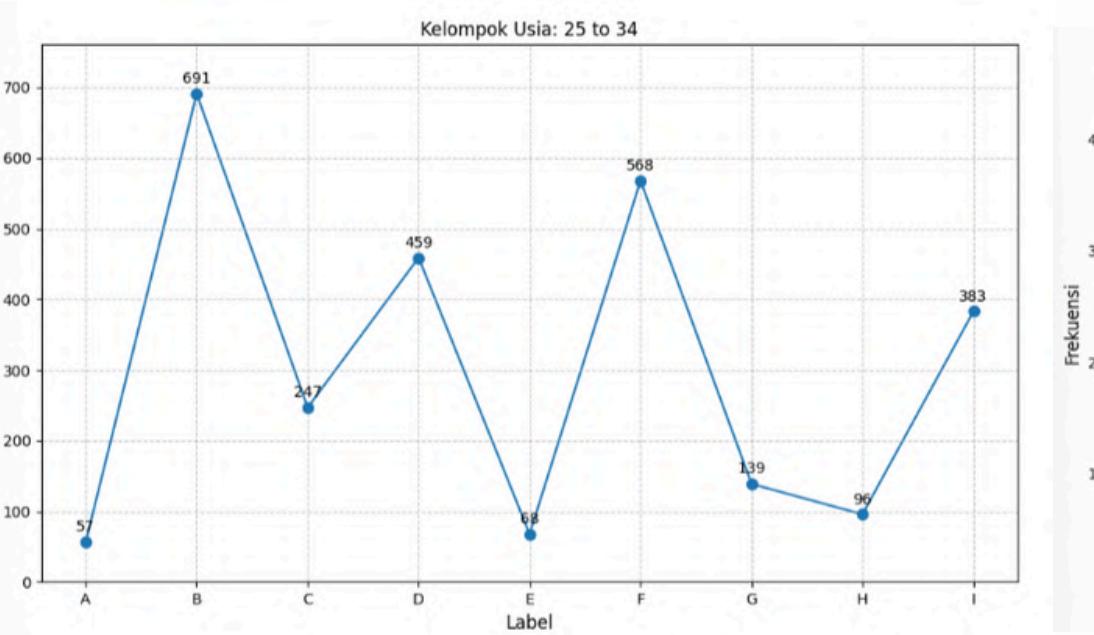
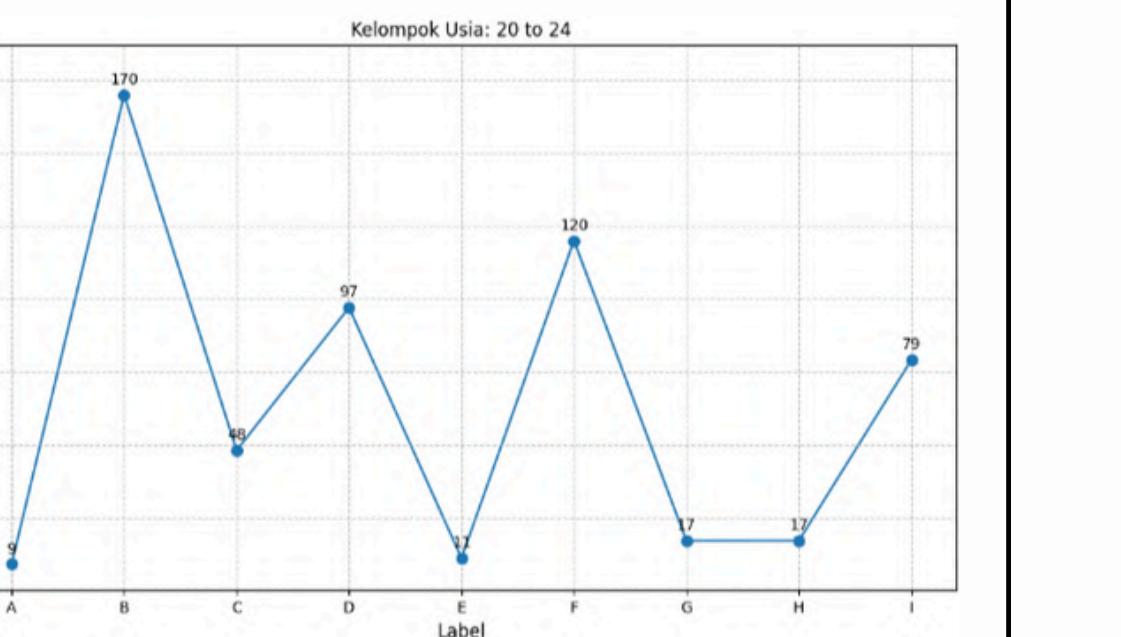
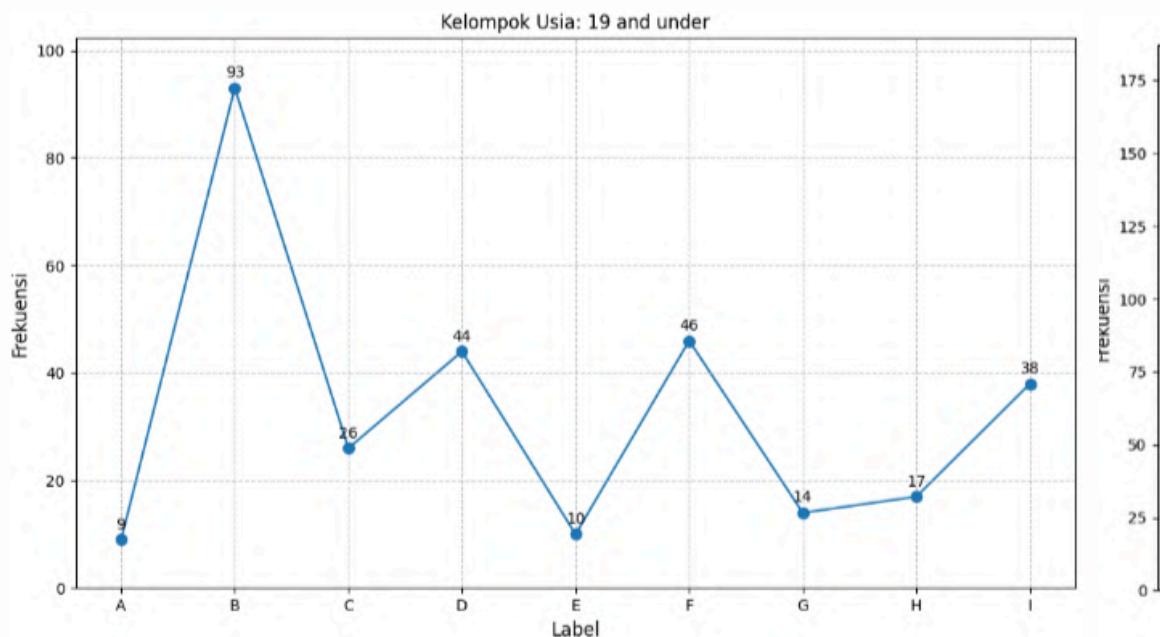
Ranking as of June 10, 2024

Remco Evenepoel

Road Code

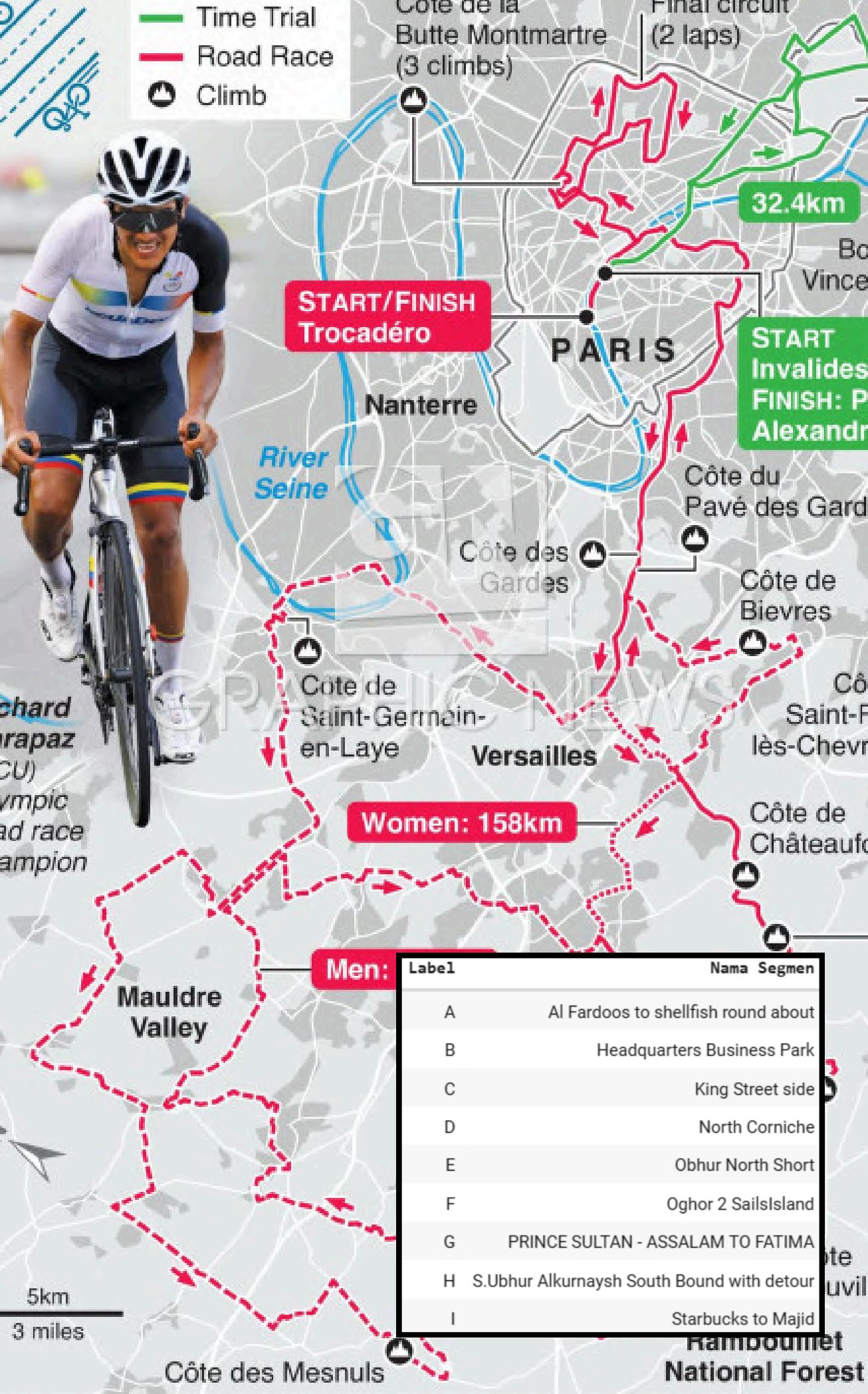
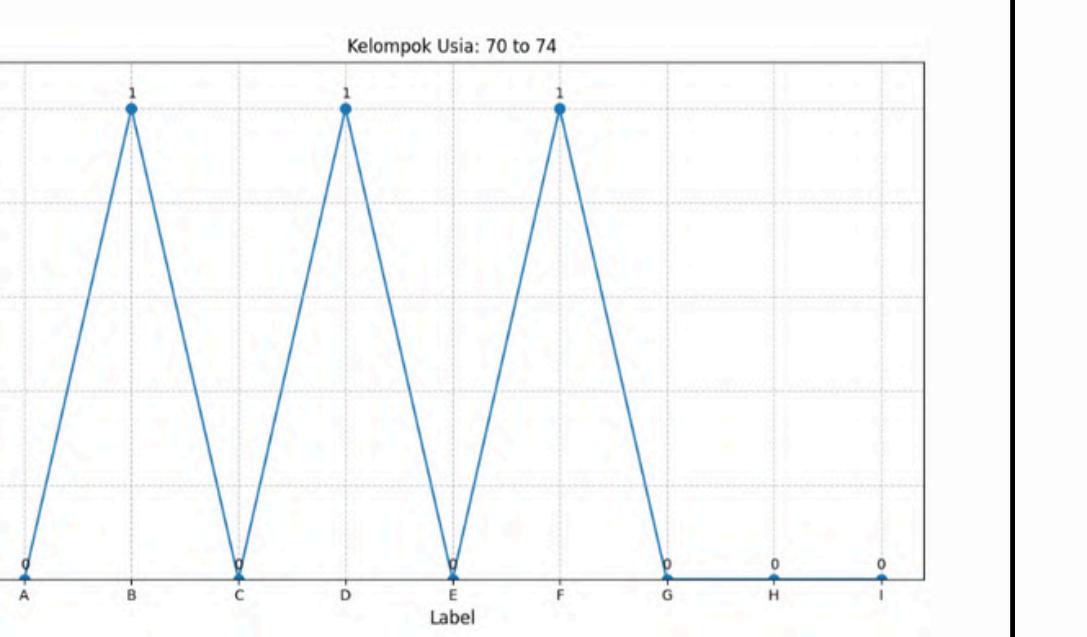
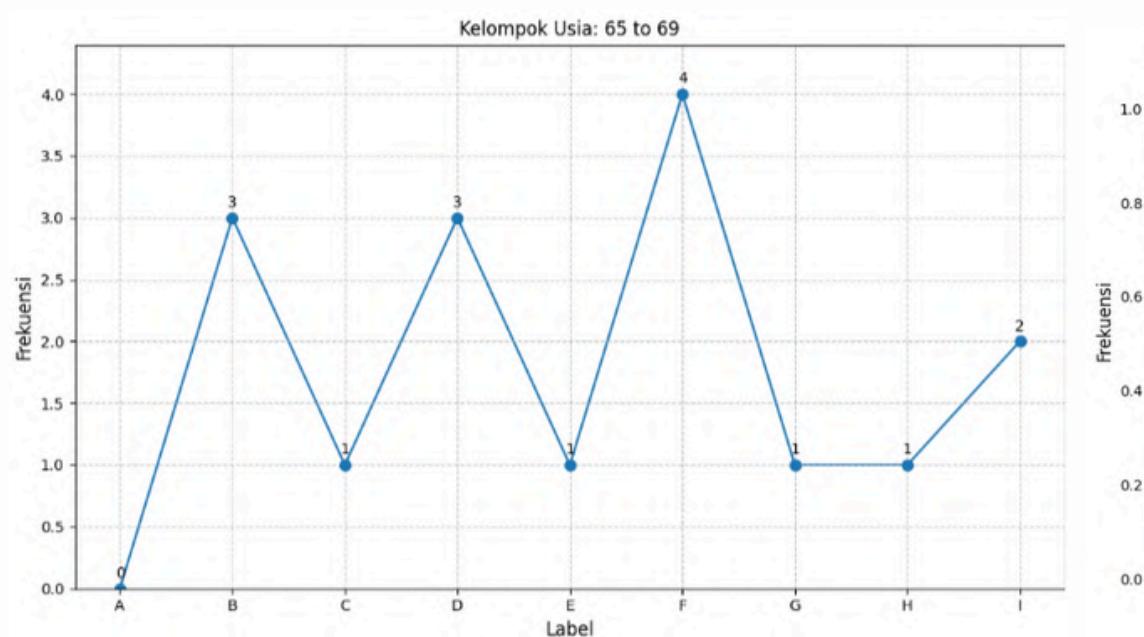
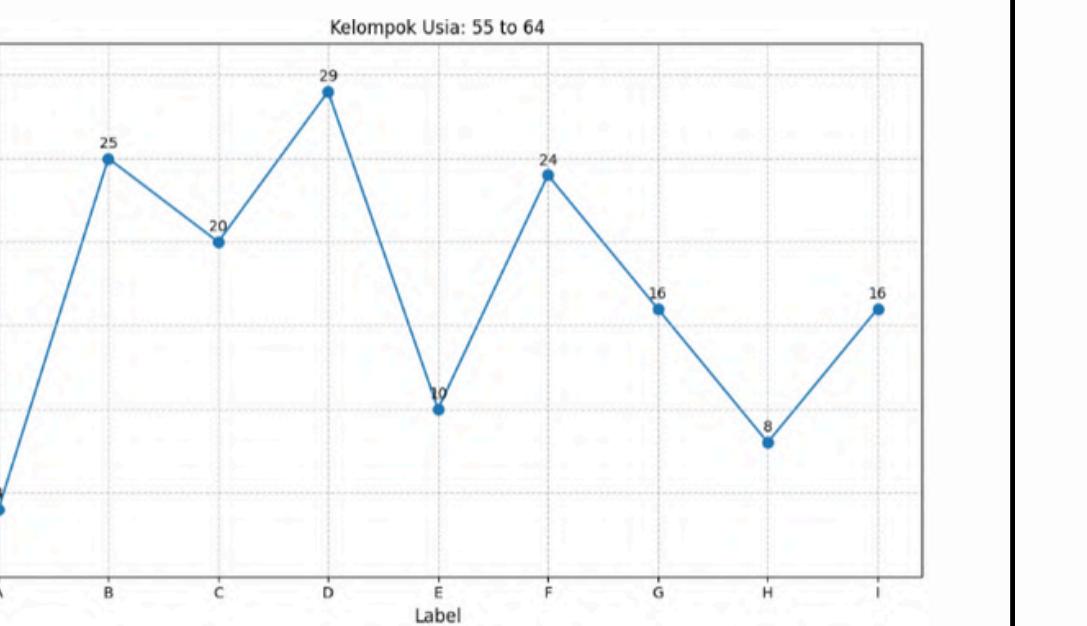
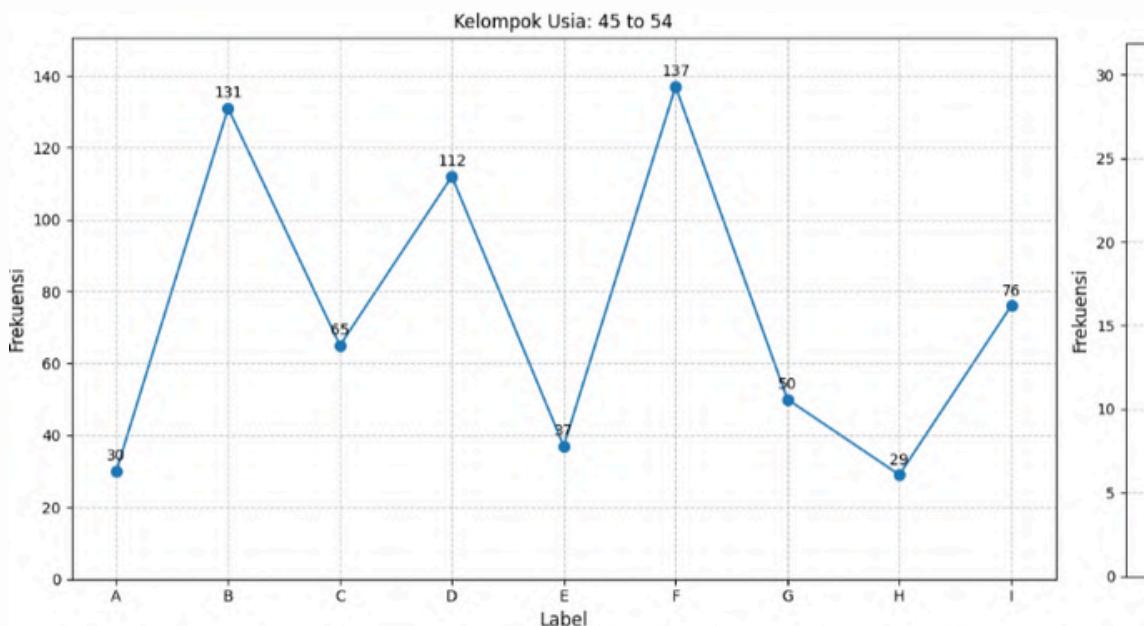
# 05

Segmen mana yang paling sering diikuti oleh peserta dari kategori usia tertentu ?



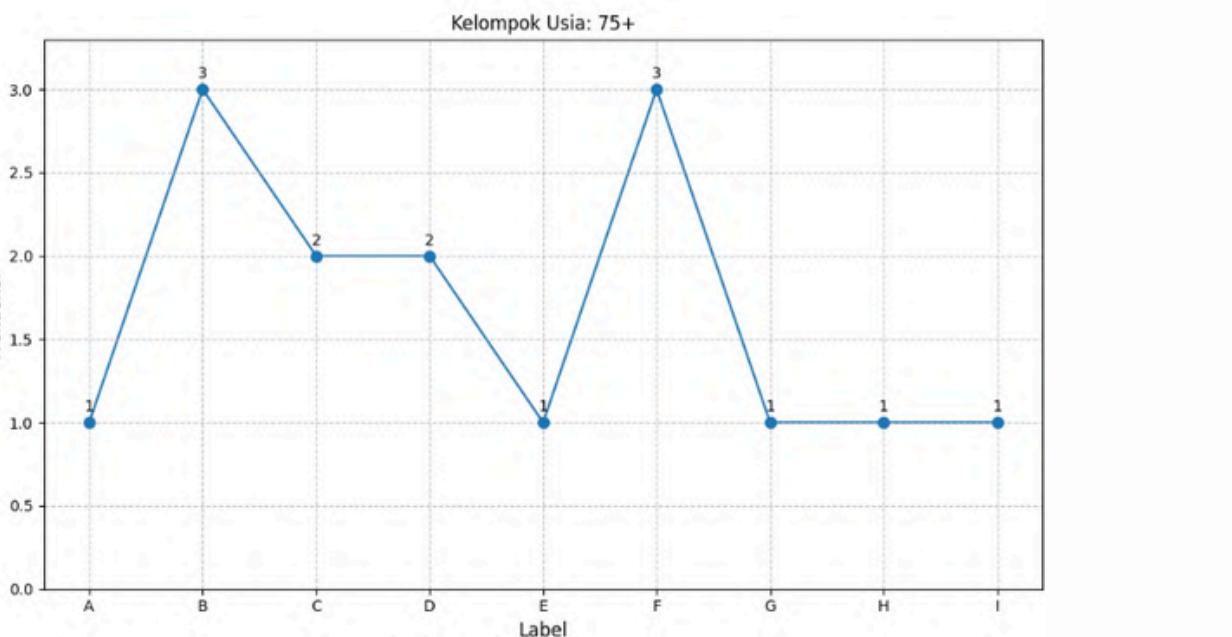
# 05

Segmen mana yang paling sering diikuti oleh peserta dari kategori usia tertentu ?



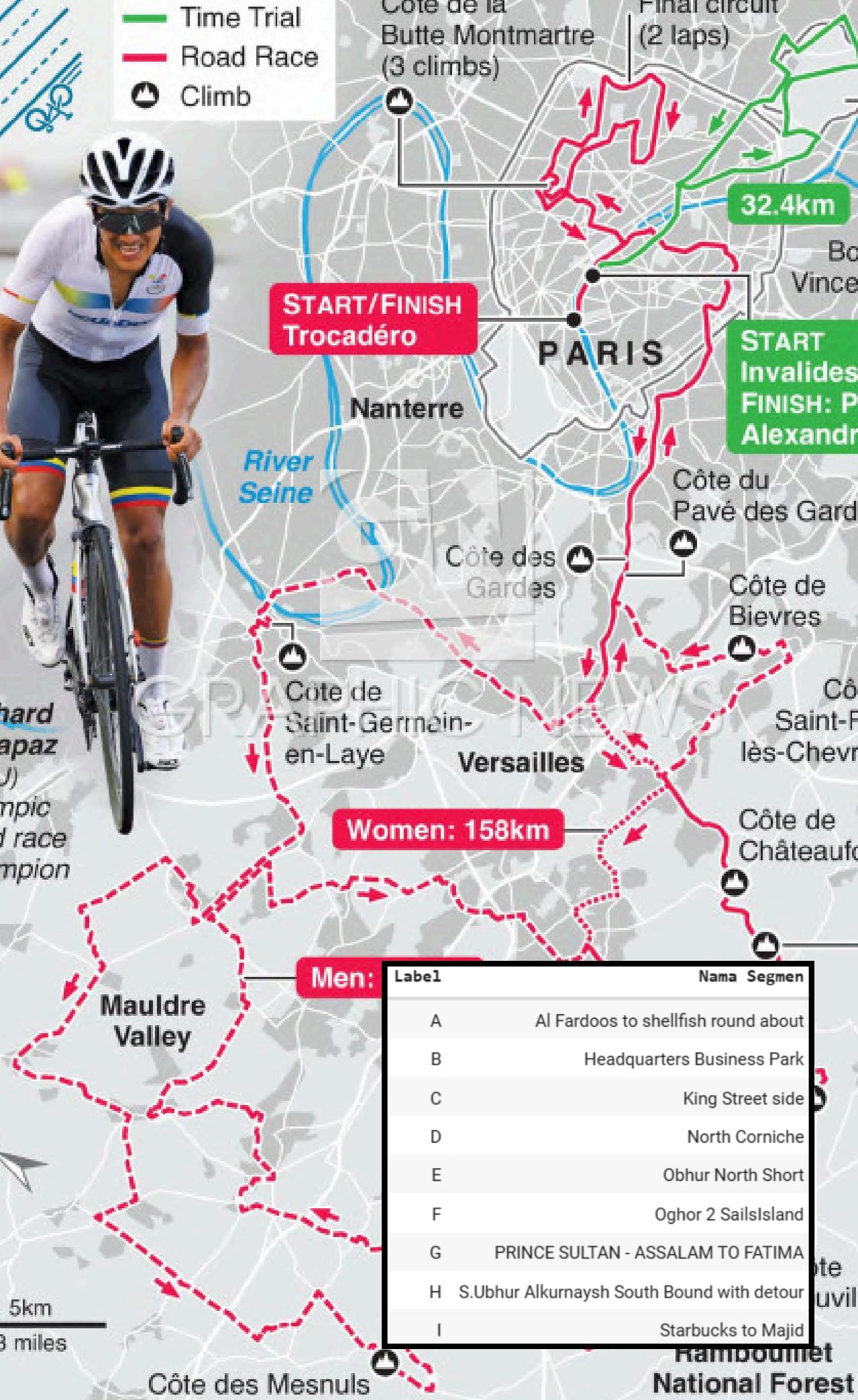
# 05

Segmen mana yang paling sering diikuti oleh peserta dari kategori usia tertentu ?



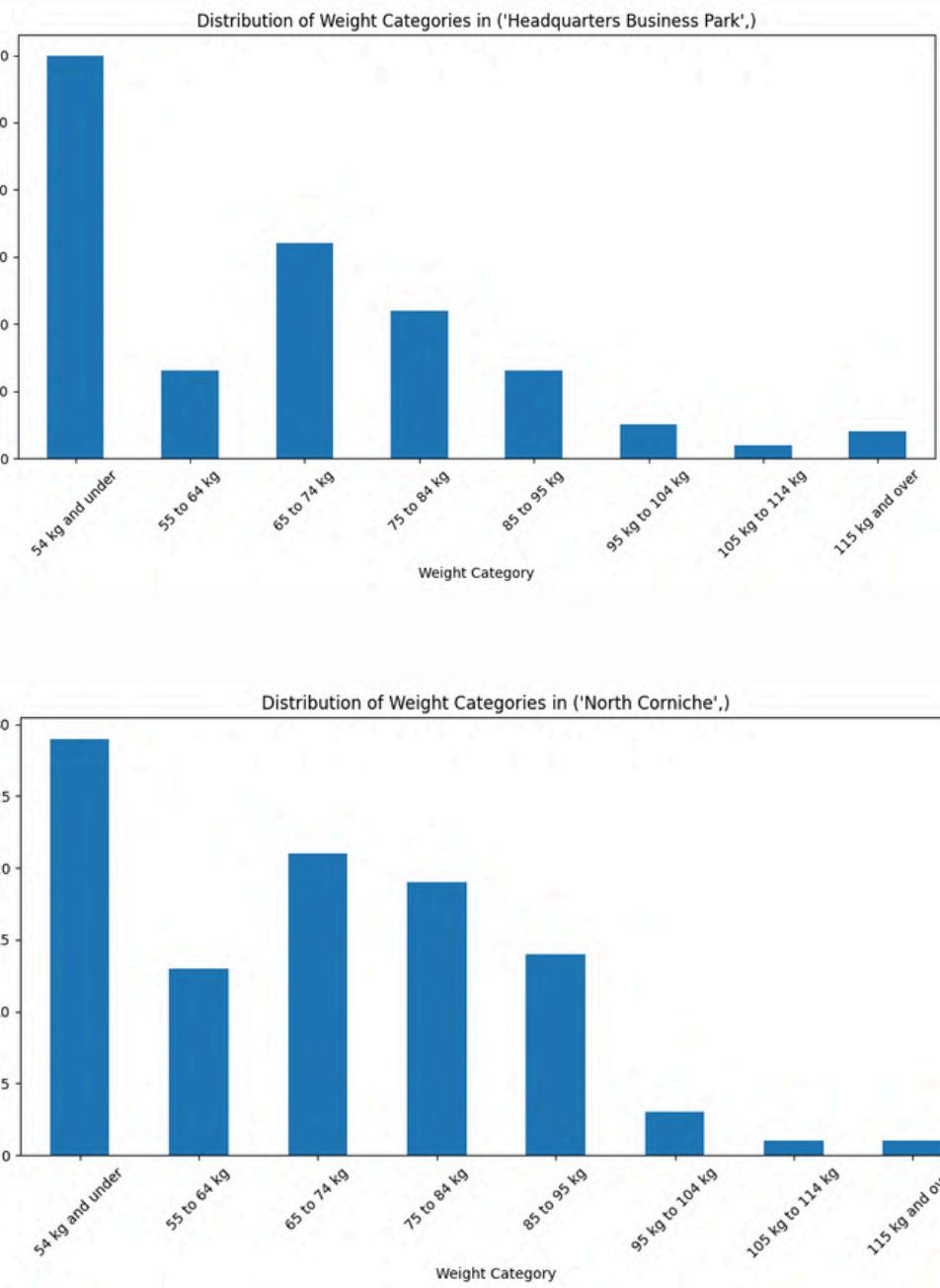
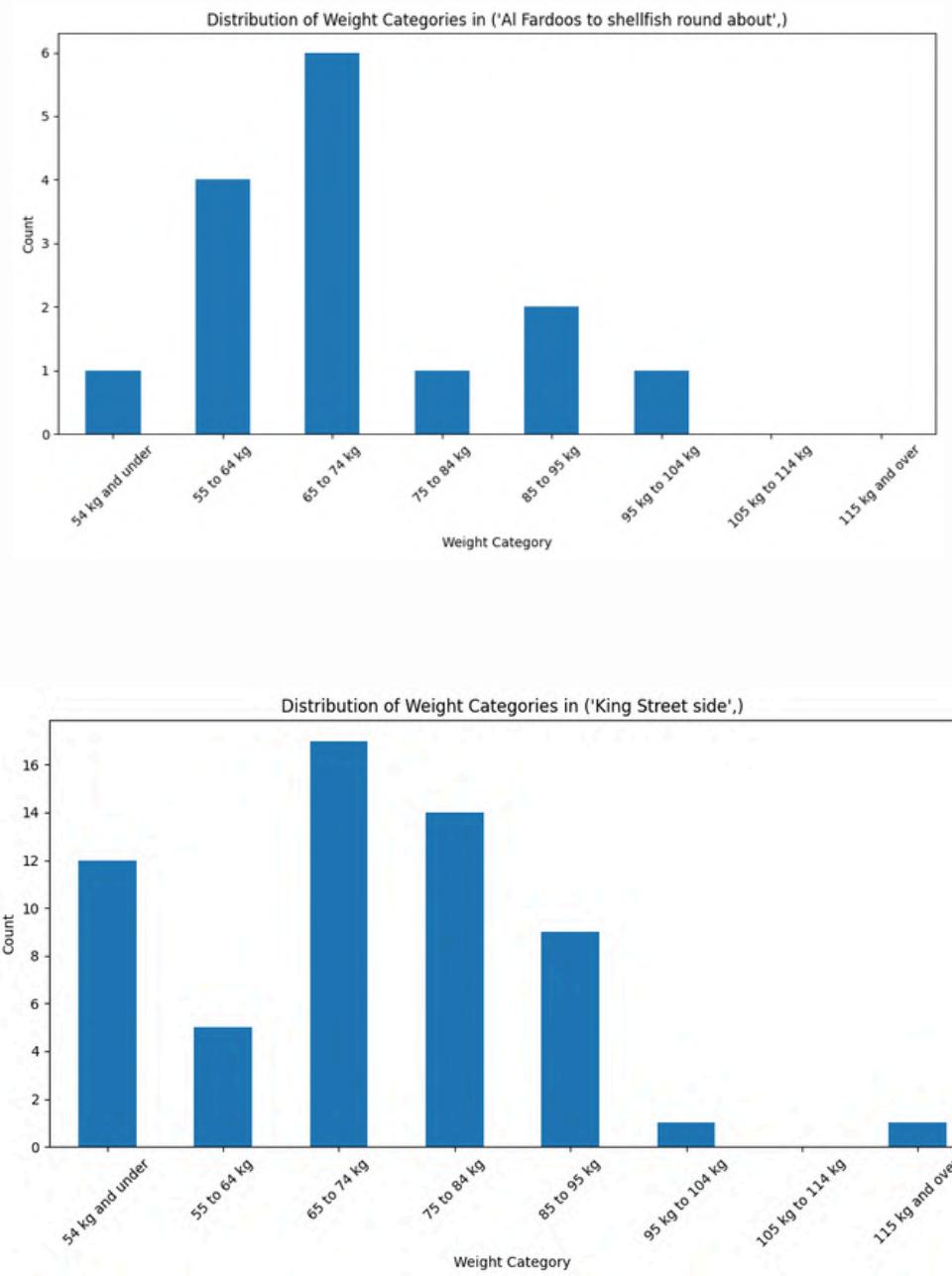
**Headquarters Business Park** adalah segmen utama yang paling sering diikuti oleh peserta dari kategori usia 19 and under, 20 to 24, 25 to 34, dan 35 to 44, dengan frekuensi tertinggi di antara segmen lainnya. **Oghor 2 SailsIsland** paling sering diikuti oleh kelompok usia 45 to 54 dan 65 to 69. Sementara itu, **North Corniche** sering diikuti oleh kelompok usia 55 to 64.

Pada kelompok usia 70 to 74, tidak ada segmen dominan, sedangkan pada usia 75+, Headquarters Business Park dan Oghor 2 SailsIsland memiliki daya tarik serupa.



# 06

Apakah peserta dari kategori berat badan tertentu memiliki kecenderungan lebih besar untuk mendominasi segmen tertentu (lebih banyak peserta dari kategori tersebut di peringkat 10%)?



Pada segmen **Al Fardoos to Shellfish Round About**, kategori berat badan dengan jumlah peserta terbanyak adalah **65 to 74 kg**, diikuti oleh **55 to 64 kg**. Sebaliknya, kategori dengan jumlah peserta terendah adalah **105 kg to 114 kg** dan **115 kg and over**, yang tidak memiliki peserta sama sekali.

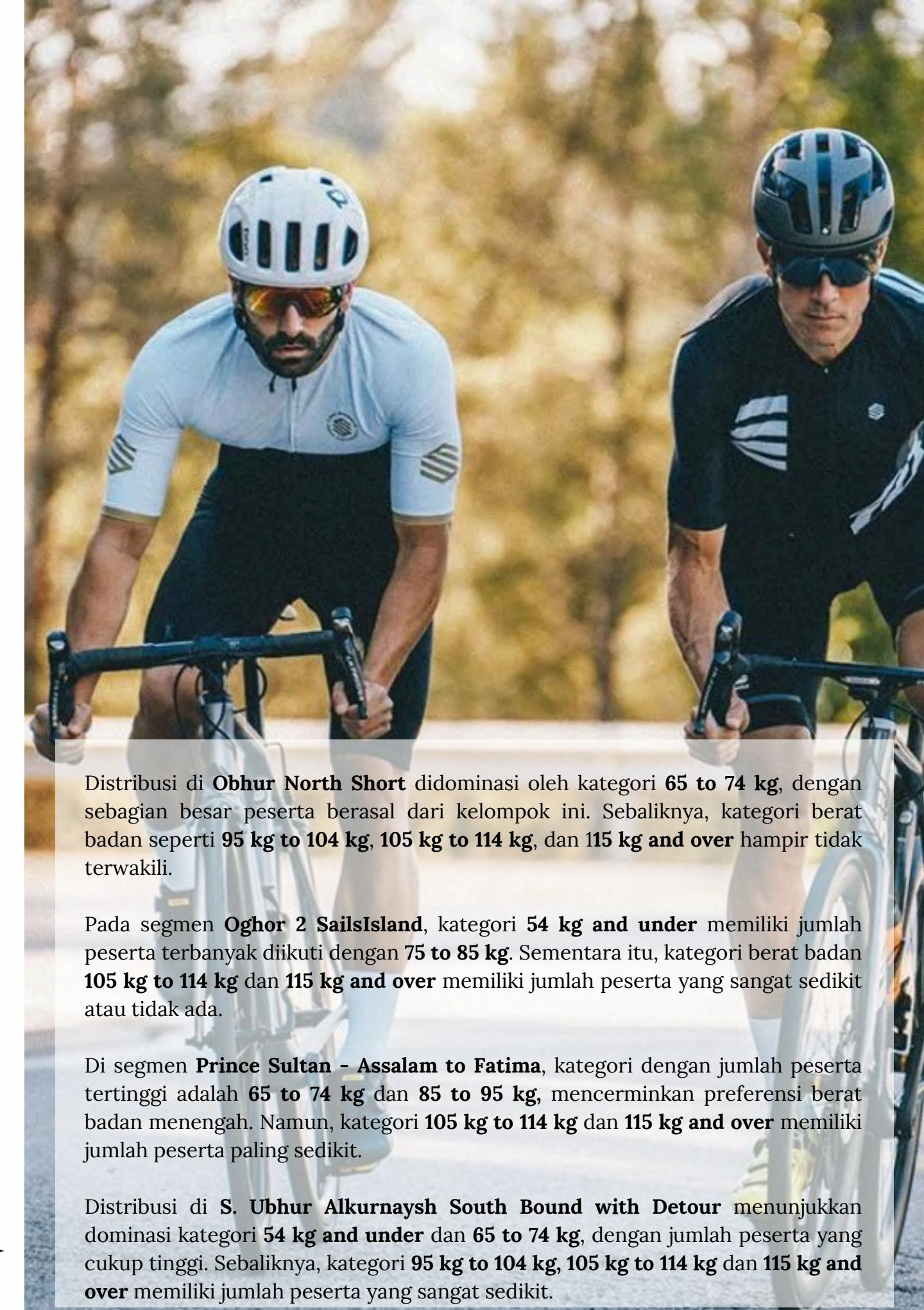
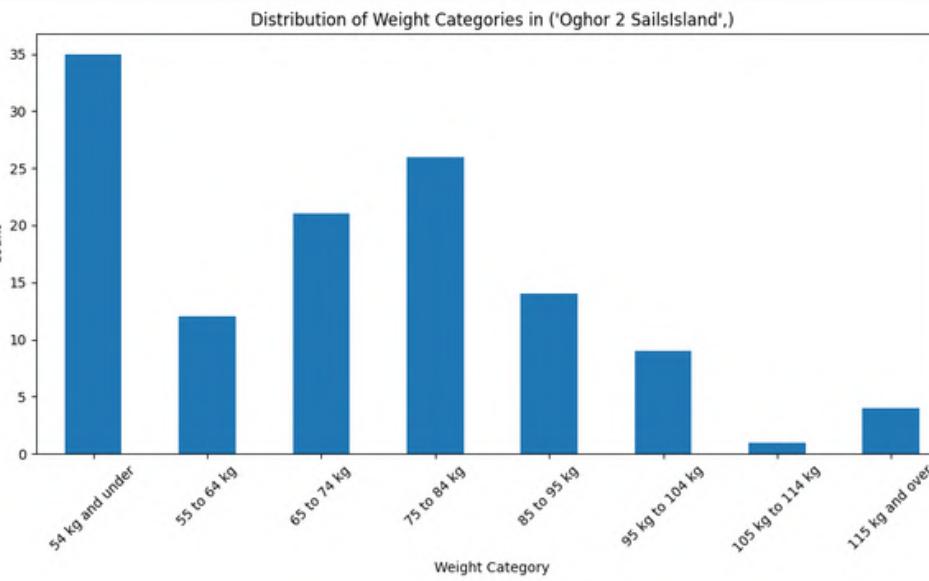
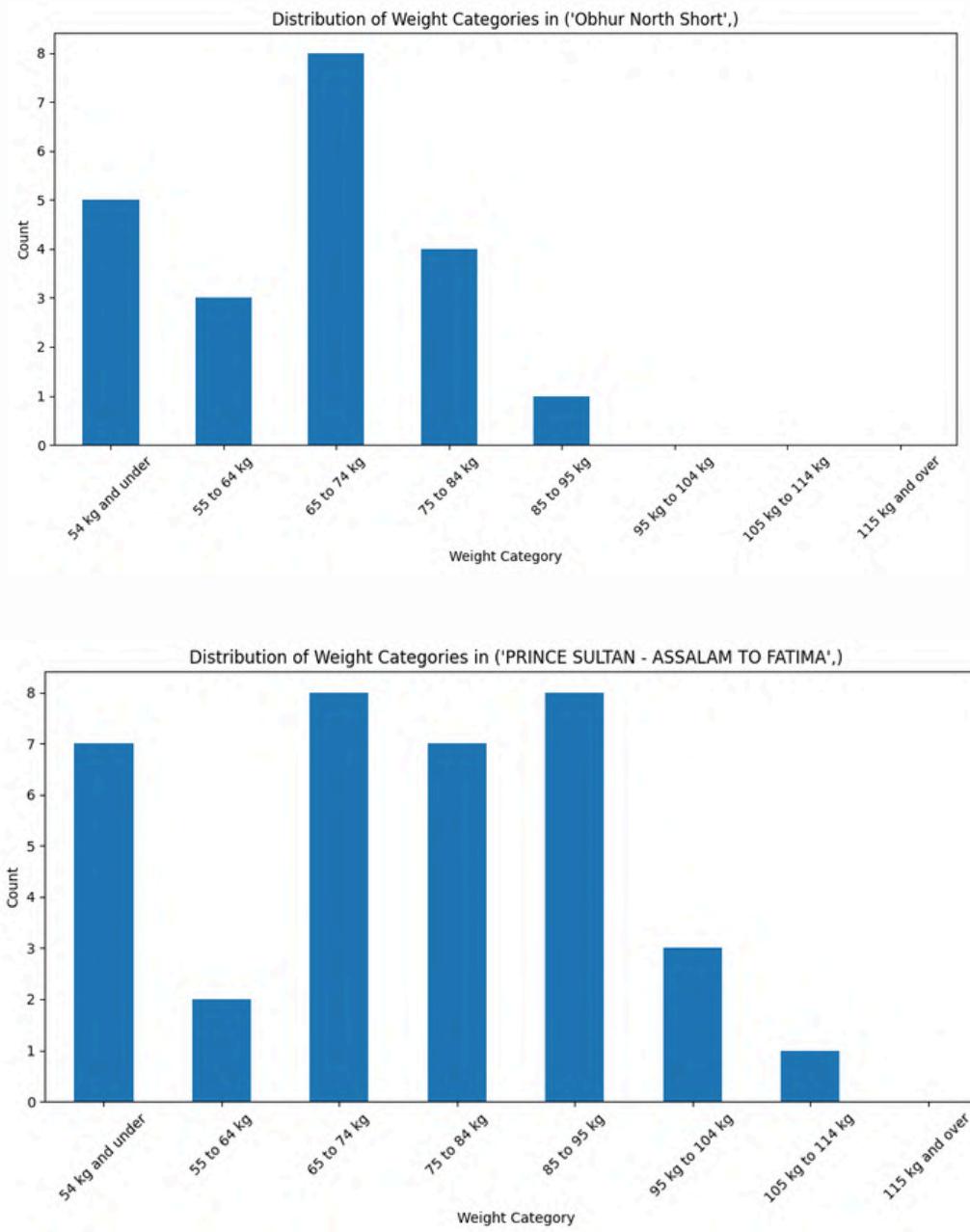
Distribusi kategori berat badan di **Headquarters Business Park** menunjukkan dominasi dari kategori **54 kg and under**, yang memiliki jumlah peserta terbanyak. Sebaliknya, kategori **105 kg to 114 kg** dan **115 kg and over** memiliki jumlah peserta paling sedikit atau bahkan tidak ada sama sekali.

Pada segmen **King Street Side**, kategori berat badan yang mendominasi adalah **65 to 74 kg** dan **75 to 84 kg**, menunjukkan konsentrasi peserta pada berat badan menengah. Sebaliknya, kategori **105 kg to 114 kg** dan **115 kg and over** memiliki jumlah peserta paling rendah.

Di **North Corniche**, kategori **54 kg and under** memiliki jumlah peserta terbanyak, diikuti oleh **65 to 74 kg**. Kategori berat badan yang lebih tinggi, seperti **105 kg to 114 kg** dan **115 kg and over**, memiliki jumlah peserta paling sedikit atau tidak ada.

# 06

Apakah peserta dari kategori berat badan tertentu memiliki kecenderungan lebih besar untuk mendominasi segmen tertentu (lebih banyak peserta dari kategori tersebut di peringkat 10%)?



Distribusi di **Obhur North Short** didominasi oleh kategori **65 to 74 kg**, dengan sebagian besar peserta berasal dari kelompok ini. Sebaliknya, kategori berat badan seperti **95 kg to 104 kg**, **105 kg to 114 kg**, dan **115 kg and over** hampir tidak terwakili.

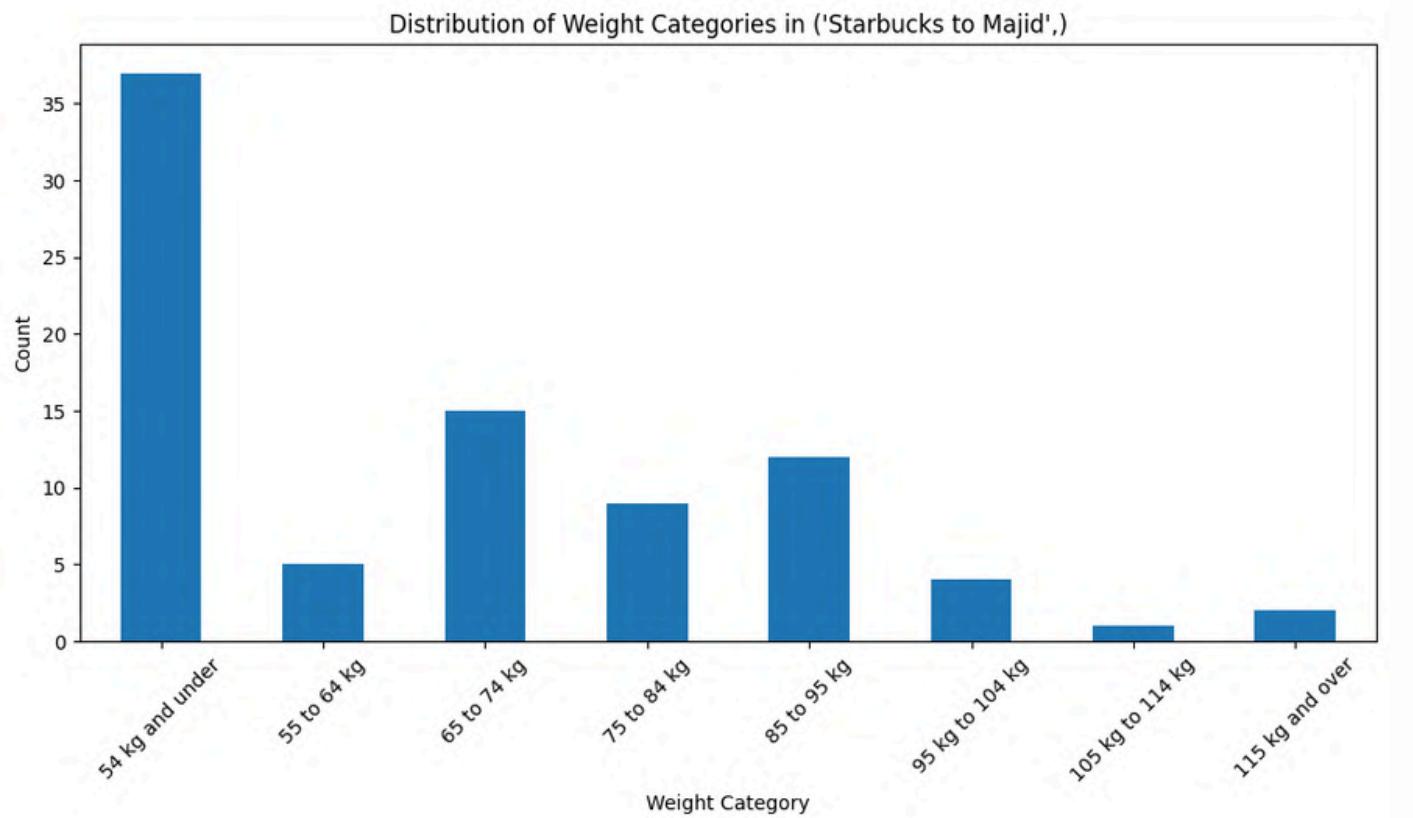
Pada segmen **Oghor 2 SailsIsland**, kategori **54 kg and under** memiliki jumlah peserta terbanyak diikuti dengan **75 to 85 kg**. Sementara itu, kategori berat badan **105 kg to 114 kg** dan **115 kg and over** memiliki jumlah peserta yang sangat sedikit atau tidak ada.

Di segmen **Prince Sultan - Assalam to Fatima**, kategori dengan jumlah peserta tertinggi adalah **65 to 74 kg** dan **85 to 95 kg**, mencerminkan preferensi berat badan menengah. Namun, kategori **105 kg to 114 kg** dan **115 kg and over** memiliki jumlah peserta paling sedikit.

Distribusi di **S. Ubhur Alkurnaysh South Bound with Detour** menunjukkan dominasi kategori **54 kg and under** dan **65 to 74 kg**, dengan jumlah peserta yang cukup tinggi. Sebaliknya, kategori **95 kg to 104 kg**, **105 kg to 114 kg** dan **115 kg and over** memiliki jumlah peserta yang sangat sedikit.

# 06

Apakah peserta dari kategori berat badan tertentu memiliki kecenderungan lebih besar untuk mendominasi segmen tertentu (lebih banyak peserta dari kategori tersebut di peringkat 10%)?



# Encoding

## Mapping

Pada Fitur *user\_age\_group* , *user\_weight\_category*, *smt\_name* dilakukan encoding dengan ketentuan sebagai berikut :

```
age_mapping = {  
    '19 and under': 0,  
    '20 to 24': 1,  
    '25 to 34': 2,  
    '35 to 44': 3,  
    '45 to 54': 4,  
    '55 to 64': 5,  
    '65 to 69': 6,  
    '70 to 74': 7,  
    '75+': 8  
}  
  
df_dt['user_age_group'] = df_dt['user_age_group'].map(age_mapping)  
df_dt.head()
```

```
mapping = {  
    'Starbucks to Majid': 0,  
    'King Street side': 1,  
    'Al Fardoos to shellfish round about': 2,  
    'Headquarters Business Park': 3,  
    'PRINCE SULTAN - ASSALAM TO FATIMA': 4,  
    'Oghor 2 SailsIsland': 5,  
    'Obhur North Short': 6,  
    'North Corniche': 7,  
    'S.Ubhur Alkurnayash South Bound with detour': 8  
}  
  
df_dt['smt_name'] = df_dt['smt_name'].map(mapping)  
df_dt.head()
```

```
# Mapping untuk User Weight Category  
mapping = {  
    '54 kg and under': 0,  
    '55 to 64 kg': 1,  
    '65 to 74 kg': 2,  
    '75 to 84 kg': 3,  
    '85 to 95 kg': 4,  
    '95 kg to 104 kg': 5,  
    '105 kg to 114 kg': 6,  
    '115 kg and over': 7  
}  
  
df_dt['user_weight_category'] = df_dt['user_weight_category'].map(mapping)  
df_dt
```

```
| df_dt = pd.get_dummies(df_dt, columns=['smt_name'])  
df_dt.head()
```



▲ terdapat metode encode *smt\_name* menggunakan One-Hot Encoding, sementara sisanya di-mapping disesuaikan dengan sifat model.

# Encoding

## Frequency Encoding

Melakukan encoding fitur *act\_title* dengan cara setiap kategori digantikan dengan frekuensi atau jumlah kemunculan kategori tersebut dalam dataset.

```
# Frequency Encoding untuk act_title
freq_map = df_dt['act_title'].value_counts(normalize=True)
df_dt['act_title'] = df_dt['act_title'].map(freq_map)
df_dt.head()
```

## Encoding attempt date

Melakukan encoding dengan memisahkan fitur *attempt\_date* menjadi tiga fitur terpisah, yaitu *attempt\_year*, *attempt\_month*, *attempt\_day*.

```
#encoding attempt_date
df_dt['attempt_date'] = pd.to_datetime(df_dt['attempt_date'])

df_dt['attempt_year'] = df_dt['attempt_date'].dt.year
df_dt['attempt_month'] = df_dt['attempt_date'].dt.month
df_dt['attempt_day'] = df_dt['attempt_date'].dt.day

df_dt = df_dt.drop('attempt_date', axis=1)
df_dt.head()
```

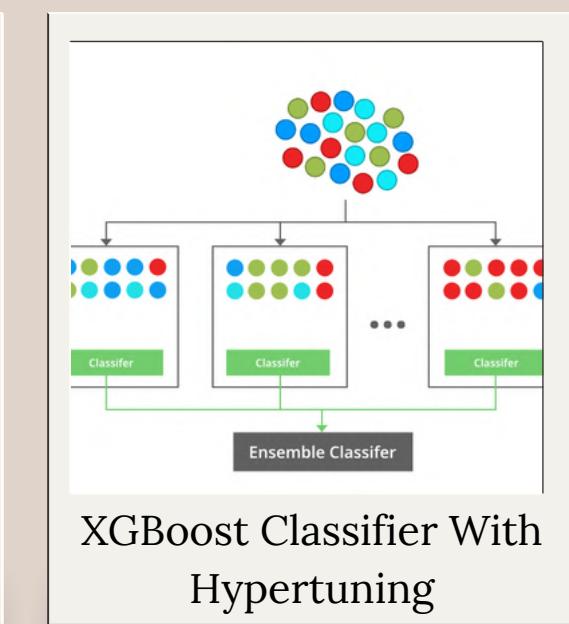
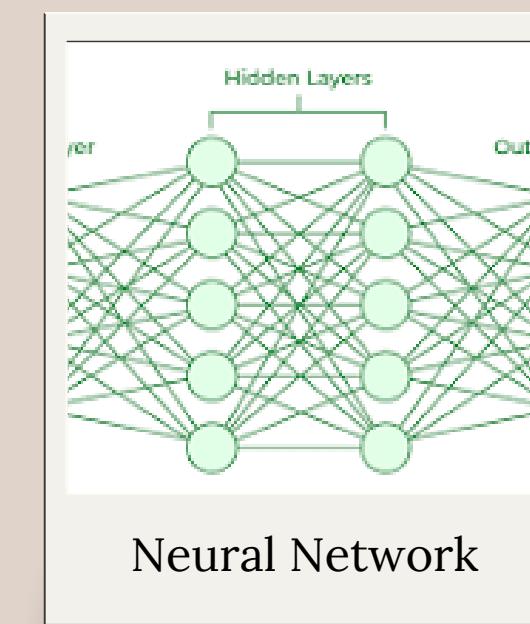
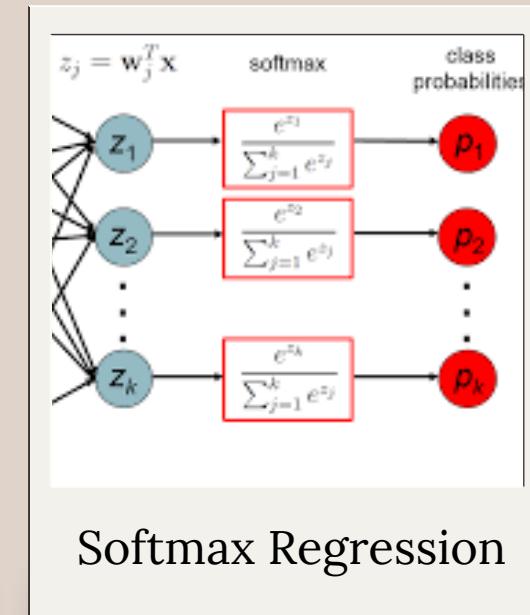
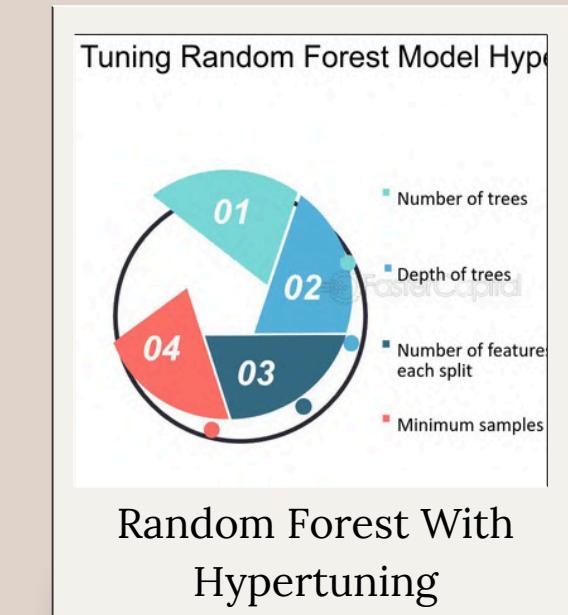
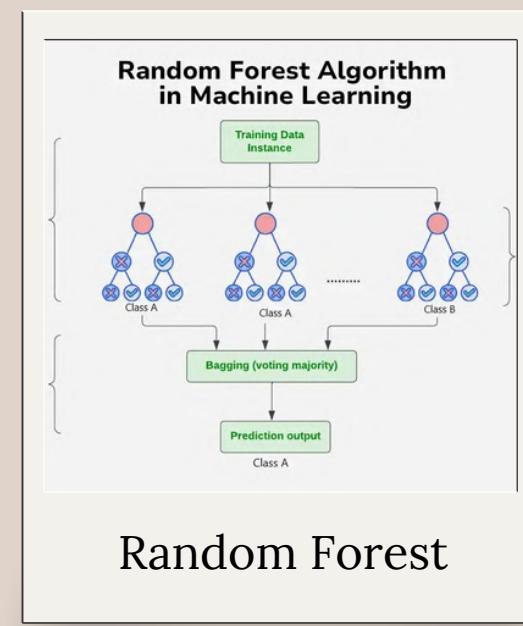
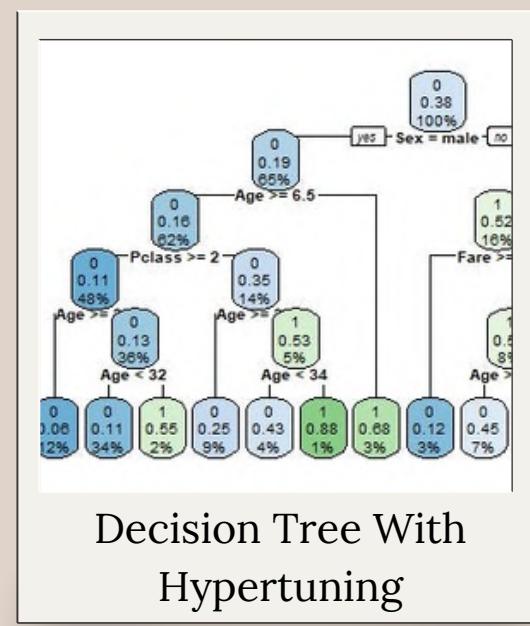
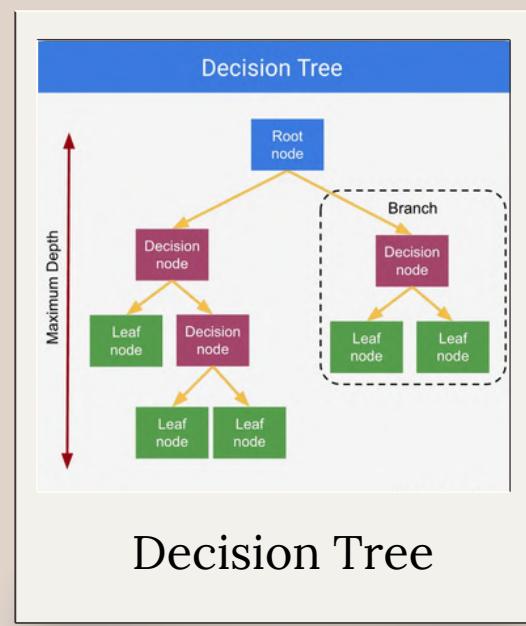


# Modelling



# Klasifikasi

Klasifikasi merupakan salah satu model machine learning yang bertugas untuk memprediksi jenis atau kelas suatu objek dalam sejumlah opsi yang terbatas.



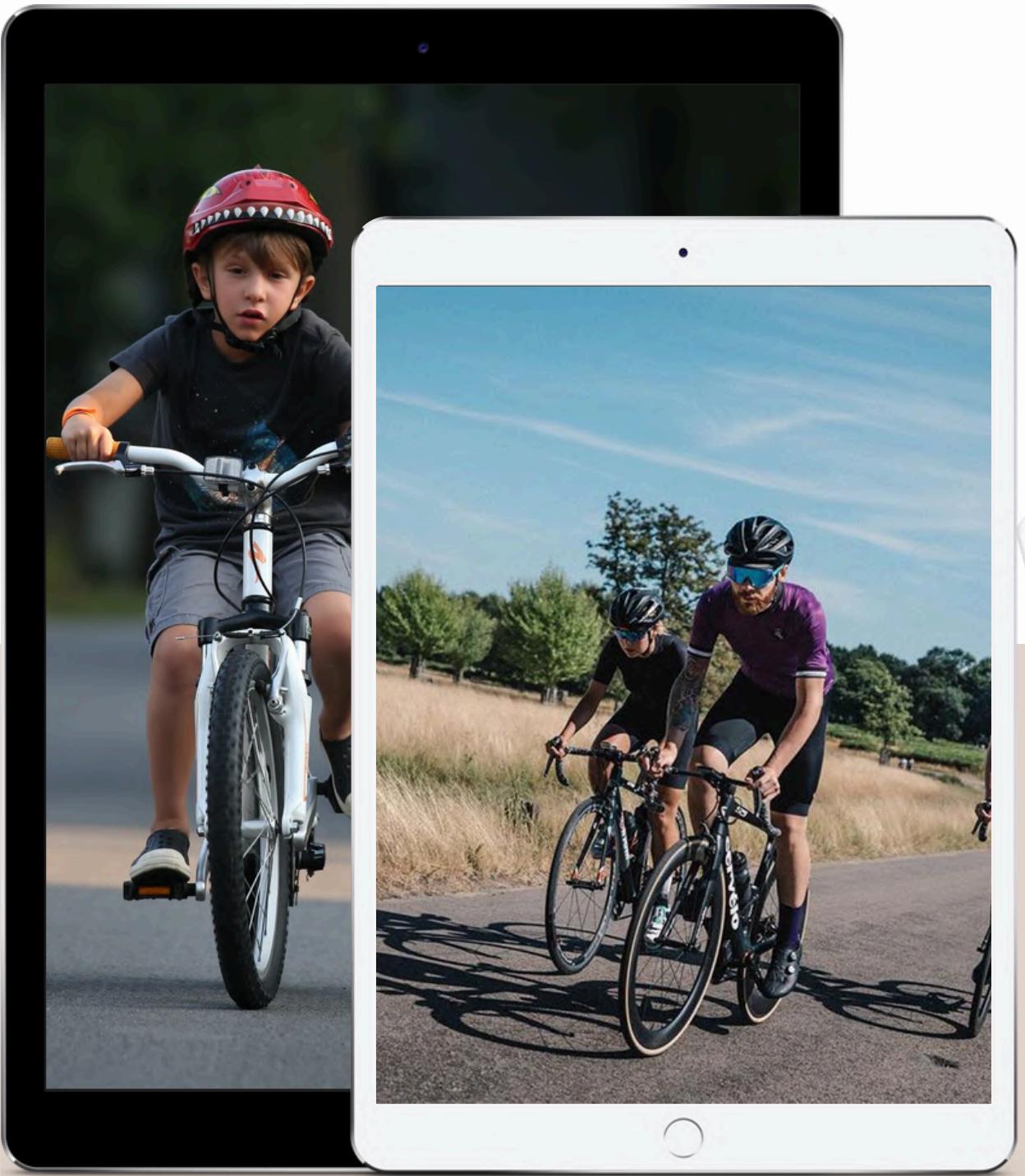
# Hasil Prediksi Model

	Accuracy	F1 Macro	F1 Micro	Precision Macro	Precision Micro	Recall Macro	Recall Micro
Decision Tree	0.9936	0.9606	0.9936	0.9446	0.9936	0.9779	0.9936
Decision Tree*	0.9936	0.9598	0.9936	0.9515	0.9936	0.9685	0.9936
Random Forest	0.9947	0.9653	0.9947	0.9713	0.9947	0.9594	0.9947
Random Forest*	0.9936	0.9589	0.9936	0.9588	0.9936	0.9589	0.9936
Softmax Regression	0.9878	0.9129	0.9878	0.9620	0.9878	0.8739	0.9878
Neural Network	0.9942	0.9616	0.9942	0.9707	0.9942	0.9528	0.9942
XGBoost Classifier*	0.9960	0.9751	0.9960	0.9625	0.9960	0.9885	0.9960

Notes: \* means with hypertuning

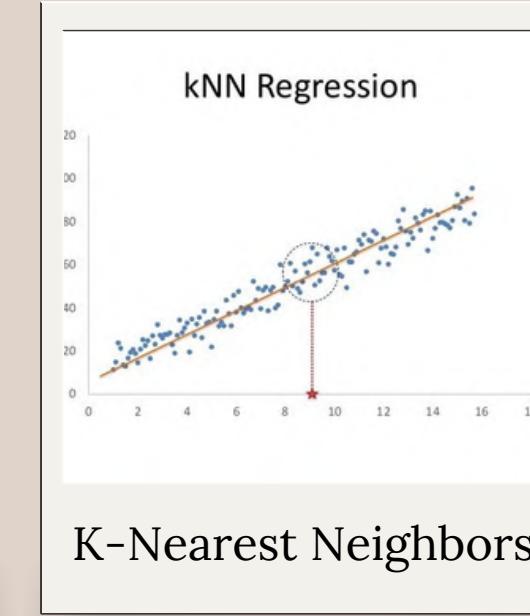
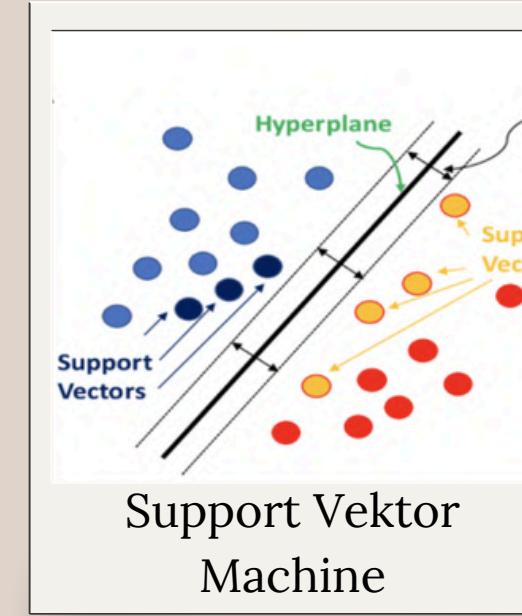
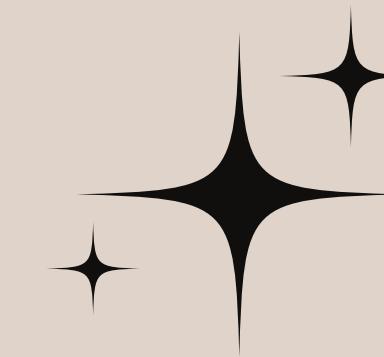
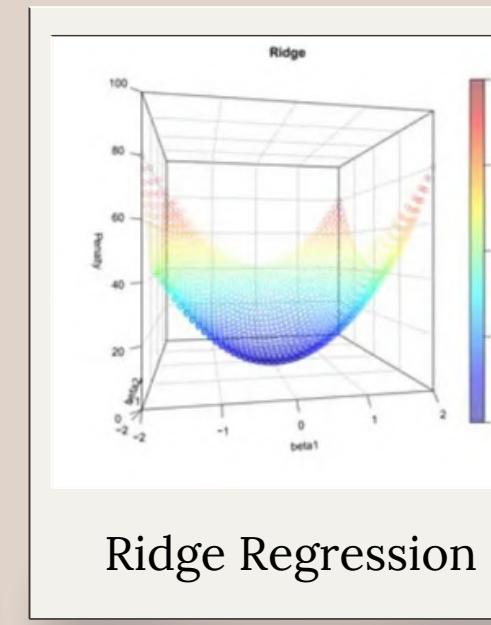
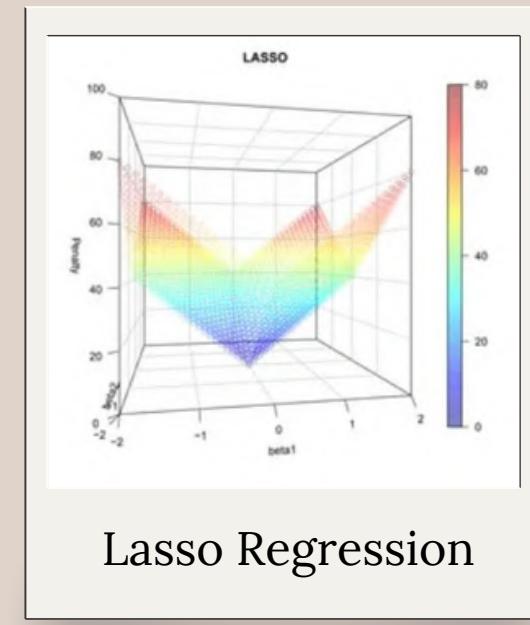
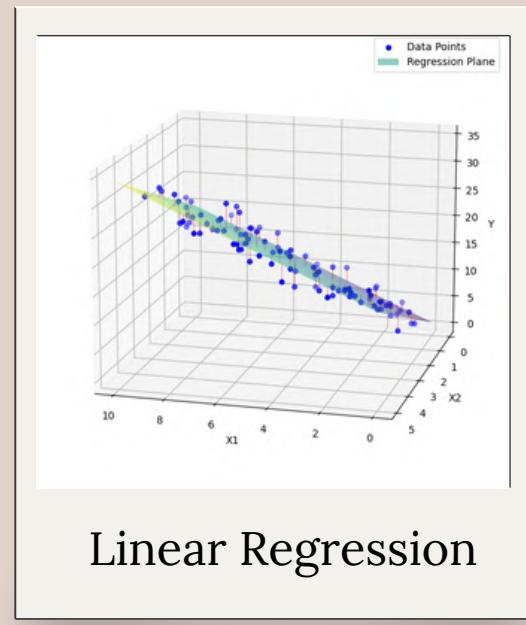
# Insight Yang Diperoleh

- Berdasarkan F1 Macro, model XGBoost Classifier unggul dengan skor 0.9751, menunjukkan keseimbangan terbaik antara Precision dan Recall pada semua kelas.
- Random Forest (tanpa tuning) berada di posisi kedua dengan skor 0.9653, menunjukkan bahwa model ini efektif menangkap pola kompleks, meskipun versi dengan tuning memiliki skor lebih rendah (0.9589) karena kemungkinan tuning yang kurang optimal.
- Neural Network mencapai skor 0.9616, menunjukkan performa baik tetapi masih di bawah XGBoost, Hal ini disebabkan oleh kebutuhan tuning yang lebih sensitif.
- Decision Tree (tanpa tuning) memiliki skor 0.9606, yang sedikit menurun menjadi 0.9598 setelah tuning, menandakan peningkatan Precision tetapi dengan penurunan Recall.
- Softmax Regression memiliki skor terendah, yaitu 0.9129, karena sifat liniernya yang kurang mampu menangani hubungan non-linear dalam data.

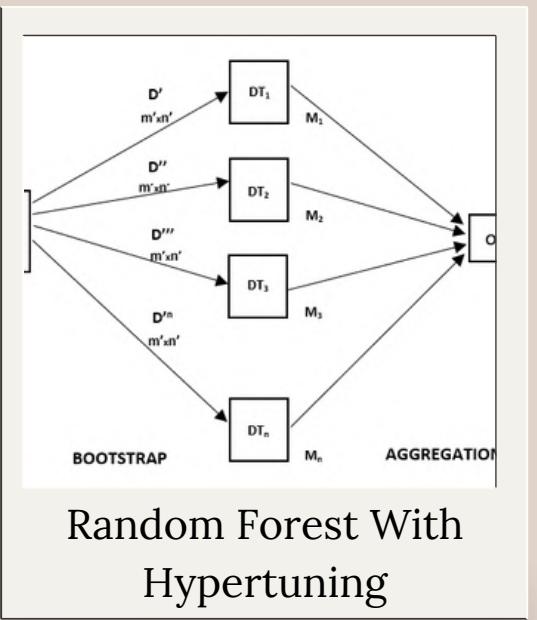
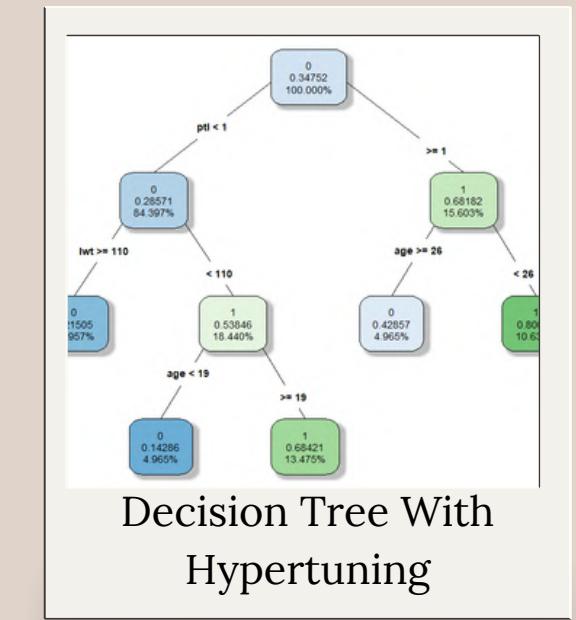


# Regresi

Regresi merupakan salah satu jenis tugas supervised learning yang bertujuan untuk memprediksi nilai kontinu dengan mempelajari hubungan antara fitur (input) dan target (output) dari data yang ada.



K-Nearest Neighbors



# Hasil Prediksi Model

	R squared	MAE	MSE	RMSE
Linear Regression	0.8745	65.0244	11505.0933	107.2618
Lasso Regression	0.8745	65.0183	11505.5208	107.2638
Ridge Regression	0.7500	106.63	33510.57	183.06
K-Nearest Neighbors*	0.8750	43.0425	14001.1380	118.3264
Random Forest*	0.9390	36.02	8174.51	90.41

Notes: \*  
means with  
hipertuning

# Insight yang Diperoleh

- Random Forest dengan hypertuning menjadi model terbaik, dengan R-squared 0.9390 dan MAE 36.02. Model ini unggul dalam menangkap hubungan non-linear, interaksi kompleks, dan noise data. Optimisasi dengan hyperparameter memberikan dampak signifikan pada hasil.
  - Linear dan Lasso Regression kurang optimal, dengan R-squared 0.8745 dan MAE sekitar 65, menunjukkan pola non-linear dalam data tidak dapat ditangkap dengan baik oleh model linier.
- 
- Ridge Regression memiliki performa terendah karena model ini hanya cocok untuk hubungan linier. Pada data dengan pola non-linear atau interaksi kompleks, Ridge tidak mampu merepresentasikan pola dengan baik, sehingga terjadi underfitting, menghasilkan R-squared rendah (0.7500) dan MAE tinggi (106.63).
  - KNN memiliki performa yang cukup baik dengan R-squared 0.8750 dan MAE 43.04, namun model ini sensitif terhadap distribusi data dan parameter jumlah tetangga ( $k$ ). Sensitivitas ini membuat performanya lebih rendah dibandingkan Random Forest yang lebih stabil terhadap variasi data.



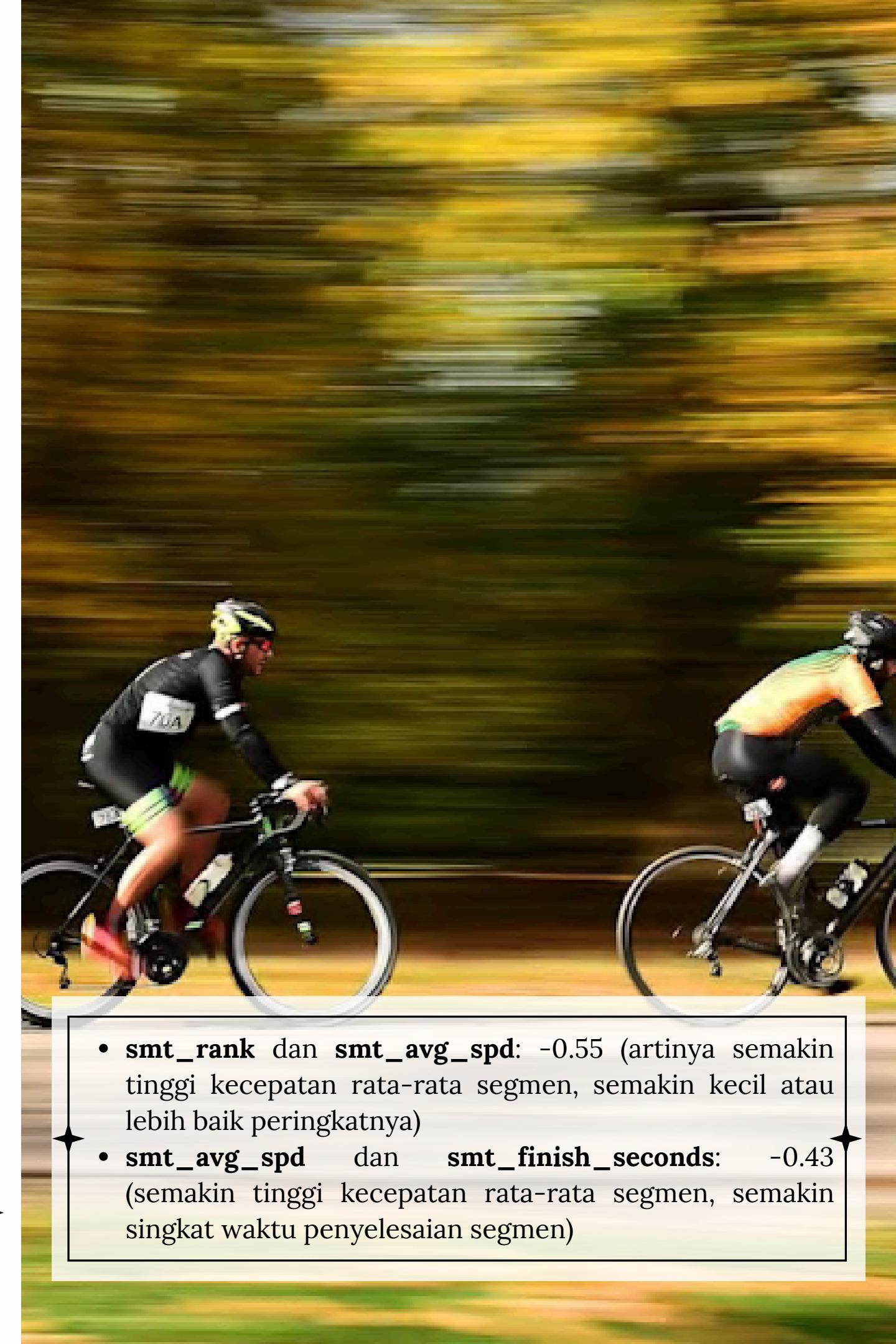
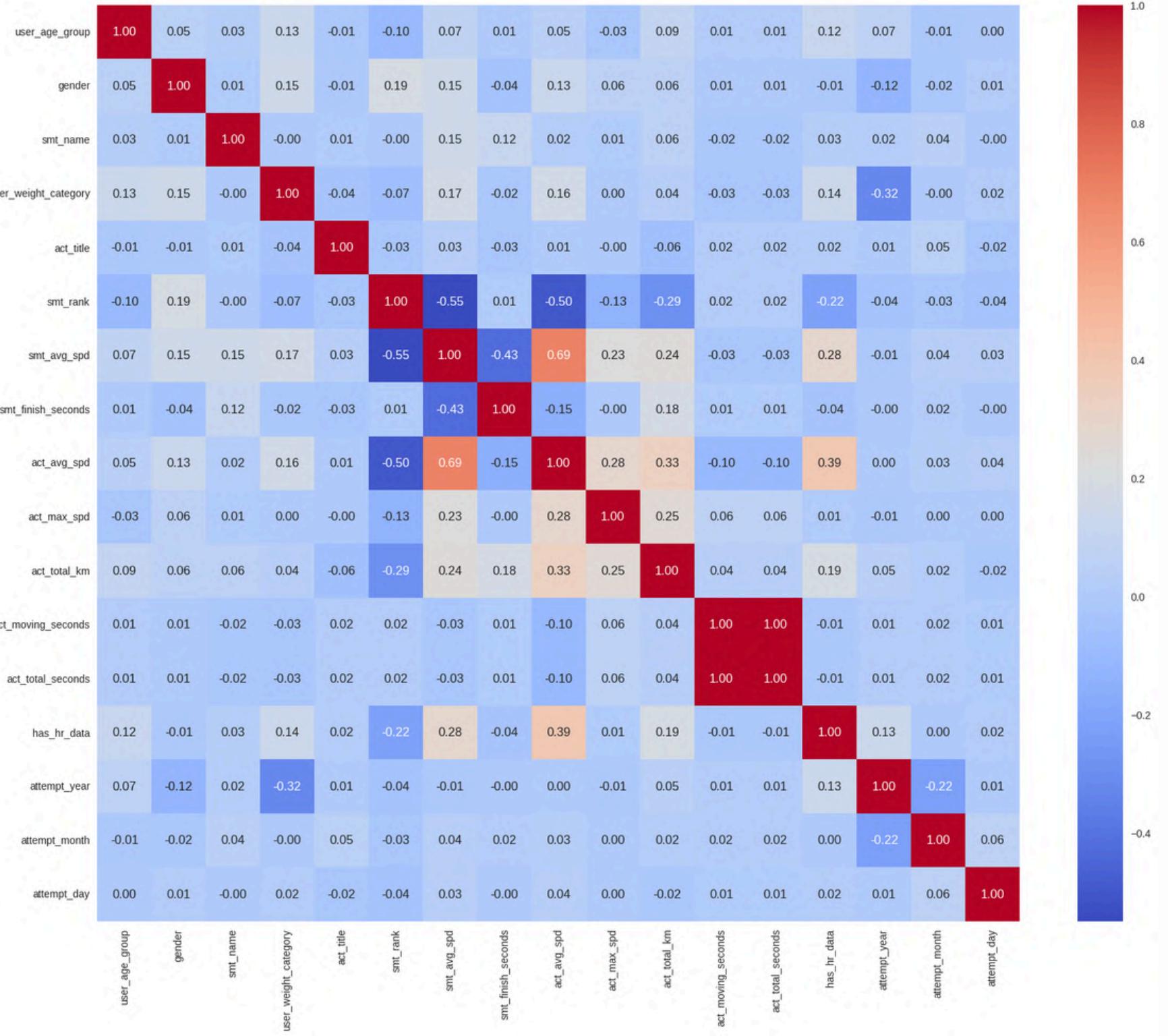
# Clustering



# K-means with Elbow Method

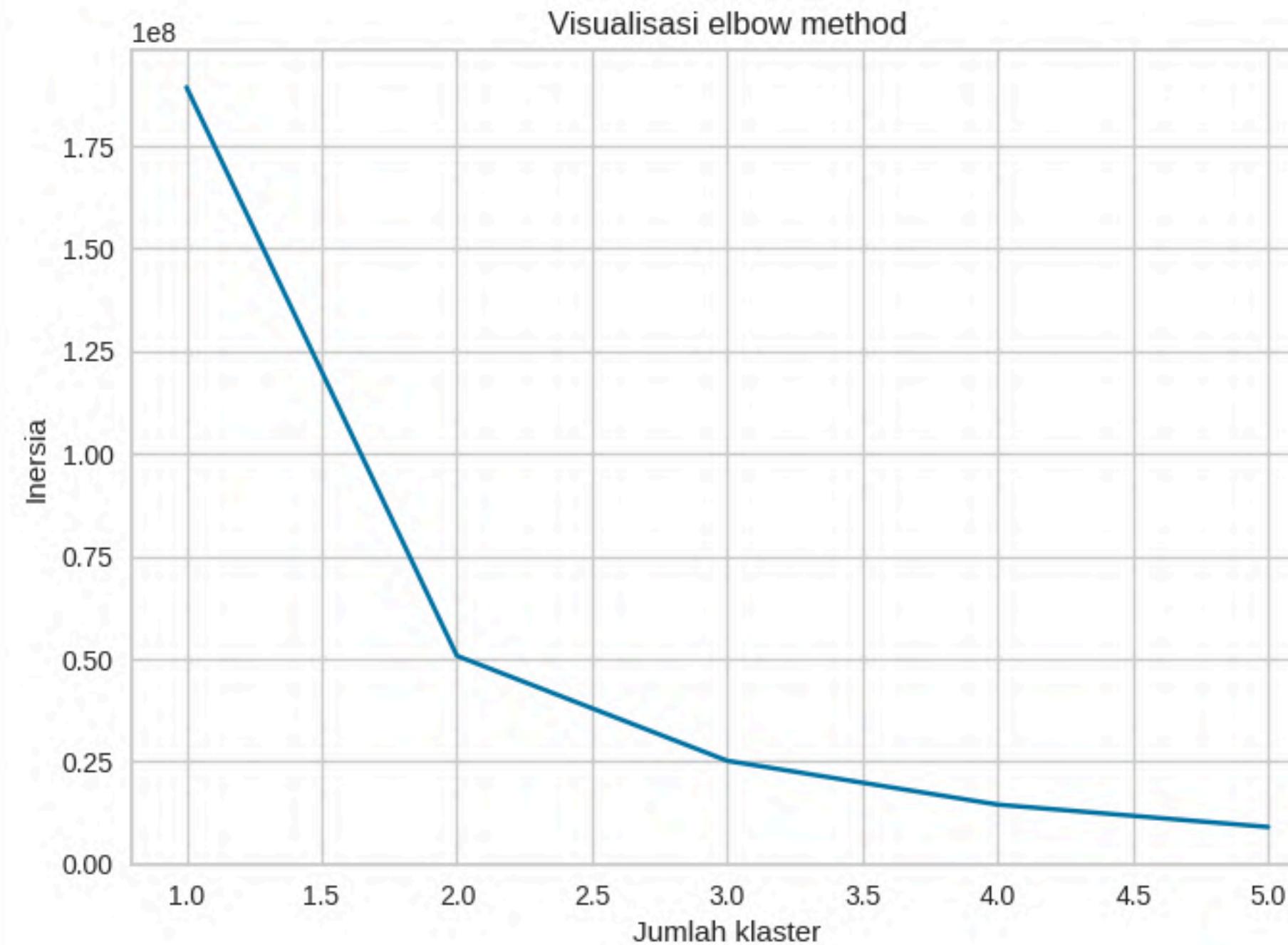


# Feature Selection



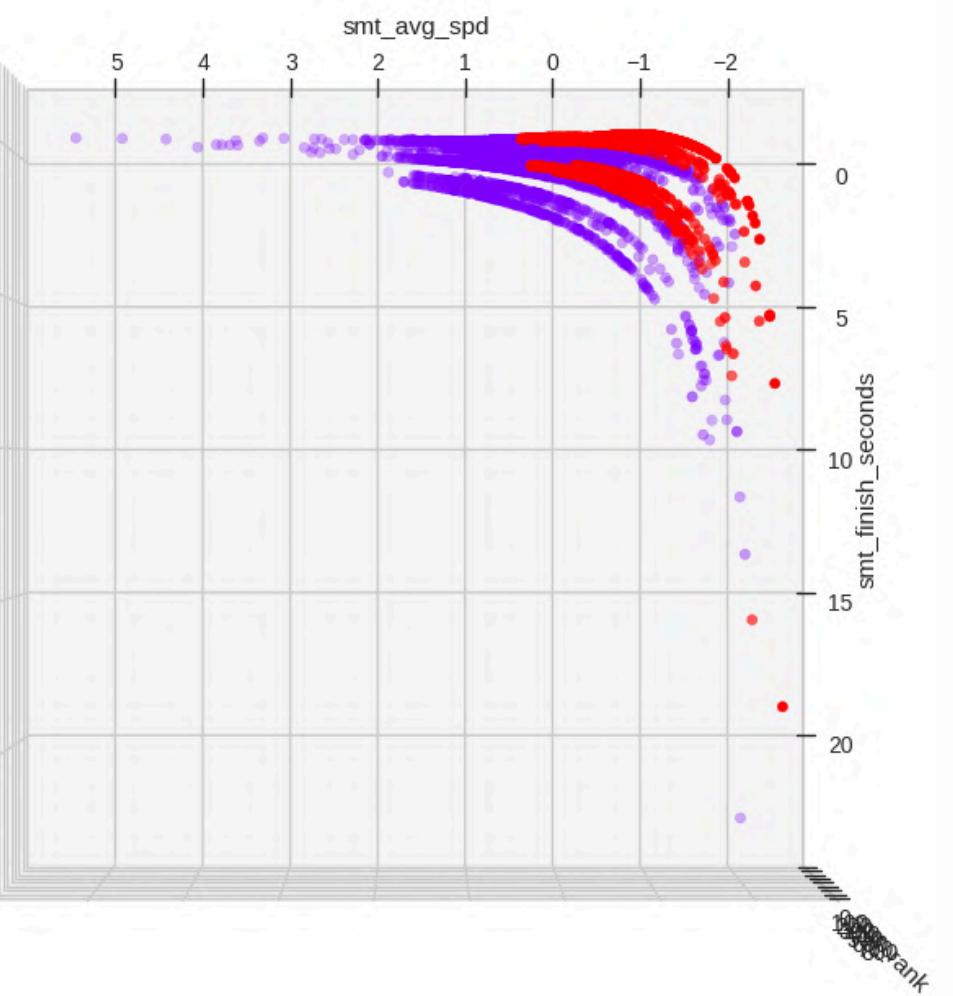
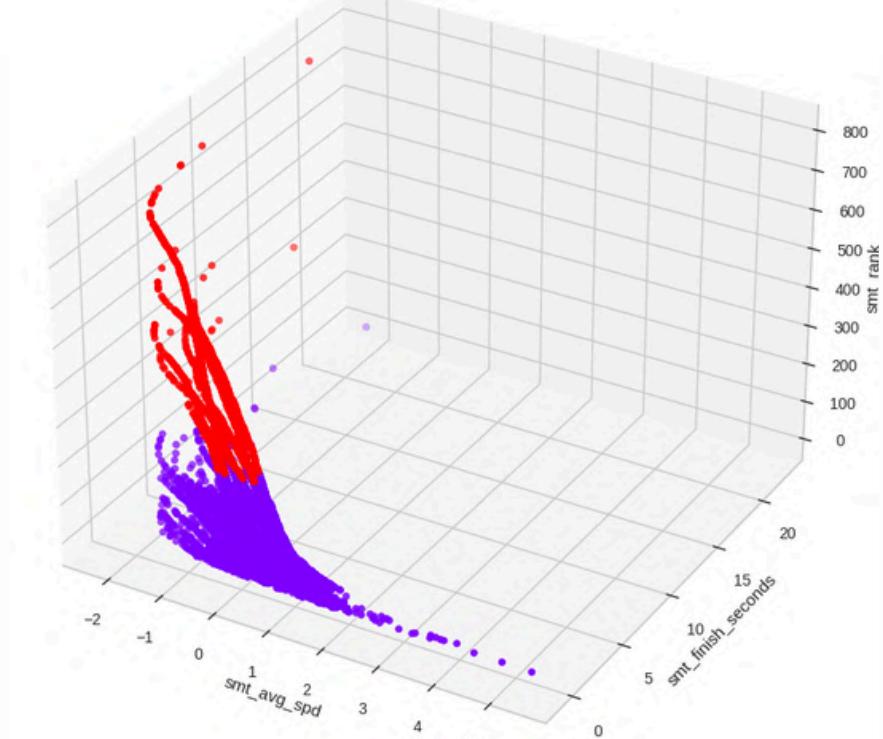
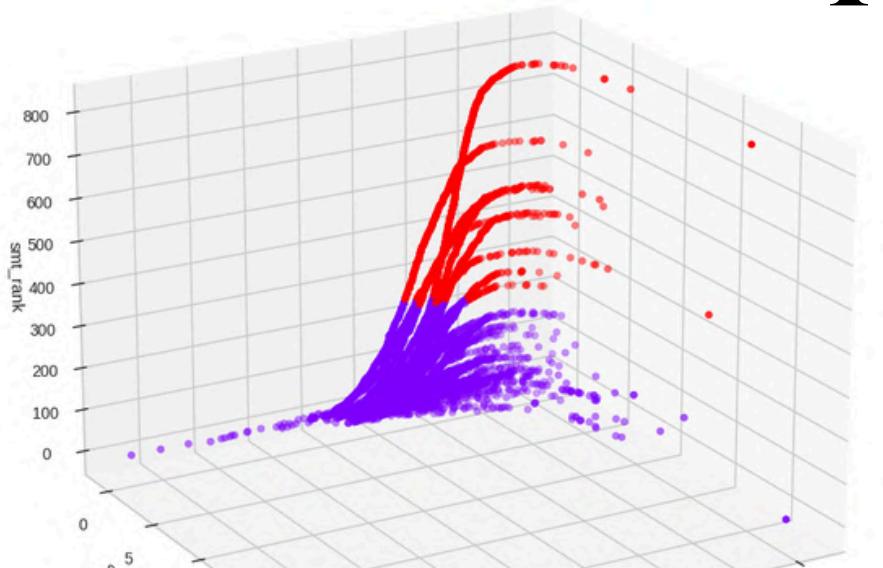
- **smt\_rank** dan **smt\_avg\_spd**: -0.55 (artinya semakin tinggi kecepatan rata-rata segmen, semakin kecil atau lebih baik peringkatnya)
- **smt\_avg\_spd** dan **smt\_finish\_seconds**: -0.43 (semakin tinggi kecepatan rata-rata segmen, semakin singkat waktu penyelesaian segmen)

# Elbow Method



- **k\_optimal = 2** karena dari grafik di mana pada saat  $k = 2$  terdapat titik "elbow" atau titik di mana penurunan nilai inersia mulai melambat.
- Sebelum  $k = 2$ , penurunan inersia lebih tajam yang berarti penambahan jumlah klaster memberikan peningkatan yang besar dalam menjelaskan variasi data.

# Hasil Analisis & Interpretasi



Semakin cepat waktu penyelesaian ('smt\_finish\_seconds') dan semakin tinggi kecepatan rata-rata ('smt\_avg\_spd'), maka semakin kecil nilai rank ('smt\_rank') yang mengindikasikan bahwa performa lebih baik.

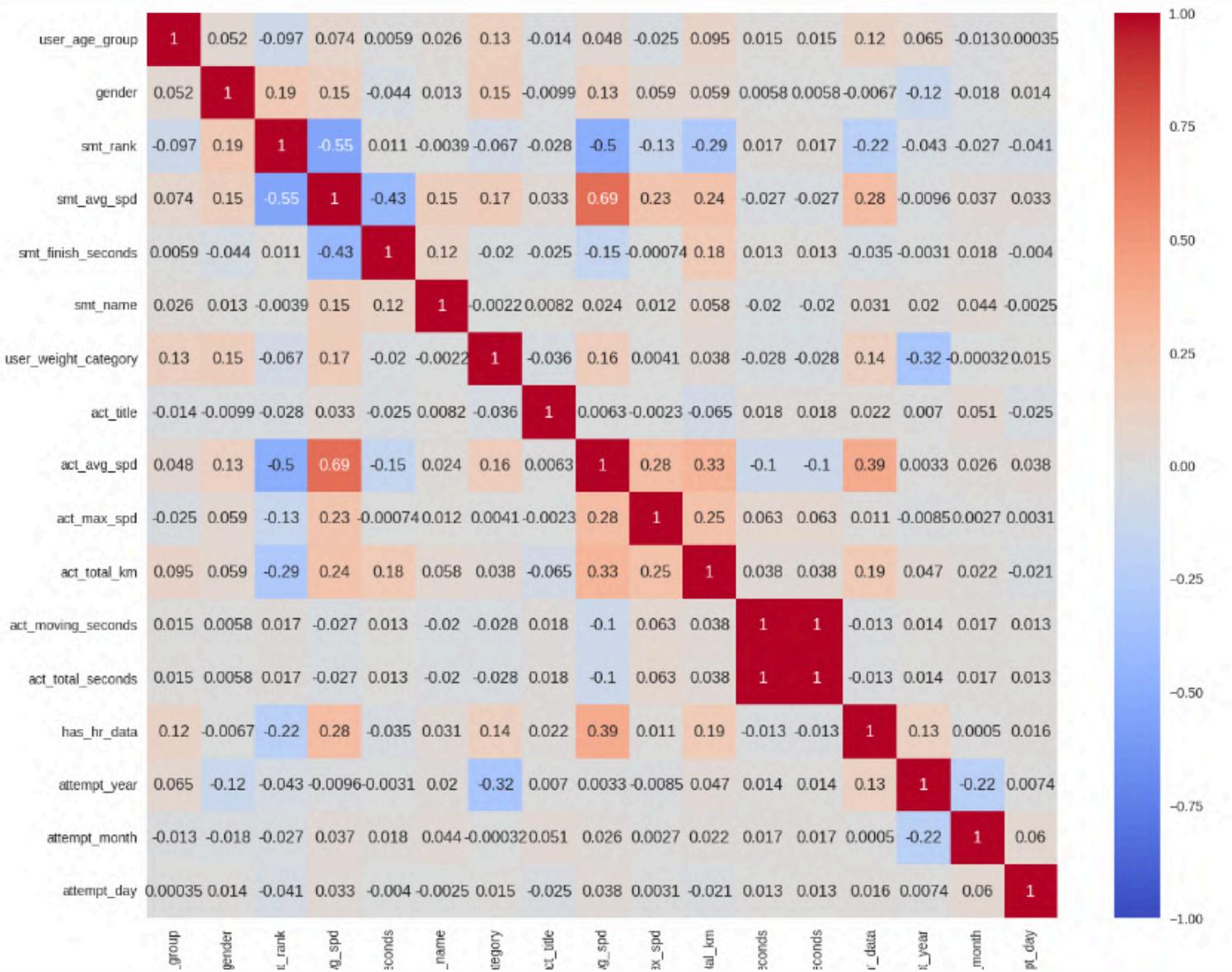


- **Klaster ungu** lebih dominan pada sebagian besar rentang data, apalagi pada nilai yang kecepatan lebih tinggi dan waktu penyelesaian yang lebih singkat.
- **Klaster merah** terdistribusi lebih jauh pada waktu penyelesaian yang lebih besar. Hal ini menandakan bahwa klaster ini mencakup peserta yang kinerjanya rendah, seperti waktu penyelesaian lebih lambat.

# K-Means with Silhouette Method

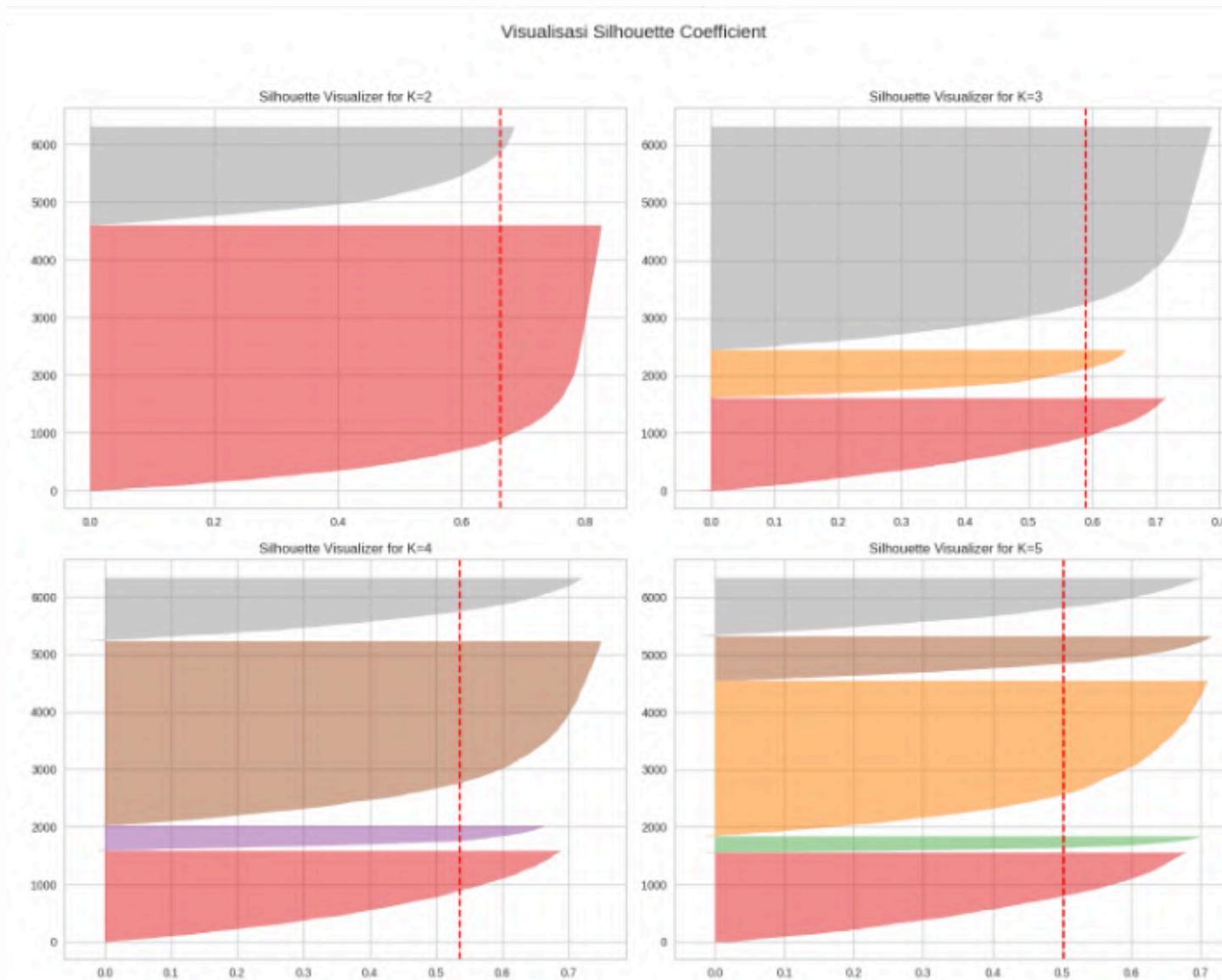


# Feature Selection



Pemilihan fitur user\_weight\_category, smt\_rank, dan act\_total\_km untuk clustering didasarkan pada korelasi rendah antar fitur agar menghindari redundansi informasi. Fitur-fitur ini mewakili aspek berbeda: user\_weight\_category untuk kondisi fisik, smt\_rank untuk performa, dan act\_total\_km untuk kapasitas bersepeda. Ketiganya relevan untuk segmentasi peserta berdasarkan kondisi fisik, kemampuan, dan kapasitas bersepeda, serta menghindari penggunaan fitur dengan korelasi tinggi yang dapat menyebabkan bias dalam clustering.

# Feature Scalling



Untuk k = 2, rata-rata silhouette\_coefficient adalah: 0.6625869490953031

Untuk k = 3, rata-rata silhouette\_coefficient adalah: 0.5911537020228168

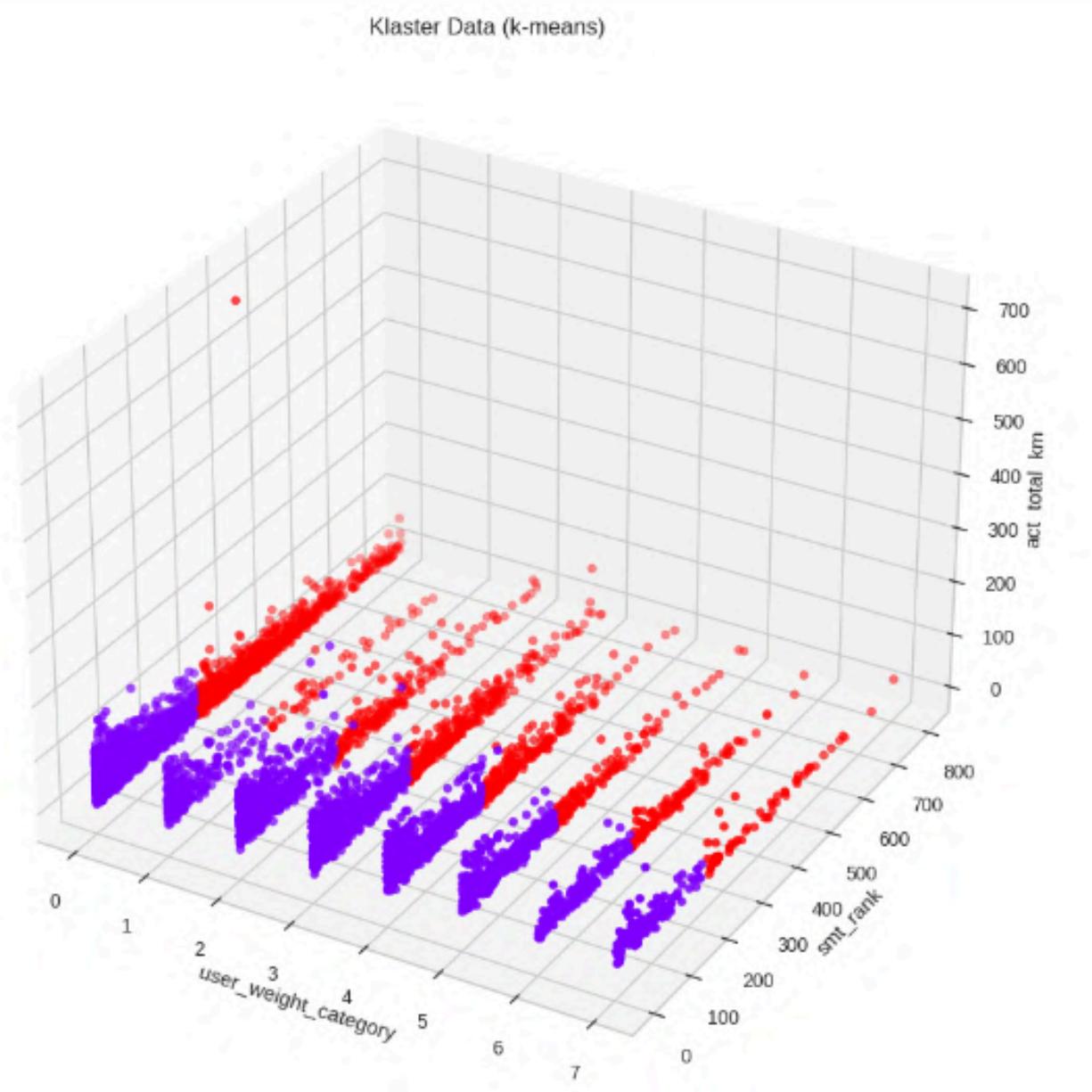
Untuk k = 4, rata-rata silhouette\_coefficient adalah: 0.5434602551365164

Untuk k = 5, rata-rata silhouette\_coefficient adalah: 0.5023615181130121



Clustering terbaik diperoleh pada k=2 dengan Silhouette Coefficient 0.6626, menunjukkan separasi cluster yang jelas. Peningkatan jumlah cluster (k=3 hingga k=5) menurunkan kualitas separasi. Selain itu, clustering dilakukan tanpa standarisasi data karena metode scaling justru menurunkan Silhouette Coefficient dan merusak separasi cluster.

# Hasil Analisis & Interpretasi

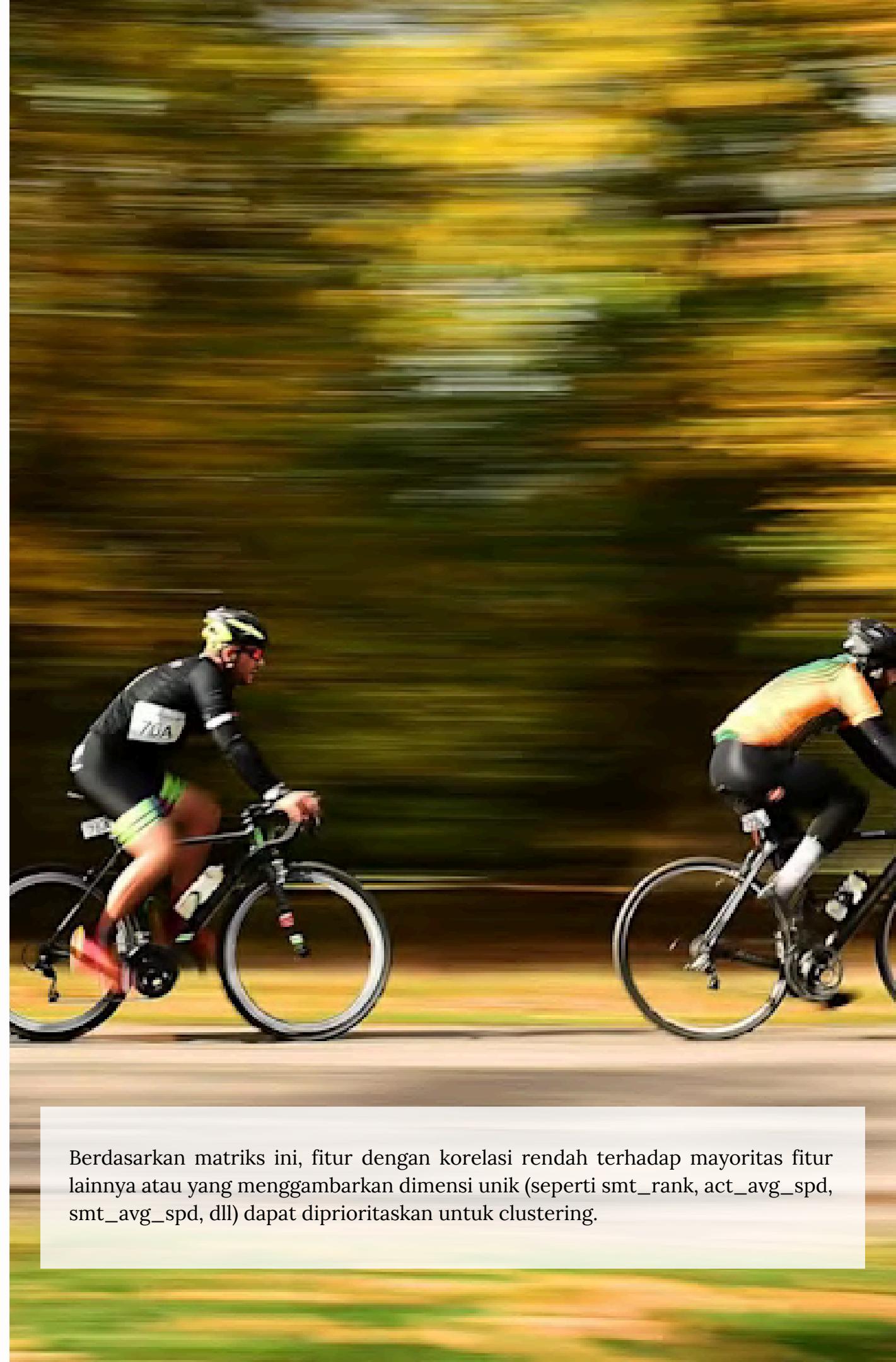
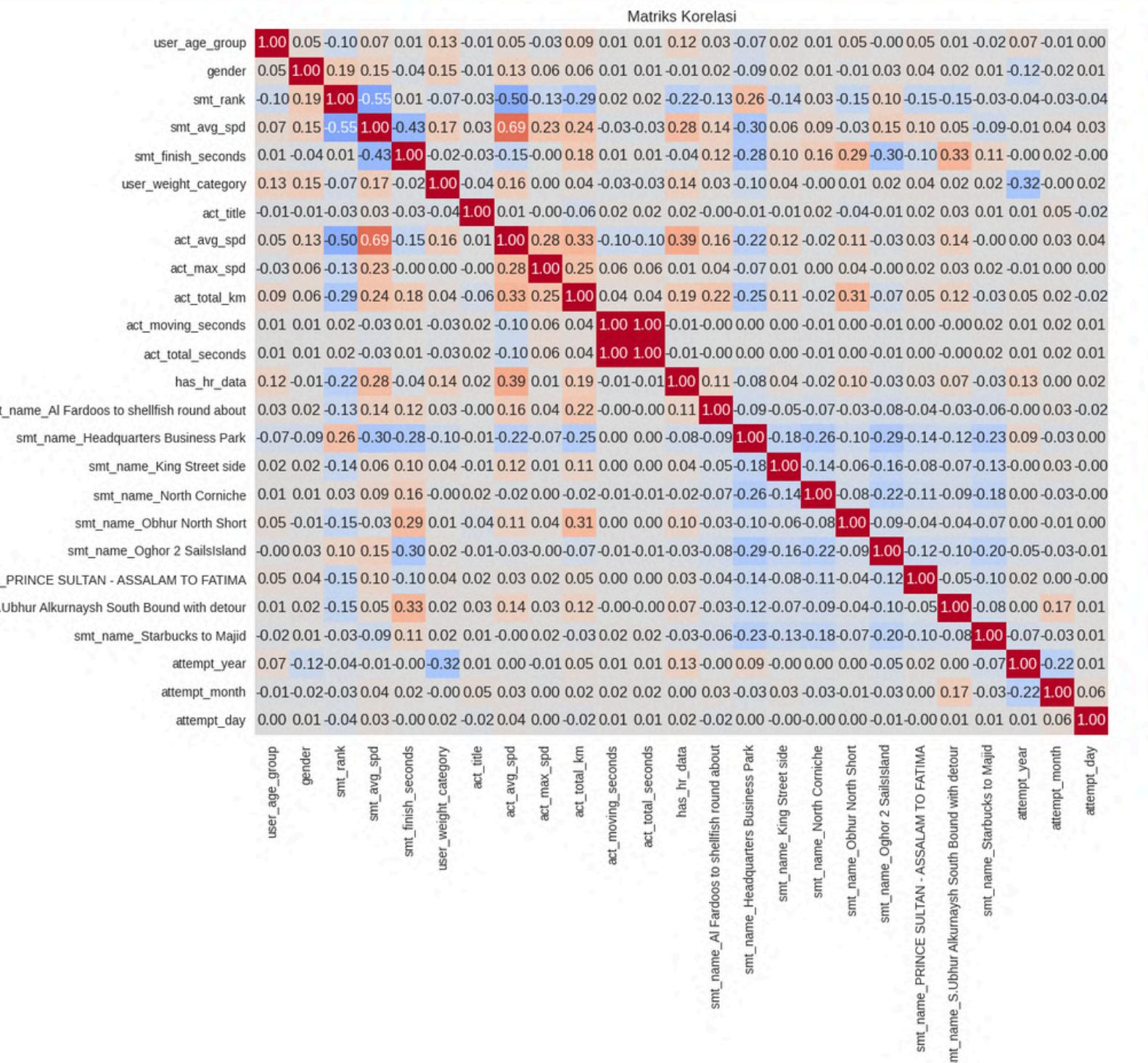


Visualisasi clustering dengan K-Means menunjukkan bahwa smt\_rank memisahkan cluster merah dengan smt\_rank rendah (peringkat lebih baik) dan cluster ungu dengan smt\_rank tinggi (peringkat lebih buruk). Variabel lain seperti user\_weight\_category dan act\_total\_km tidak memengaruhi pembagian cluster karena clusterisasi ini mencerminkan distribusi performa berdasarkan smt\_rank.

# Hierarchical Clustering (ward)

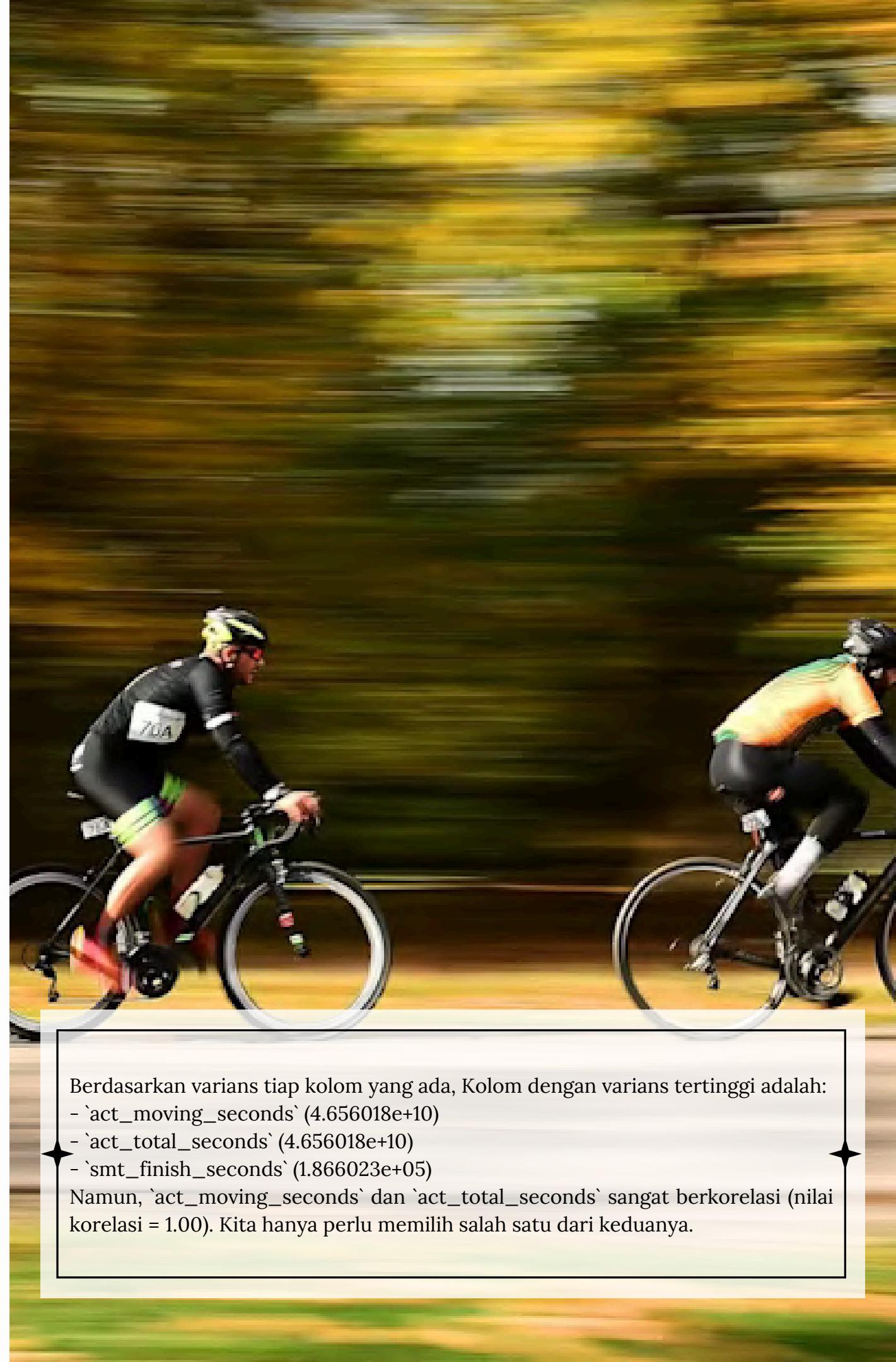


# Feature Selection



# Varians

```
act_moving_seconds           4.656018e+10
act_total_seconds             4.656018e+10
smt_finish_seconds            1.866023e+05
smt_rank                      3.009798e+04
act_total_km                  7.295827e+02
act_max_spd                   2.908846e+02
attempt_day                    7.525260e+01
smt_avg_spd                   7.136972e+01
act_avg_spd                   3.424061e+01
attempt_month                  1.501286e+01
user_weight_category           4.137421e+00
attempt_year                   2.425878e+00
user_age_group                 1.176589e+00
smt_name_Headquarters Business Park 1.852846e-01
smt_name_Oghor 2 SailsIsland      1.601780e-01
has_hr_data                     1.578764e-01
smt_name_North Corniche          1.396867e-01
smt_name_Starbucks to Majid        1.185953e-01
smt_name_King Street side          8.524878e-02
smt_name_PRINCE SULTAN - ASSALAM TO FATIMA 5.409121e-02
smt_name_S.Ubhur Alkurnaysh South Bound with detour 3.889134e-02
gender                          3.859910e-02
smt_name_Obhur North Short         3.092974e-02
smt_name_Al Fardoos to shellfish round about 2.433357e-02
act_title                        1.251300e-02
dtype: float64
```

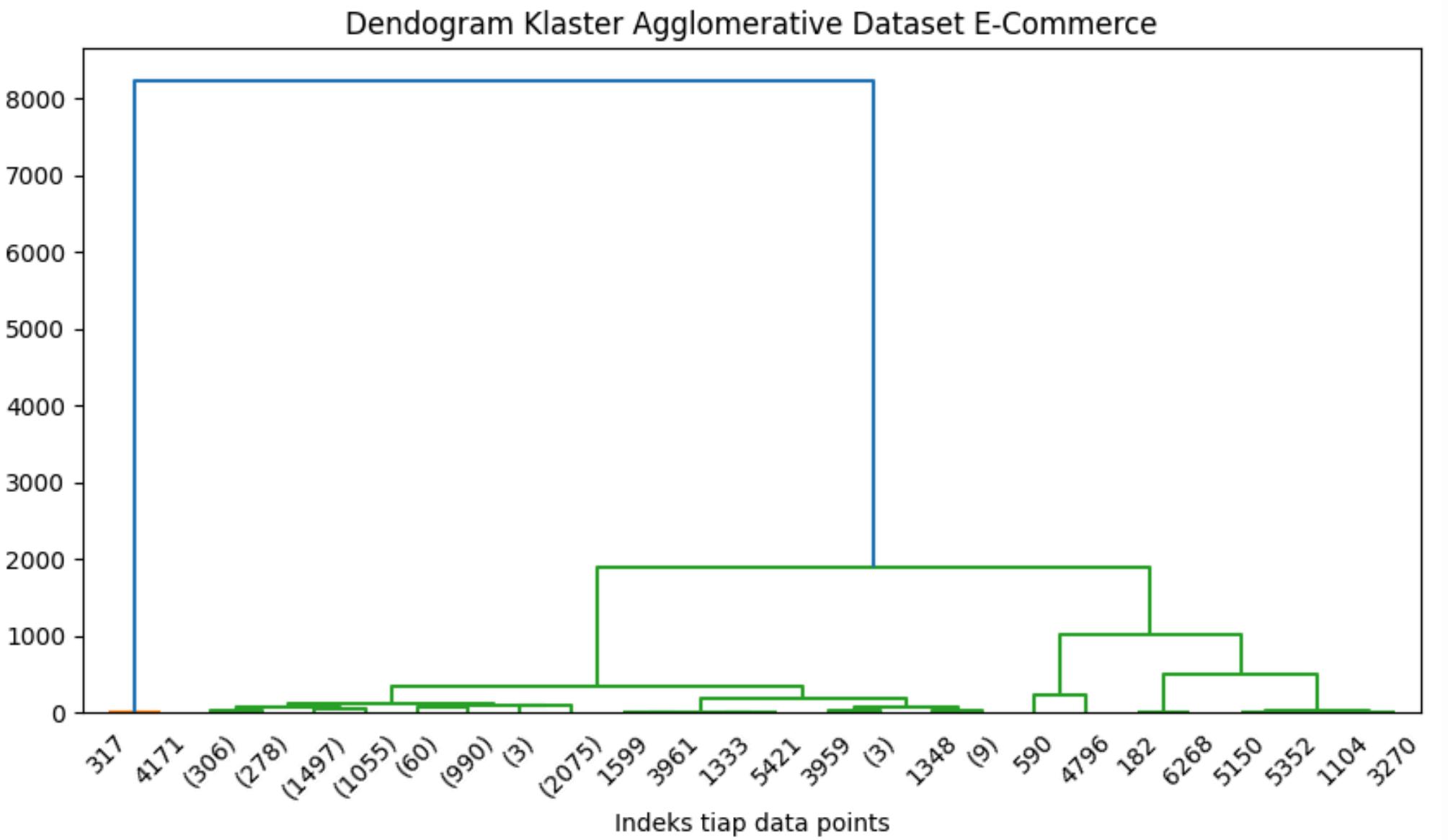


Berdasarkan varians tiap kolom yang ada, Kolom dengan varians tertinggi adalah:

- `act\_moving\_seconds` (4.656018e+10)
- `act\_total\_seconds` (4.656018e+10)
- `smt\_finish\_seconds` (1.866023e+05)

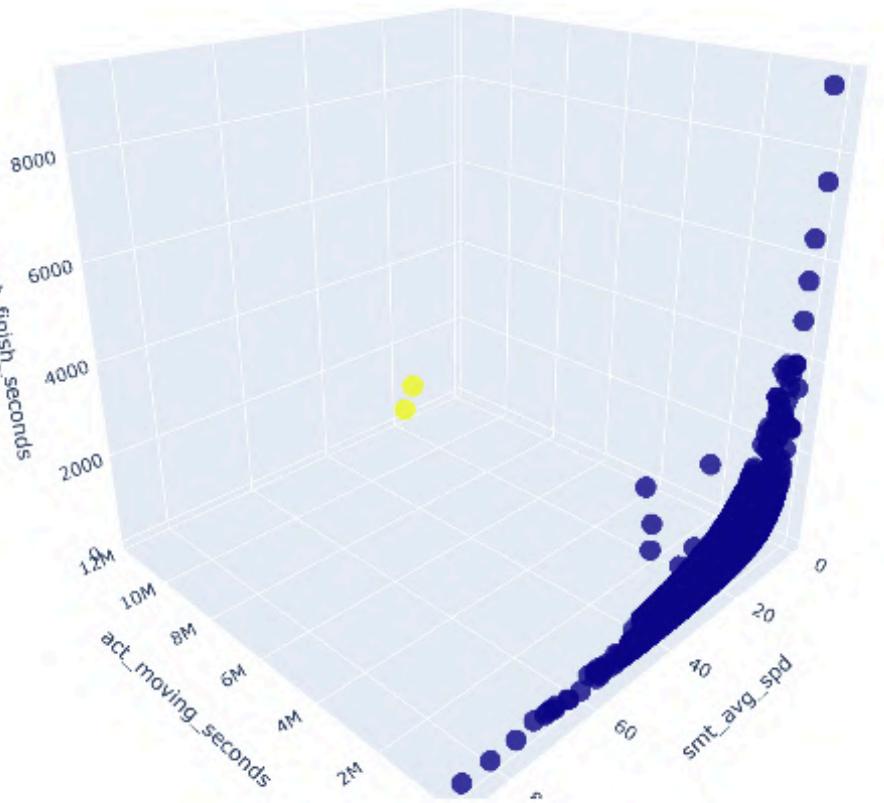
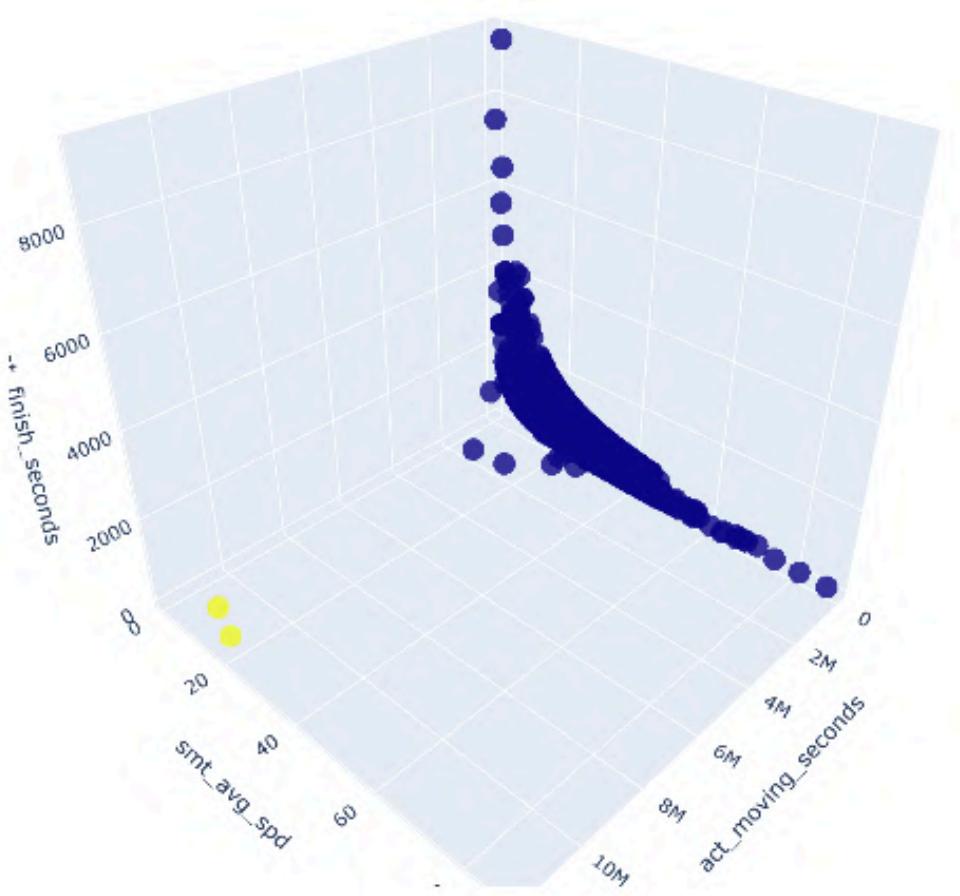
Namun, `act\_moving\_seconds` dan `act\_total\_seconds` sangat berkorelasi (nilai korelasi = 1.00). Kita hanya perlu memilih salah satu dari keduanya.

# Visualisasi Dendogram



Dendrogram menunjukkan bahwa dataset CSL dapat dikelompokkan menjadi **dua klaster besar** berdasarkan jarak dissimilarity yang sangat tinggi. Namun, terdapat beberapa sub-klaster yang lebih kecil dengan jarak dissimilarity rendah, yang menunjukkan adanya kemiripan antar data dalam kelompok tersebut. Secara keseluruhan,

# Hasil Analisis & Interpretasi

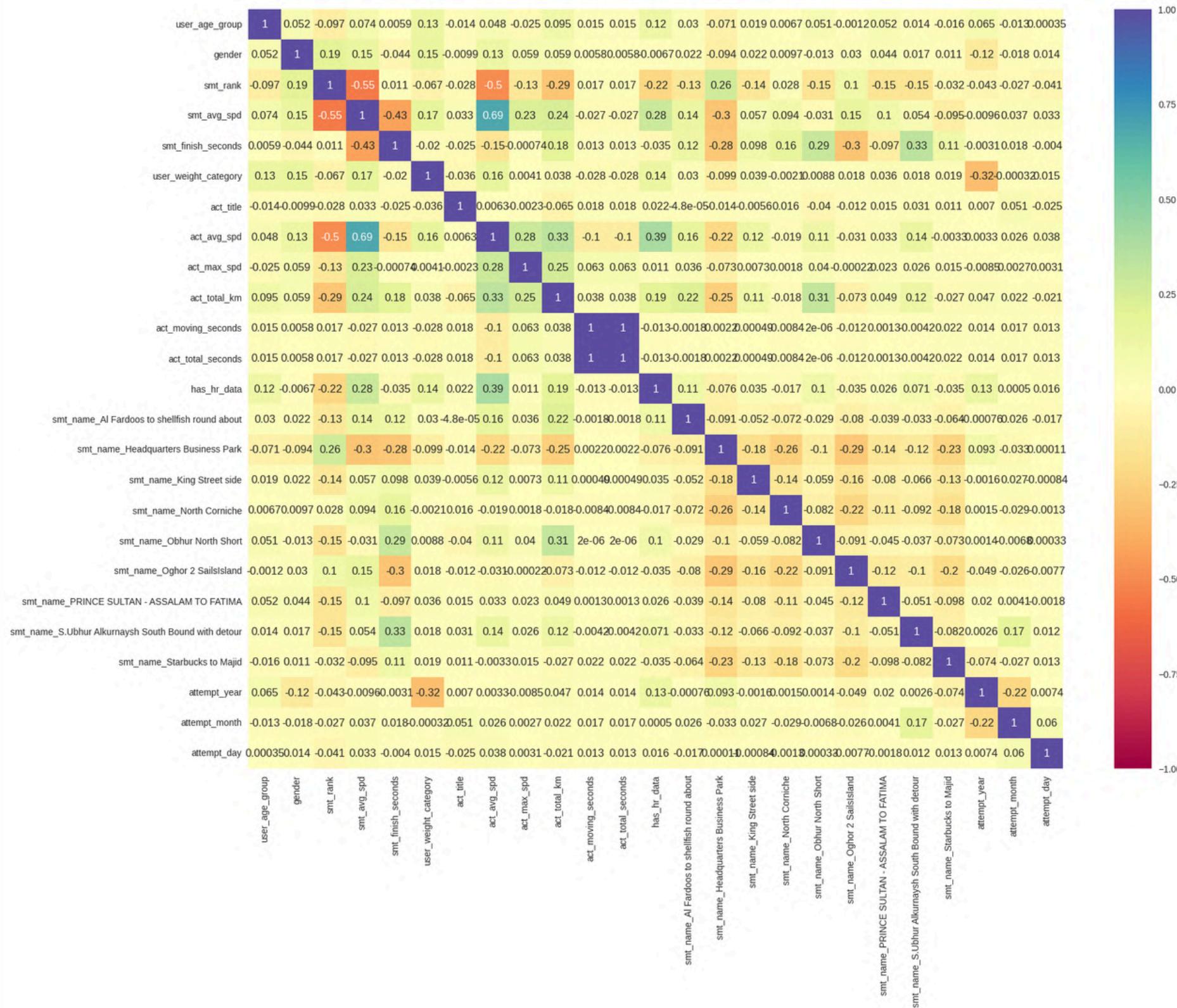


Pada clustering dengan Agglomerative Clustering, terdapat dua klaster utama: biru dan kuning. Klaster biru mendominasi dengan data yang terpusat pada act\_moving\_seconds dan smt\_finish\_seconds, serta memiliki kecepatan rata-rata yang lebih rendah. Sebaliknya, klaster kuning mewakili kelompok kecil dengan smt\_avg\_spd yang jauh lebih tinggi, menunjukkan outliers atau kelompok dengan performa di atas rata-rata.

# Hierarchical Clustering (Average Linkage)



# Feature Selection

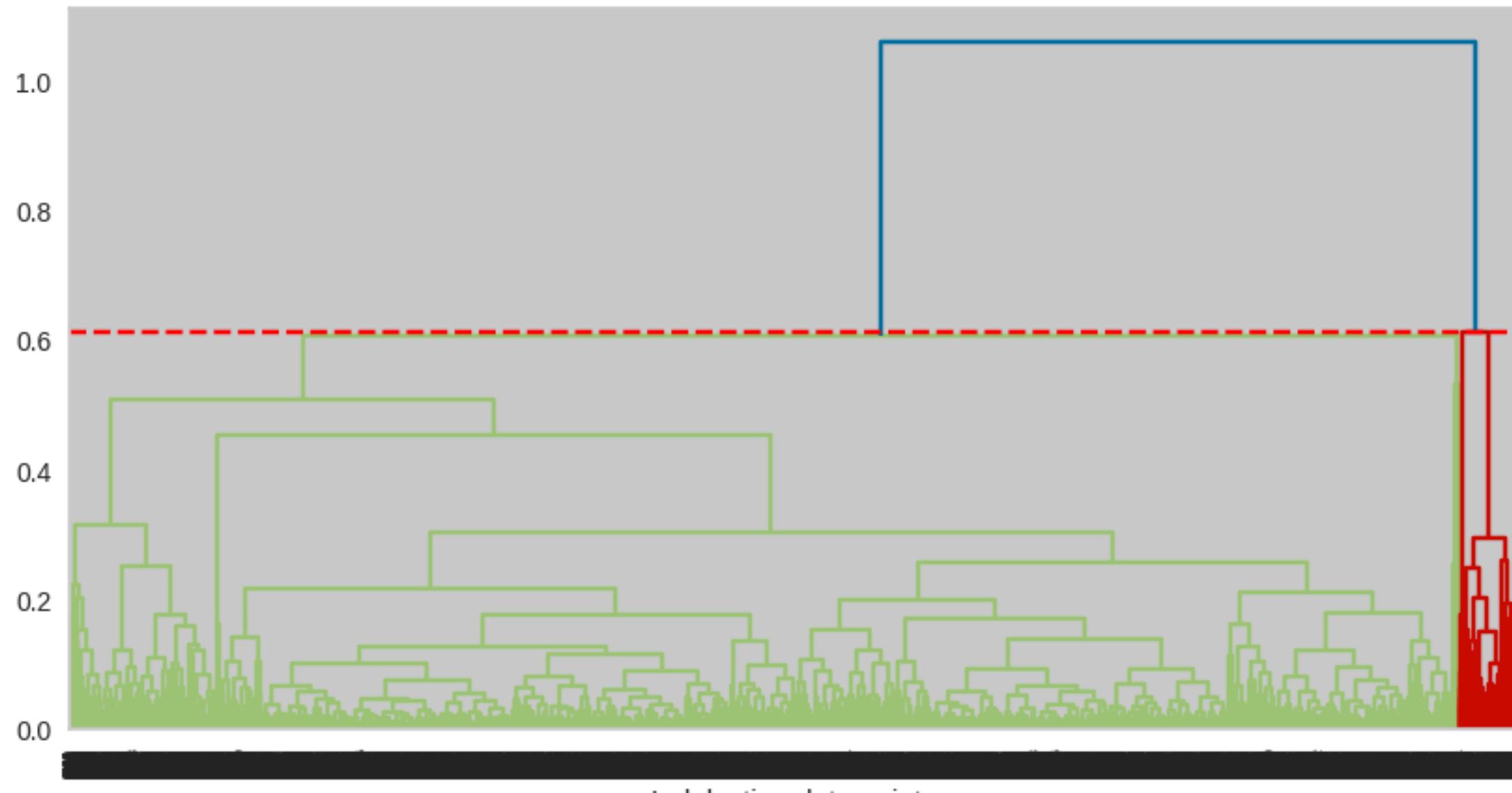


Saya memilih fitur **act\_total\_km**, **gender**, dan **act\_avg\_spd** untuk clustering karena beberapa hal berikut :

- Ketiga fitur ini memiliki korelasi yang tidak terlalu kuat antar satu sama lain, sehingga masing-masing memberikan informasi yang unik dan menghindari redundansi.
- act\_total\_km mencerminkan kapasitas bersepeda, gender menggambarkan demografi peserta, dan act\_avg\_spd menunjukkan performa kecepatan rata-rata.
- Fitur-fitur ini relevan untuk mengelompokkan pesepeda berdasarkan jarak tempuh, jenis kelamin, dan kecepatan, sehingga segmentasi menjadi lebih akurat.
- Dengan memilih fitur yang tidak berkorelasi tinggi, bisa menghindari potensi bias dalam hasil clustering yang dapat disebabkan oleh fitur-fitur yang saling terkait.

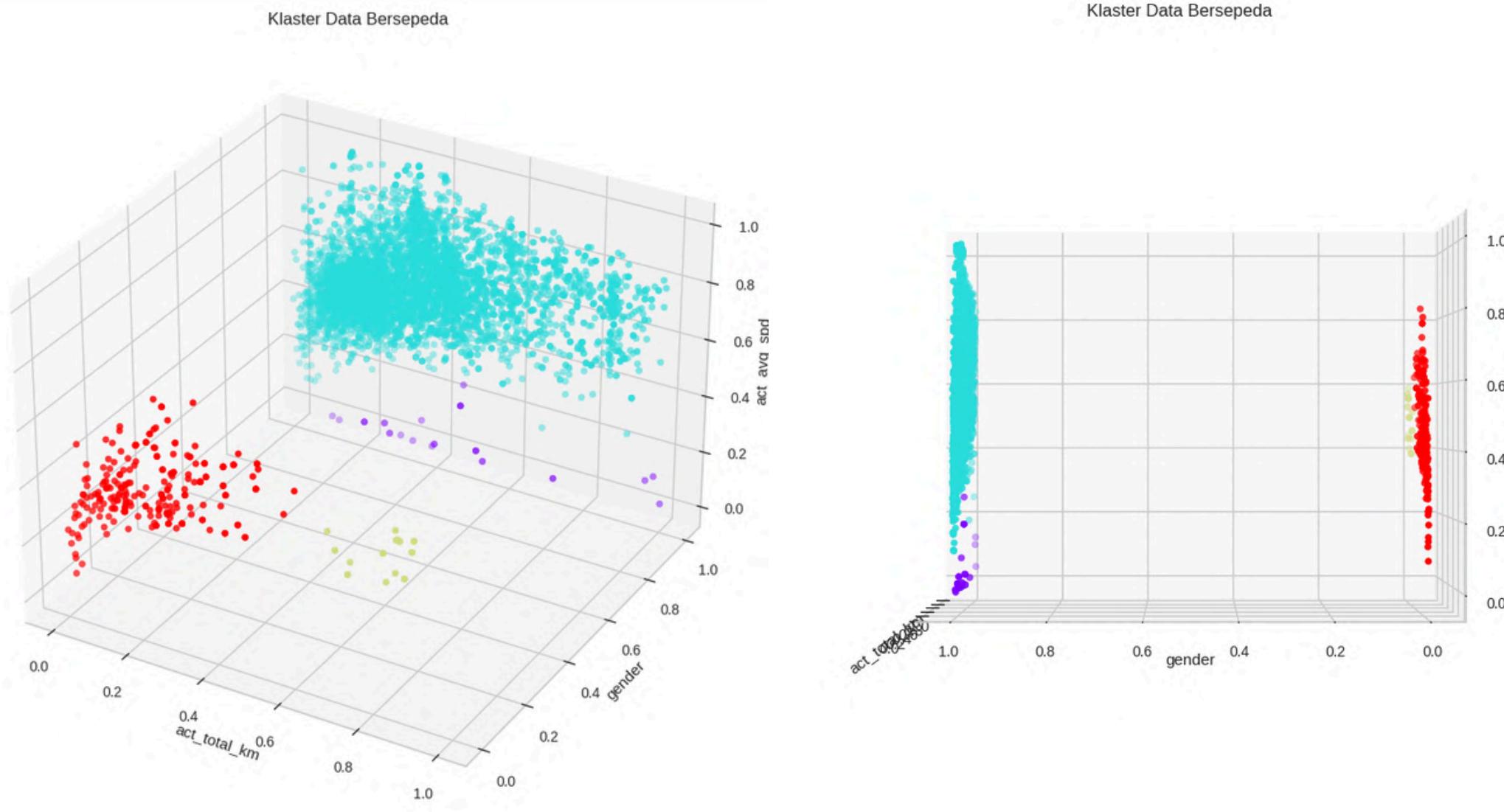
# Dendogram Agglomerative Cluster

Dendogram Klaster Agglomerative Dataset Cycling Segments Leaderboard



Saya menentukan Threshold pada dendrogram dengan mencari jarak penggabungan terbesar antara klaster yang berurutan. Dengan mengurutkan jarak tersebut dan menemukan perbedaan terbesar di antaranya, titik tersebut dijadikan sebagai threshold yang ditandai pada dendrogram. Garis threshold ini membantu menentukan jumlah klaster optimal dengan memisahkan klaster yang memiliki jarak penggabungan signifikan, sehingga menghasilkan segmentasi yang lebih jelas dalam dataset.

# Hasil Analisis & Interpretasi



Hasil clustering mengungkapkan pola performa bersepeda yang berbeda berdasarkan gender, kecepatan rata-rata, dan total jarak tempuh.

**Klaster biru muda** terdiri dari laki-laki dengan kecepatan tinggi dan jarak tempuh konsisten, sementara **klaster ungu** juga laki-laki tetapi dengan kecepatan lebih rendah namun tetap konsisten. Hal ini menunjukkan adanya variasi fokus dalam kelompok laki-laki antara kecepatan dan kestabilan.

Sementara itu, klaster merah dan kuning menggambarkan performa perempuan. **Klaster merah** mencakup perempuan dengan jarak tempuh konsisten dan kecepatan moderat, sedangkan **klaster kuning** menunjukkan perempuan dengan jarak tempuh lebih jauh dan kecepatan serupa. Pola ini menunjukkan bahwa laki-laki cenderung lebih fokus pada kecepatan, sementara perempuan menekankan konsistensi jarak tempuh.



Thank  
You

