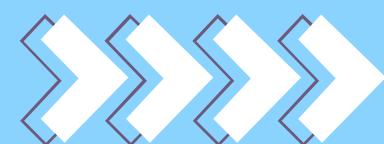




# UNSUPERVISED LEARNING



# ANGGOTA TEAM



- M RIZQI FADHILAH
- M ARVIN FADRIANSYAH
- MELLIZA NASTASIA IZAZI
- THUFAEL BINTANG ALFATTAH
- ZULFIKAR FAUZI
- ANNISA SULISTYANINGSIH
- NIKEN MUSTIKAWENI
- GALIH REFA



# DAFTAR PEMBAHASAN

## 1. EDA

EDA PADA DATASET UNTUK MENDAPATKAN PEMAHAMAN UMUM MENGENAI DATA DAN MEMANDU PROSES FEATURE ENGINEERING



## 2. FEATURE ENGINEERING

PILIH FITUR-FITUR YANG MASUK AKAL SECARA BISNIS UNTUK DIGUNAKAN SEBAGAI FITUR CLUSTERING

## 3. MODELING + EVALUASI

TEMUKAN JUMLAH CLUSTER YANG OPTIMAL DAN EVALUASI CLUSTER YANG DIHASILKAN DENGAN VISUALISASI DAN SILHOUETTE SCORE

## 4. INTERPRETASI MODEL + REKOMENDASI

INTERPRETASI CLUSTER YANG DIHASILKAN SECARA BISNIS DAN BERIKAN REKOMENDASI YANG SESUAI



# DATA DICTIONARY

## D.1. Data Dictionary

- member\_no : Nomor unik yang diberikan kepada setiap anggota program frequent flyer.
- ffp\_date : Tanggal ketika anggota pertama kali bergabung dengan program frequent flyer.
- first\_flight\_date : Tanggal penerbangan pertama yang dilakukan anggota setelah bergabung dengan program frequent flyer.
- gender : Jenis kelamin anggota.
- ffp\_tier : Tingkat atau status keanggotaan dalam program frequent flyer.
- work\_city : Kota tempat anggota bekerja.
- work\_province : Provinsi tempat anggota bekerja.
- work\_country : Negara tempat anggota bekerja.
- age : Usia anggota pada saat data dikumpulkan.
- load\_time : Waktu ketika data dimuat atau diperbarui dalam sistem.
- flight\_count : Jumlah total penerbangan yang dilakukan oleh anggota.
- bp\_sum : Total jumlah poin atau bonus yang diperoleh anggota dari penerbangan atau aktivitas lain.
- sum\_yr\_1 : Jumlah total pengeluaran atau poin yang diperoleh pada tahun pertama periode analisis.
- sum\_yr\_2 : Jumlah total pengeluaran atau poin yang diperoleh pada tahun kedua periode analisis.
- seg\_km\_sum : Total jarak yang ditempuh oleh anggota dalam kilometer dari semua penerbangan yang dilakukan.
- last\_flight\_date : Tanggal penerbangan terakhir yang dilakukan oleh anggota.
- last\_to\_end : Jumlah hari antara penerbangan terakhir yang dilakukan anggota dan akhir periode analisis data.
- avg\_interval : Rata-rata interval waktu antara penerbangan-penerbangan yang dilakukan oleh anggota.
- max\_interval : Interval waktu maksimum antara penerbangan-penerbangan yang dilakukan anggota.
- exchange\_count : Jumlah penukaran poin atau hadiah yang dilakukan oleh anggota.
- avg\_discount : Rata-rata diskon yang diterima anggota dari program loyalitas atau promosi khusus.
- points\_sum : Total jumlah poin yang diperoleh anggota dari seluruh penerbangan dan aktivitas.
- point\_notflight : Jumlah poin yang diperoleh anggota dari aktivitas selain penerbangan, seperti penukaran poin atau promosi.

**Tidak semua variabel dalam data dictionary harus dipakai secara aktif dalam aplikasi atau sistem. Data dictionary sering kali mencakup informasi tentang semua variabel atau elemen data yang ada dalam sistem, bahkan jika beberapa dari variabel tersebut tidak digunakan secara langsung.**



# DATA TYPE

## D.2. Tipe Data

```
▶ # Informasi umum pada data  
data_flight.info()  
  
→ <class 'pandas.core.frame.DataFrame'>  
RangeIndex: 62988 entries, 0 to 62987  
Data columns (total 23 columns):  
 #   Column           Non-Null Count  Dtype     
---  --  
 0   member_no        62988 non-null   int64    
 1   ffp_date         62988 non-null   object    
 2   first_flight_date 62988 non-null   object    
 3   gender           62985 non-null   object    
 4   ffp_tier          62988 non-null   int64    
 5   work_city         60719 non-null   object    
 6   work_province    59740 non-null   object    
 7   work_country     62962 non-null   object    
 8   age               62568 non-null   float64  
 9   load_time         62988 non-null   object    
 10  flight_count      62988 non-null   int64    
 11  bp_sum            62988 non-null   int64    
 12  sum_yr_1          62437 non-null   float64  
 13  sum_yr_2          62850 non-null   float64  
 14  seg_km_sum        62988 non-null   int64    
 15  last_flight_date 62988 non-null   object    
 16  last_to_end       62988 non-null   int64    
 17  avg_interval      62988 non-null   float64  
 18  max_interval      62988 non-null   int64    
 19  exchange_count    62988 non-null   int64    
 20  avg_discount      62988 non-null   float64  
 21  points_sum        62988 non-null   int64    
 22  point_notflight   62988 non-null   int64    
dtypes: float64(5), int64(10), object(8)  
memory usage: 11.1+ MB
```



Pada dataset kali ini terdapat 3 tipe data yaitu integer, object, dan float

fitur-fitur yang memiliki tipe data :

1. Integer = member\_no, ffp\_tier, flight\_count, bp\_sum, seg\_km\_sum, last\_to\_end, max\_interval, exchange\_count, points\_sum, dan point\_notflight
2. Object = ffp\_date, first\_flight\_date, gender, work\_city, work\_provine, work\_country, dan last\_flight\_date
3. Float = age, sum\_yr\_1, sum\_yr\_2, avg\_interval, dan avg\_discount





# EXPLORATORY DATA ANALYSIS



# KONVERSI TIPE DATA YANG SESUAI

```
▶ data_flight = data_flight.astype({  
    'member_no' : 'str',  
    'ffp_tier' : 'str'  
})
```

Lakukan konversi tipe data `member\_no` dan `ffp\_tier` dari integer menjadi string

```
[ ] # Ubah data yang tidak valid  
data_flight['last_flight_date'] = data_flight['last_flight_date'].str.replace('2014/2/29 0:00:00', '2/28/2014')
```

Lakukan konversi tipe data tanggal dari tipedata objek menjadi datetime dan ubah menjadi format datetime yang umum digunakan (YYYY-MM-DD)



# KONVERSI TIPE DATA YANG SESUAI

```
▶ # Inisialisasikan kolom tanggal
date_column = ['ffp_date', 'load_time', 'first_flight_date', 'last_flight_date']

# Konversi tipe data
for date_col in date_column :
    data_flight[date_col] = pd.to_datetime(data_flight[date_col], format='%m/%d/%Y')

# Tampilkan data
data_flight.head()
```

PERLU MENGUBAH FORMAT DATE TIME MENJADI FORMAT YANG BENAR YAITU PADA DATA YANG SEHARUSNYA TAHUN KABISAT MAKAN PERLU ADANYA PERGANTIAN TANGGALNYA. FORMAT YANG DIGUNAKAN JUGA PERLU MENGGUNAKAN YYYY/MM/DD



**INFLVATORS**  
INFORMATION INOVATORS



# PERIKSA MISSING VALUE DAN DUPLIKASI DATA

```
Kolom member_no memiliki 0 missing value
Kolom ffp_date memiliki 0 missing value
Kolom first_flight_date memiliki 0 missing value
Kolom gender memiliki 3 missing value
Kolom ffp_tier memiliki 0 missing value
Kolom work_city memiliki 2269 missing value
Kolom work_province memiliki 3248 missing value
Kolom work_country memiliki 26 missing value
Kolom age memiliki 420 missing value
Kolom load_time memiliki 0 missing value
Kolom flight_count memiliki 0 missing value
Kolom bp_sum memiliki 0 missing value
Kolom sum_yr_1 memiliki 551 missing value
Kolom sum_yr_2 memiliki 138 missing value
Kolom seg_km_sum memiliki 0 missing value
Kolom last_flight_date memiliki 0 missing value
Kolom last_to_end memiliki 0 missing value
Kolom avg_interval memiliki 0 missing value
Kolom max_interval memiliki 0 missing value
Kolom exchange_count memiliki 0 missing value
Kolom avg_discount memiliki 0 missing value
Kolom points_sum memiliki 0 missing value
Kolom point_notflight memiliki 0 missing value
```

```
[ ] for column in data_flight.columns:
    total_null = data_flight[column].isna().sum()
    print(f'Kolom {column} memiliki {total_null} missing value')

[ ] # Periksa missing value pada tiap kolom
for column in data_flight.columns:
    total_null = data_flight[column].isna().sum()
    if(total_null > 0):
        print(f'Kolom {column} memiliki {total_null} missing value')
```

- Kolom gender memiliki 3 missing value
- Kolom work\_city memiliki 2269 missing value
- Kolom work\_province memiliki 3248 missing value
- Kolom work\_country memiliki 26 missing value
- Kolom age memiliki 420 missing value
- Kolom sum\_yr\_1 memiliki 551 missing value
- Kolom sum\_yr\_2 memiliki 138 missing value



# PERIKSA MISSING VALUE DAN DUPLIKASI DATA

Missing value terdapat di 7 Kolom dan perlu dilakukan penangan dengan memberikan nilai atau mengisikan nilai Median pada Kolom Age yang none , kemudian mengisi nilai 0 pada sum\_yr 1 dan 2

```
[ ] # Isikan median umur pada kolom Age yang None
median_age = data_flight['age'].median()
data_flight['age'] = data_flight['age'].fillna(median_age)

# Isikan 0 pada kolom sum_yr_1 dan sum_yr_2 yang None
data_flight[['sum_yr_1', 'sum_yr_2']] = data_flight[['sum_yr_1', 'sum_yr_2']].fillna(0)
```

Pemberian data atau nilai median harapannya adalah tidak mengganggu data lainnya karena perlu penanganan dan juga pengisian data nol agar tidak kosong datanya. karena kita tidak tau berapa kali dia terbang ? atau tidak terbang sama sekali.



# PERIKSA MISSING VALUE DAN DUPLIKASI DATA

▶ # Melakukan pemeriksaan pada duplikasi data  
print(f'Terdapat {data\_flight.duplicated().sum()} data yang duplikat')

→ Terdapat 0 data yang duplikat

Sama seperti hasil sebelumnya, maka tiap data akan unique atau bisa dibilang tidak memiliki duplikasi data.



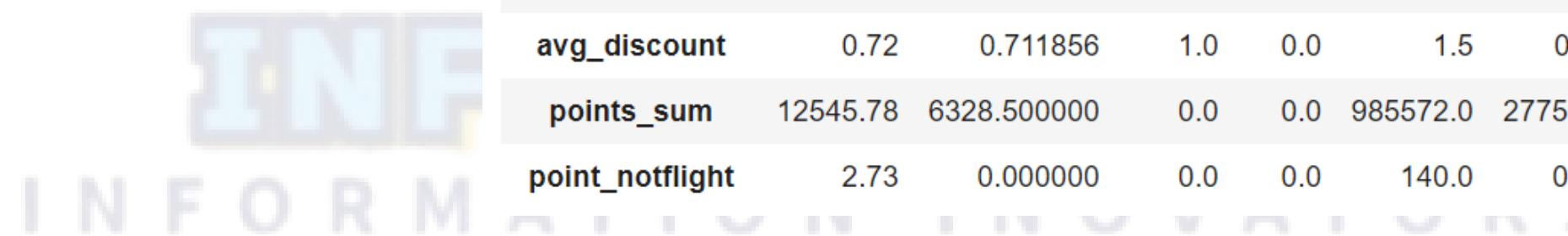
# STATISTIK DESKRIPTIF

```
[ ] number_data = data_flight.select_dtypes(include='number')

statistics = pd.DataFrame({
    'mean': number_data.mean().round(2),
    'median': number_data.median(),
    'modus': number_data.mode().iloc[0],
    'min': number_data.min(),
    'max': number_data.max(),
    'q1': number_data.quantile(0.25),
    'q2': number_data.quantile(0.50),
    'q3': number_data.quantile(0.75),
    'stdev': number_data.std().round(2)
})

display(statistics)
```

	mean	median	modus	min	max	q1	q2	q3	stdev
age	42.47	41.000000	41.0	6.0	110.0	35.000000	41.000000	48.000000	9.85
flight_count	11.84	7.000000	2.0	2.0	213.0	3.000000	7.000000	15.000000	14.05
bp_sum	10925.08	5700.000000	0.0	0.0	505308.0	2518.000000	5700.000000	12831.000000	16339.49
sum_yr_1	5308.53	2754.000000	0.0	0.0	239560.0	960.000000	2754.000000	6524.250000	8089.29
sum_yr_2	5591.75	2761.500000	0.0	0.0	234188.0	768.000000	2761.500000	6826.250000	8697.77
seg_km_sum	17123.88	9994.000000	3934.0	368.0	580717.0	4747.000000	9994.000000	21271.250000	20960.84
last_to_end	176.12	108.000000	1.0	1.0	731.0	29.000000	108.000000	268.000000	183.82
age_member	1487.00	1272.000000	1175.0	367.0	3439.0	734.000000	1272.000000	2181.000000	847.22
avg_interval	67.75	44.666667	2.0	0.0	728.0	23.370370	44.666667	82.000000	77.52
max_interval	166.03	143.000000	2.0	0.0	728.0	79.000000	143.000000	228.000000	123.40
exchange_count	0.32	0.000000	0.0	0.0	46.0	0.000000	0.000000	0.000000	1.14
avg_discount	0.72	0.711856	1.0	0.0	1.5	0.611997	0.711856	0.809476	0.19
points_sum	12545.78	6328.500000	0.0	0.0	985572.0	2775.000000	6328.500000	14302.500000	20507.82
point_notflight	2.73	0.000000	0.0	0.0	140.0	0.000000	0.000000	1.000000	7.36



# STATISTIK DESKRIPTIF

Berdasarkan statistik deskriptif dari gambar, berikut adalah kesimpulan sederhana:

- Usia rata-rata anggota adalah sekitar 42 tahun, dengan variasi yang moderat. Ini menunjukkan bahwa program frequent flyer menarik bagi berbagai rentang usia.
- Sebagian besar anggota melakukan penerbangan kurang dari 12 kali, tetapi ada beberapa anggota yang melakukan penerbangan sangat sering, hingga 213 kali. Ini menunjukkan adanya kelompok anggota yang sangat aktif.
- Poin bonus (bp\_sum) yang diperoleh anggota bervariasi secara signifikan, dengan rata-rata sekitar 10.925 poin. Beberapa anggota memiliki akumulasi poin yang sangat tinggi, menunjukkan loyalitas atau keterlibatan yang kuat.
- Jarak tempuh penerbangan juga bervariasi, dengan beberapa anggota menempuh jarak lebih dari 580.000 km, menunjukkan beberapa anggota adalah frequent flyers jarak jauh.
- Interval antara penerbangan anggota cukup bervariasi, dengan beberapa anggota memiliki jeda waktu panjang antara penerbangan. Rata-rata interval waktu sekitar 68 hari, tetapi bisa mencapai hingga 728 hari.
- Sebagian besar anggota tidak sering menukarkan poin mereka, dengan rata-rata pertukaran yang rendah (0,32 kali).
- Diskon rata-rata yang diterima anggota sekitar 72%, menunjukkan bahwa program ini memberikan nilai yang cukup baik dalam bentuk diskon kepada anggotanya.

Secara keseluruhan, data menunjukkan bahwa ada perbedaan yang signifikan dalam perilaku dan keterlibatan anggota dalam program frequent flyer, dengan beberapa anggota sangat aktif dan yang lain kurang aktif atau jarang melakukan penerbangan.

	mean	median	modus	min	max	q1	q2	q3	stdev
age	42.47	41.000000	41.0	6.0	110.0	35.000000	41.000000	48.000000	9.85
flight_count	11.84	7.000000	2.0	2.0	213.0	3.000000	7.000000	15.000000	14.05
bp_sum	10925.08	5700.000000	0.0	0.0	505308.0	2518.000000	5700.000000	12831.000000	16339.49
sum_yr_1	5308.53	2754.000000	0.0	0.0	239560.0	960.000000	2754.000000	6524.250000	8089.29
sum_yr_2	5591.75	2761.500000	0.0	0.0	234188.0	768.000000	2761.500000	6826.250000	8697.77
seg_km_sum	17123.88	9994.000000	3934.0	368.0	580717.0	4747.000000	9994.000000	21271.250000	20960.84
last_to_end	176.12	108.000000	1.0	1.0	731.0	29.000000	108.000000	268.000000	183.82
age_member	1487.00	1272.000000	1175.0	367.0	3439.0	734.000000	1272.000000	2181.000000	847.22
avg_interval	67.75	44.666667	2.0	0.0	728.0	23.370370	44.666667	82.000000	77.52
max_interval	166.03	143.000000	2.0	0.0	728.0	79.000000	143.000000	228.000000	123.40
exchange_count	0.32	0.000000	0.0	0.0	46.0	0.000000	0.000000	0.000000	1.14
avg_discount	0.72	0.711856	1.0	0.0	1.5	0.611997	0.711856	0.809476	0.19
points_sum	12545.78	6328.500000	0.0	0.0	985572.0	2775.000000	6328.500000	14302.500000	20507.82
point_notflight	2.73	0.000000	0.0	0.0	140.0	0.000000	0.000000	1.000000	7.36



# UNIQUE VALUE KATEGORIKAL

```
# Seleksi hanya kolom bertipe non-numerik
object_columns = ['gender', 'ffp_tier', 'work_city', 'work_province', 'work_country']

for obj in object_columns:
    if (obj not in ['lsoa_code']):
        nilai_unik = data_flight[obj].astype("str").sort_values().unique()
        print(f'Kolom {obj} Terdapat {len(nilai_unik)} nilai unik')
        print(f'Isi : {", ".join(nilai_unik)}\n')
```

→ Kolom **gender**

Terdapat 3 nilai unik  
Isi : *Female, Male, nan*

Kolom **ffp\_tier**

Terdapat 3 nilai unik  
Isi : *4, 5, 6*

Kolom **work\_city**

Terdapat 3235 nilai unik  
Isi : *#NAME?, \*, \*\*, -, --, ., .beijing, .kunming, .shanghai, .shenzhen, .xishui, .zhaoyangqu, .zhongqing, /, @,*

Kolom **work\_province**

Terdapat 1166 nilai unik  
Isi : *#NAME?, \*, \*\*, -, --, ---, ., ., .beijing, .guangxizhuangzuzizhiqu, .hubei, .shanghai, .yunnan, .zhongqin,*

Kolom **work\_country**

Terdapat 119 nilai unik  
Isi : *AA, AB, AE, AN, AR, AS, AT, AU, AZ, BB, BD, BE, BG, BI, BN, BR, BS, BZ, CA, CC, CD, CH, CL, CM, CN, CR, CY,*

S



# DISTRIBUSI DATA

```

import plotly.express as px

def histogram_and_boxplot(data, columns, color):
    fig = px.histogram(
        data,
        x = columns,
        marginal = 'box',
        color_discrete_sequence = [color],
        nbins = 50
    )

    fig.update_traces(
        marker_line_width = 1,
        marker_line_color = 'white'
    )

    fig.update_layout(
        plot_bgcolor = 'rgba(0, 0, 0, 0)',
        title = dict(
            text = f"<b>Distribusi <span style='color:{color}'>{columns.capitalize()}</b></span>",
            font = dict(
                size = 28,
                color = '#757882'
            ),
            y = 0.92,
            x = 0.5
        ),
        yaxis = dict(
            title = '',
            showgrid = False,
            showline = False,
            showticklabels = False,
            zeroline = False,
        ),
        margin = dict(
            t = 80,
            b = 10,
            r = 20
        )
    )

    fig.show()

```

1

```
[ ] # Random warna dari Plotly
colors = px.colors.qualitative.Plotly
num_colors = len(number_data.columns)
palette = colors[:num_colors]
```

```
for column, color in zip(number_data.columns, palette):
    histogram_and_boxplot(
        data = number_data,
        columns = column,
        color = color
    )
```

2



**INNOVATORS**  
INFORMATION INNOVATORS

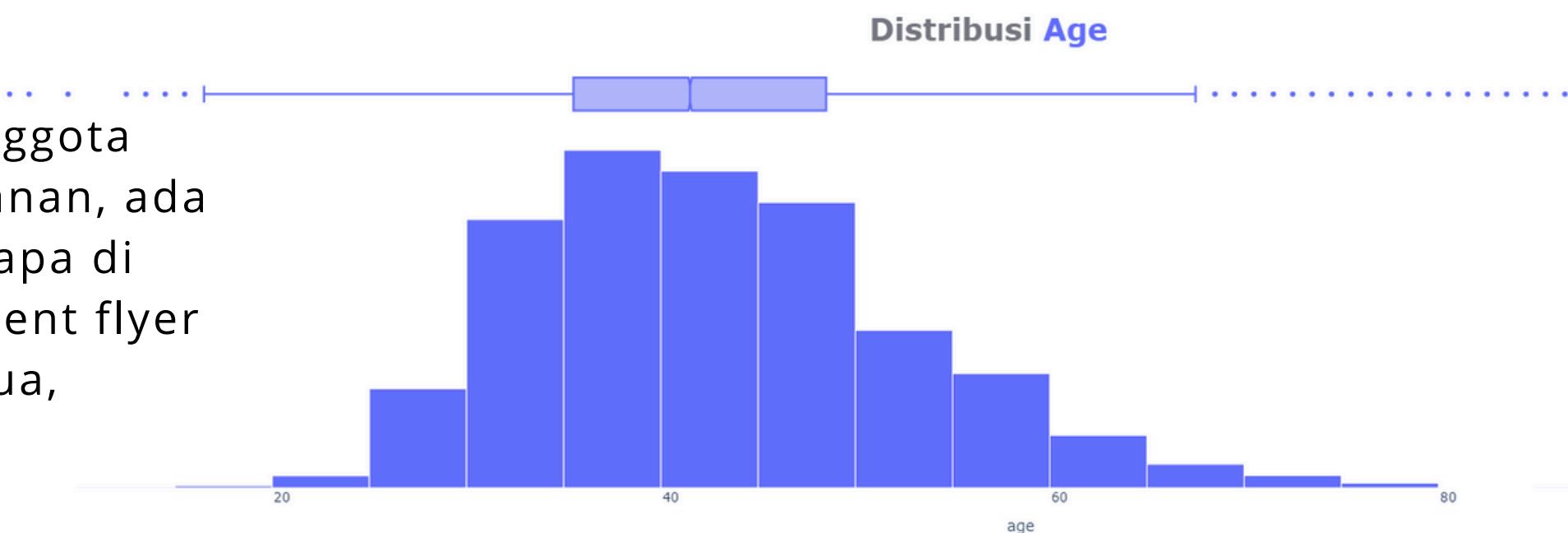


# DISTRIBUSI DATA

Distribusi usia ini cenderung sedikit skew ke kanan (positif), yang berarti ada lebih banyak anggota yang lebih muda dari median dibandingkan dengan yang jauh lebih tua.

Ada beberapa outlier di kedua sisi distribusi. Di sisi kiri, ada anggota yang berusia sangat muda (kurang dari 20 tahun), dan di sisi kanan, ada anggota yang lebih tua (mendekati 80 tahun dan bahkan beberapa di atas 100 tahun). Outlier ini menunjukkan bahwa program frequent flyer juga menarik bagi anggota yang sangat muda maupun sangat tua, meskipun jumlahnya kecil.

Program frequent flyer tampaknya menarik bagi orang dewasa dalam usia kerja produktif (35-50 tahun). Meskipun demikian, adanya anggota dari kelompok usia yang lebih muda dan lebih tua menunjukkan bahwa program ini memiliki daya tarik lintas generasi, meskipun dengan partisipasi yang lebih rendah di kalangan yang sangat muda atau sangat tua.



## DISTRIBUSI DATA

Distribusi `flight_count` sangat miring ke kanan (right-skewed). Mayoritas anggota memiliki jumlah penerbangan yang rendah, dengan sebagian besar anggota melakukan hanya beberapa penerbangan.

Frekuensi tertinggi ada pada nilai rendah, terutama antara 0 hingga sekitar 10 penerbangan.

Distribusi ini menunjukkan bahwa sebagian besar anggota adalah pelancong yang jarang, dengan hanya sedikit anggota yang melakukan penerbangan dalam jumlah besar. Outlier dan skewness menunjukkan adanya perbedaan besar dalam frekuensi perjalanan antar anggota, yang mungkin perlu ditangani dalam analisis clustering, misalnya dengan melakukan transformasi data atau mempertimbangkan skala logaritmik untuk mengurangi dampak outlier.



Distribusi `Flight_count`

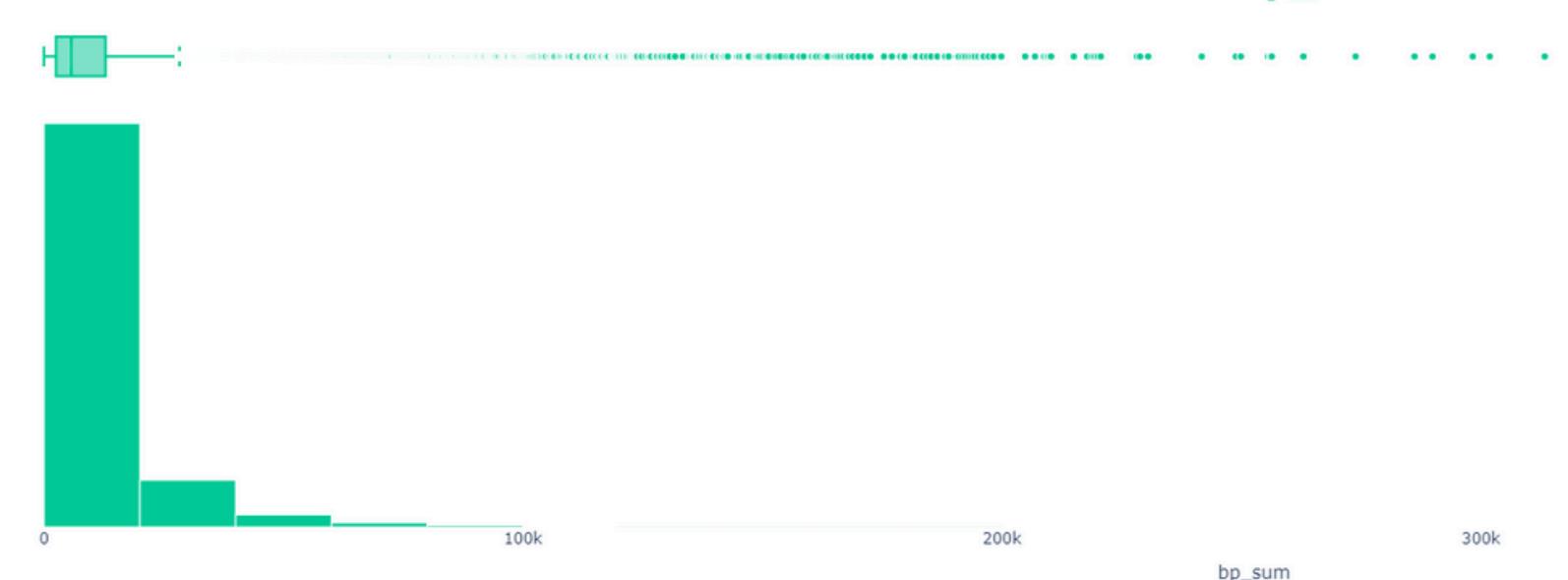


# DISTRIBUSI DATA

Distribusi bp\_sum menunjukkan skewness yang sangat kuat ke kanan (right-skewed). Sebagian besar anggota memiliki nilai bp\_sum yang rendah, dengan sebagian besar data terkonsentrasi di dekat nol.

Histogram memperlihatkan bahwa frekuensi tertinggi berada pada nilai bp\_sum yang sangat kecil, sementara sebagian kecil anggota memiliki nilai yang sangat tinggi.

Distribusi ini menunjukkan bahwa sebagian besar anggota memperoleh poin dalam jumlah kecil, tetapi ada beberapa anggota yang memperoleh jumlah poin yang sangat besar, membuat distribusi menjadi sangat miring ke kanan. Dalam analisis clustering, outlier ini mungkin perlu diperhatikan, dan mungkin perlu dipertimbangkan transformasi data (seperti log transformation) untuk mengurangi dampak dari nilai ekstrem tersebut pada hasil analisis.

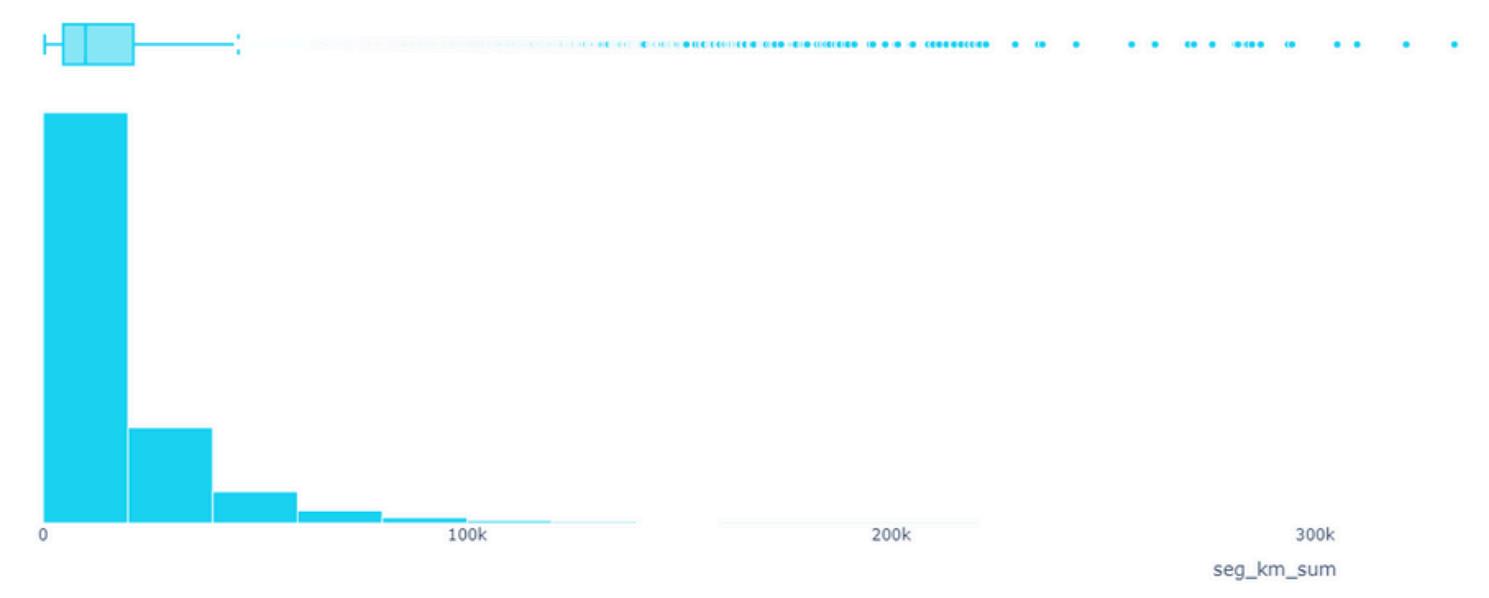


## DISTRIBUSI DATA

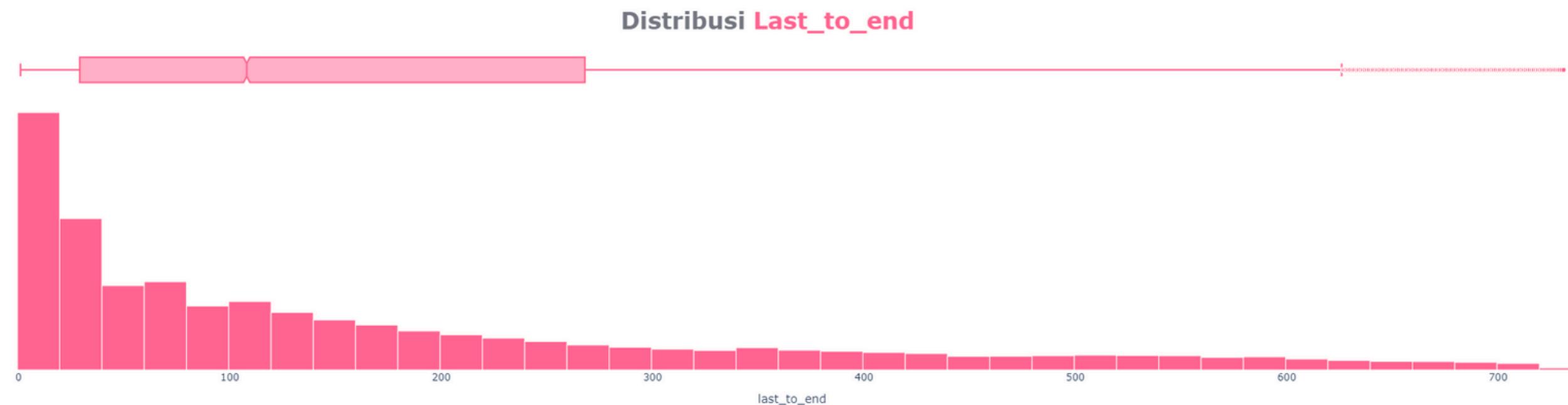
Distribusi seg\_km\_sum sangat miring ke kanan (right-skewed). Mayoritas anggota memiliki total jarak tempuh yang relatif rendah, dengan sebagian besar data berada pada nilai yang rendah.

Histogram menunjukkan bahwa jumlah anggota yang memiliki jarak tempuh rendah sangat tinggi dibandingkan dengan mereka yang menempuh jarak yang lebih jauh.

Distribusi ini mengindikasikan bahwa sebagian besar anggota menempuh jarak yang relatif pendek, sementara ada beberapa anggota yang menempuh jarak yang sangat jauh, menghasilkan distribusi yang sangat skewed ke kanan. Outlier dengan jarak tempuh yang sangat tinggi menunjukkan perbedaan besar dalam aktivitas perjalanan di antara anggota, yang perlu dipertimbangkan dalam analisis lebih lanjut, misalnya dalam konteks clustering, untuk menangani skewness yang kuat ini.



# DISTRIBUSI DATA



Distribusi ini menunjukkan bahwa mayoritas anggota memiliki jumlah hari yang relatif rendah antara penerbangan terakhir mereka dan akhir periode analisis. Semakin besar jumlah harinya, semakin sedikit anggota yang masuk dalam kategori tersebut, terlihat dari bar yang semakin menurun seiring dengan meningkatnya nilai.

Sebagian besar anggota frequent flyer melakukan penerbangan terakhir mereka baru-baru ini sebelum akhir periode analisis. Namun, ada beberapa anggota yang tidak terbang untuk waktu yang lama, dengan jarak hari yang sangat besar antara penerbangan terakhir dan akhir periode analisis.



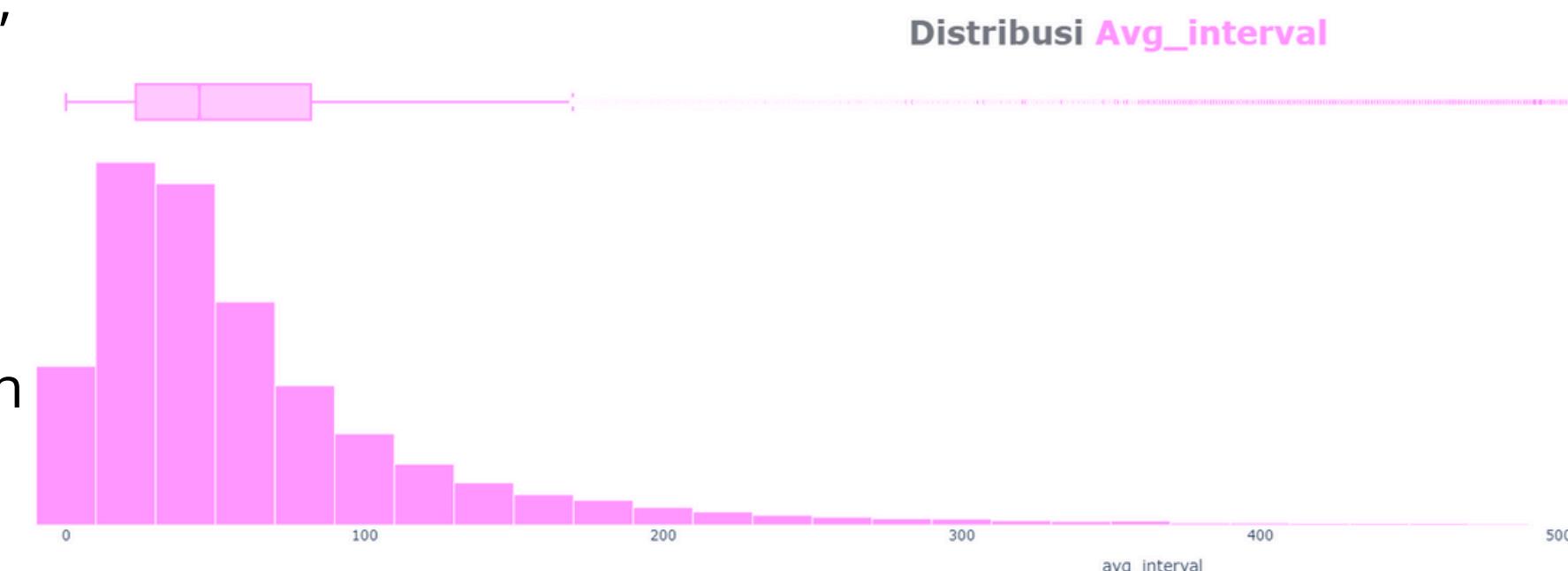
**INFLVATORS**  
INFORMATION INOVATORS



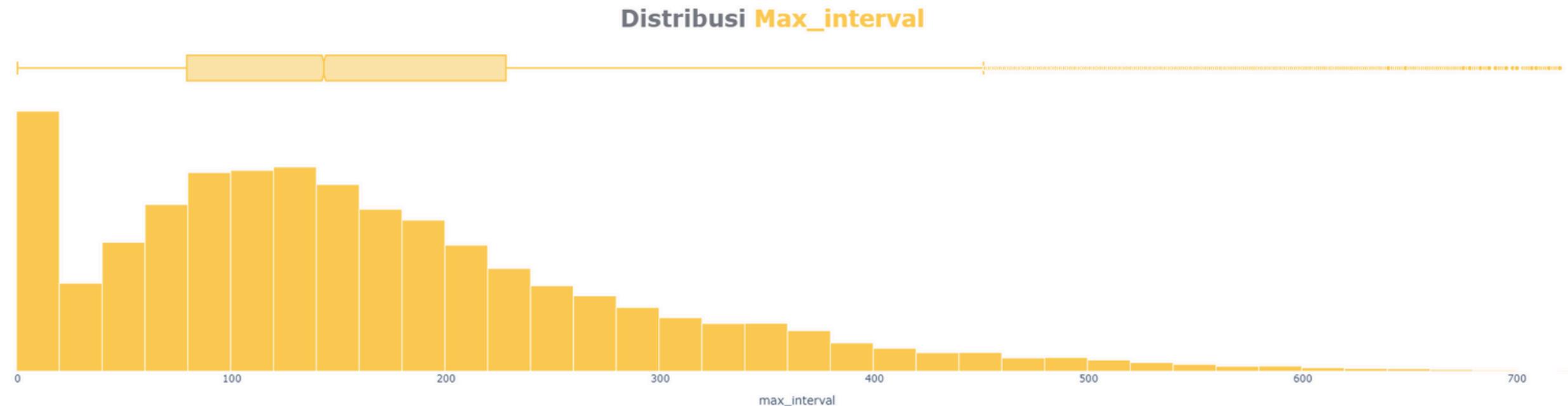
# DISTRIBUSI DATA

Boxplot menampilkan rentang interkuartil yang lebih kecil, dan distribusi memiliki beberapa outlier yang cukup jauh, menunjukkan bahwa beberapa anggota memiliki interval penerbangan yang sangat jarang.

Rata-rata interval waktu antara penerbangan menunjukkan bahwa sebagian besar anggota sering terbang, dengan jarak waktu yang relatif pendek antara satu penerbangan dan penerbangan berikutnya. Meskipun demikian, ada sejumlah kecil anggota yang jarang terbang, dengan rata-rata interval yang cukup panjang.



# DISTRIBUSI DATA



Boxplot menunjukkan rentang yang lebih luas dengan banyak outlier yang menunjukkan adanya anggota yang memiliki interval sangat panjang antara penerbangan.

Interval maksimum antara penerbangan menunjukkan bahwa banyak anggota mengalami jeda yang lebih panjang pada suatu titik waktu tertentu. Meskipun sebagian besar anggota tetap terbang secara konsisten dengan interval maksimum yang relatif pendek, ada beberapa anggota yang memiliki interval maksimum yang sangat panjang, menunjukkan periode tidak aktif yang signifikan antara penerbangan.



# KORELASI

## Korelasi Tiap Fitur

	age	flight_count	bp_sum	sum_yr_1	sum_yr_2	seg_km_sum	last_to_end	age_member	avg_interval	max_interval	exchange_count	avg_discount	points_sum	point_notflight
age	1	0.1059566	0.1317176	0.1364339	0.09633025	0.110779	-0.05111594	0.2908458	-0.02366379	0.01775001	0.02466551	0.09767054	0.1044729	-0.06310855
flight_count	0.1059566	1	0.8715729	0.6997854	0.7611619	0.8992717	-0.5866051	0.1853673	-0.3082468	0.05072182	0.3977223	0.1203202	0.8441005	0.3542313
bp_sum	0.1317176	0.8715729	1	0.7357198	0.7738222	0.9323577	-0.514482	0.1815903	-0.2984259	0.01818873	0.3948547	0.3564361	0.9577013	0.3306966
sum_yr_1	0.1364339	0.6997854	0.7357198	1	0.3656224	0.7186361	-0.1920214	0.1819097	-0.2499624	0.01143787	0.3537495	0.2413253	0.7039778	0.2746404
sum_yr_2	0.09633025	0.7611619	0.7738222	0.3656224	1	0.7729065	-0.7350683	0.1441969	-0.2154803	0.04483968	0.3331337	0.1582371	0.7538651	0.3169917
seg_km_sum	0.110779	0.8992717	0.9323577	0.7186361	0.7729065	1	-0.5391334	0.1705415	-0.2991579	0.02956187	0.3953744	0.05358154	0.8989517	0.3398081
last_to_end	-0.05111594	-0.5866051	-0.514482	-0.1920214	-0.7350683	-0.5391334	1	-0.1270018	-0.0368642	-0.252914	-0.2471423	-0.02500236	-0.5080775	-0.2459691
age_member	0.2908458	0.1853673	0.1815903	0.1819097	0.1441969	0.1705415	-0.1270018	1	-0.01663816	0.05676177	0.2568756	0.07111389	0.1813927	0.1616816
avg_interval	-0.02366379	-0.3082468	-0.2984259	-0.2499624	-0.2154803	-0.2991579	-0.0368642	-0.01663816	1	0.8171423	-0.1990715	-0.06237404	-0.2925378	-0.1762953
max_interval	0.01775001	0.05072182	0.01818873	0.01143787	0.04483968	0.02956187	-0.252914	0.05676177	0.8171423	1	-0.06738292	-0.02633823	0.01612081	-0.04919274
exchange_count	0.02466551	0.3977223	0.3948547	0.3537495	0.3331337	0.3953744	-0.2471423	0.2568756	-0.1990715	-0.06738292	1	0.08840689	0.42231	0.5792605
avg_discount	0.09767054	0.1203202	0.3564361	0.2413253	0.1582371	0.05358154	-0.02500236	0.07111389	-0.06237404	-0.02633823	0.08840689	1	0.3238891	0.04277157
points_sum	0.1044729	0.8441005	0.9577013	0.7039778	0.7538651	0.8989517	-0.5080775	0.1813927	-0.2925378	0.01612081	0.42231	0.3238891	1	0.483531
point_notflight	-0.06310855	0.3542313	0.3306966	0.2746404	0.3169917	0.3398081	-0.2459691	0.1616816	-0.1762953	-0.04919274	0.5792605	0.04277157	0.483531	1

### Korelasi Positif Tinggi:

**flight\_count, bp\_sum, seg\_km\_sum, points\_sum:** Semua fitur ini memiliki korelasi positif yang sangat kuat satu sama lain. Ini menunjukkan bahwa semakin sering seseorang terbang, semakin banyak kilometer yang ditempuh dan poin yang dikumpulkan.

**sum\_yr\_1 dan sum\_yr\_2:** Pengeluaran atau akumulasi poin di tahun pertama dan kedua sangat terkait, menunjukkan konsistensi perilaku pelanggan dari tahun ke tahun.

### Korelasi Negatif Tinggi:

**last\_to\_end dengan sum\_yr\_2 dan seg\_km\_sum:** Semakin lama waktu sejak aktivitas terakhir, semakin rendah pengeluaran atau jarak tempuh, menandakan pentingnya keterlibatan terus-menerus.



# KORELASI

## Korelasi Tiap Fitur

	age	flight_count	bp_sum	sum_yr_1	sum_yr_2	seg_km_sum	last_to_end	age_member	avg_interval	max_interval	exchange_count	avg_discount	points_sum	point_notflight
age	1	0.1059566	0.1317176	0.1364339	0.09633025	0.110779	-0.05111594	0.2908458	-0.02366379	0.01775001	0.02466551	0.09767054	0.1044729	-0.06310855
flight_count	0.1059566	1	0.8715729	0.6997854	0.7611619	0.8992717	-0.5866051	0.1853673	-0.3082468	0.05072182	0.3977223	0.1203202	0.8441005	0.3542313
bp_sum	0.1317176	0.8715729	1	0.7357198	0.7738222	0.9323577	-0.514482	0.1815903	-0.2984259	0.01818873	0.3948547	0.3564361	0.9577013	0.3306966
sum_yr_1	0.1364339	0.6997854	0.7357198	1	0.3656224	0.7186361	-0.1920214	0.1819097	-0.2499624	0.01143787	0.3537495	0.2413253	0.7039778	0.2746404
sum_yr_2	0.09633025	0.7611619	0.7738222	0.3656224	1	0.7729065	-0.7350683	0.1441969	-0.2154803	0.04483968	0.3331337	0.1582371	0.7538651	0.3169917
seg_km_sum	0.110779	0.8992717	0.9323577	0.7186361	0.7729065	1	-0.5391334	0.1705415	-0.2991579	0.02956187	0.3953744	0.05358154	0.8989517	0.3398081
last_to_end	-0.05111594	-0.5866051	-0.514482	-0.1920214	-0.7350683	-0.5391334	1	-0.1270018	-0.0368642	-0.252914	-0.2471423	-0.02500236	-0.5080775	-0.2459691
age_member	0.2908458	0.1853673	0.1815903	0.1819097	0.1441969	0.1705415	-0.1270018	1	-0.01663816	0.05676177	0.2568756	0.07111389	0.1813927	0.1616816
avg_interval	-0.02366379	-0.3082468	-0.2984259	-0.2499624	-0.2154803	-0.2991579	-0.0368642	-0.01663816	1	0.8171423	-0.1990715	-0.06237404	-0.2925378	-0.1762953
max_interval	0.01775001	0.05072182	0.01818873	0.01143787	0.04483968	0.02956187	-0.252914	0.05676177	0.8171423	1	-0.06738292	-0.02633823	0.01612081	-0.04919274
exchange_count	0.02466551	0.3977223	0.3948547	0.3537495	0.3331337	0.3953744	-0.2471423	0.2568756	-0.1990715	-0.06738292	1	0.08840689	0.42231	0.5792605
avg_discount	0.09767054	0.1203202	0.3564361	0.2413253	0.1582371	0.05358154	-0.02500236	0.07111389	-0.06237404	-0.02633823	0.08840689	1	0.3238891	0.04277157
points_sum	0.1044729	0.8441005	0.9577013	0.7039778	0.7538651	0.8989517	-0.5080775	0.1813927	-0.2925378	0.01612081	0.42231	0.3238891	1	0.483531
point_notflight	-0.06310855	0.3542313	0.3306966	0.2746404	0.3169917	0.3398081	-0.2459691	0.1616816	-0.1762953	-0.04919274	0.5792605	0.04277157	0.483531	1

### Korelasi Sedang:

**avg\_interval dan max\_interval:** Korelasi kuat menunjukkan pola keteraturan dalam aktivitas penerbangan atau transaksi.

**exchange\_count dan points\_sum:** Hubungan sedang antara **jumlah pertukaran dan total poin** menunjukkan bahwa pertukaran poin sering terjadi pada pelanggan dengan banyak poin.

### Korelasi Rendah:

**age** dengan sebagian besar fitur lainnya: Usia tidak berkorelasi signifikan dengan fitur-fitur terkait penerbangan atau poin, menunjukkan bahwa usia bukan faktor penentu utama dalam akumulasi poin atau aktivitas penerbangan.

### Kesimpulan:

Fitur-fitur yang berhubungan dengan penerbangan dan akumulasi poin (seperti **flight\_count, seg\_km\_sum, bp\_sum, dan points\_sum**) sangat berkorelasi dan penting untuk analisis lebih lanjut. Menjaga keterlibatan pelanggan adalah kunci, seperti yang ditunjukkan oleh korelasi negatif dengan last\_to\_end.





**INFOLVATORS**  
INFORMATION INOVATORS

# FEATURE ENGINEERING



**INFOLVATORS**  
INFORMATION INOVATORS



```
# Hitung tanggal analisis
end_date_now = (data_flight['last_flight_date'] + pd.to_timedelta(data_flight['last_to_end'], unit = 'd')).max()

# Hitung selisih hari
age_member = (end_date_now - data_flight['ffp_date']).dt.days

# Masukan data ke dataframe
data_flight.insert(17, 'age_member', age_member)
```

data\_flight

	member_no	ffp_date	first_flight_date	gender	ffp_tier	work_city	work_province	work_country	age	load_time	flight_count	bp_sum	sum_yr_1	sum_yr_2	seg_km_sum	last_flight_date	last_to_end	age_member	avg_interval	max
0	54993	2006-11-02	2008-12-24	Male	6	.	beijing	CN	31.0	2014-03-31	210	505308	239560.0	234188.0	580717	2014-03-31	1	2708	3.483254	
1	28065	2007-02-19	2007-08-03	Male	6	Nan	beijing	CN	42.0	2014-03-31	140	362480	171483.0	167434.0	293678	2014-03-25	7	2599	5.194245	
2	55106	2007-02-01	2007-08-30	Male	6	.	beijing	CN	40.0	2014-03-31	135	351159	163618.0	164982.0	283712	2014-03-21	11	2617	5.298507	
3	21189	2008-08-22	2008-08-23	Male	5	Los Angeles	CA	US	64.0	2014-03-31	23	337314	116350.0	125500.0	281336	2013-12-26	97	2049	27.863636	
4	39546	2009-04-10	2009-04-15	Male	6	guiyang	guizhou	CN	48.0	2014-03-31	152	273844	124560.0	130702.0	309928	2014-03-27	5	1818	4.788079	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
62983	18375	2011-05-20	2013-06-05	Female	4	guangzhou	guangdong	CN	25.0	2014-03-31	2	0	0.0	0.0	1134	2013-06-09	297	1048	4.000000	
62984	36041	2010-03-08	2013-09-14	Male	4	foshan	guangdong	CN	38.0	2014-03-31	4	0	0.0	0.0	8016	2014-01-03	89	1486	37.000000	
62985	45690	2006-03-30	2006-12-02	Female	4	guangzhou	guangdong	CN	43.0	2014-03-31	2	0	0.0	0.0	2594	2014-03-03	29	2925	166.000000	
62986	61027	2013-02-06	2013-02-14	Female	4	guangzhou	guangdong	CN	36.0	2014-03-31	2	0	0.0	0.0	3934	2013-02-26	400	420	12.000000	
62987	61340	2013-02-17	2013-02-17	Female	4	shanghai	.	CN	29.0	2014-03-31	2	0	0.0	0.0	4222	2013-02-23	403	409	6.000000	

62988 rows × 24 columns

Selisih antara Tanggal Analisis dan Tanggal Join Member dalam satuan hari.

Tanggal analisis didapatkan dari penjumlahan kolom last\_flight\_date dengan last\_to\_end lalu cari yang paling maksimum

R S



INFLATORS  
INFORMATION INOVATORS





# MODELING + EVALUASI



# Features Selection

```
[ ] data_clustering = data_flight[['member_no', 'ffp_tier', 'flight_count', 'seg_km_sum', 'age_member']]  
data_clustering
```

	member_no	ffp_tier	flight_count	seg_km_sum	age_member
0	54993	6	210	580717	2708
1	28065	6	140	293678	2599
2	55106	6	135	283712	2617
3	21189	5	23	281336	2049
4	39546	6	152	309928	1818
...	...	...	...	...	...
62983	18375	4	2	1134	1048
62984	36041	4	4	8016	1486

Fitur-fitur yang dipilih dalam data\_clustering dimaksudkan untuk menangkap aspek-aspek penting dari profil setiap anggota yang relevan untuk clustering:

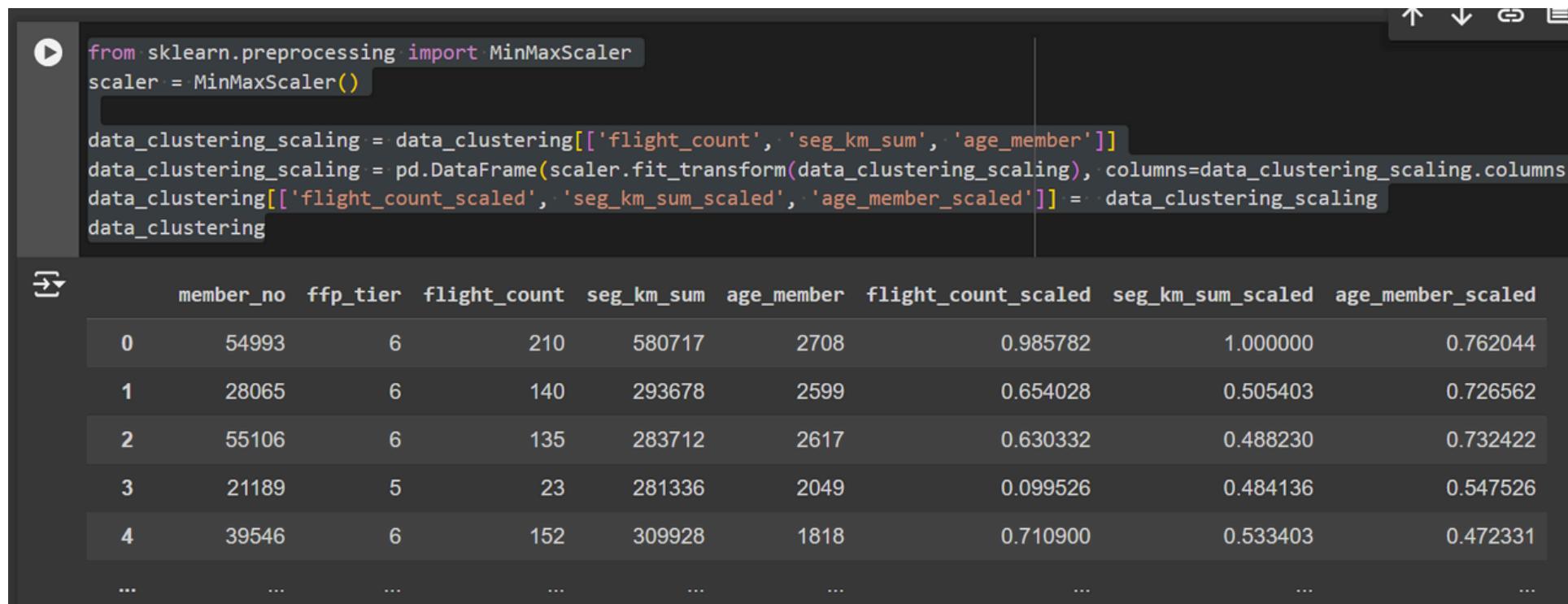
- member\_no memberikan identifikasi unik untuk setiap anggota, memungkinkan Anda untuk melacak kembali klaster-klaster ke individu tertentu.
- ffp\_tier memberikan wawasan tentang tingkat loyalitas anggota, yang mungkin berkorelasi dengan perilaku dan preferensi mereka.
- flight\_count dan seg\_km\_sum keduanya merupakan fitur yang berbasis aktivitas yang membantu mengidentifikasi seberapa sering dan seberapa jauh anggota bepergian, yang merupakan faktor penting dalam mengelompokkan berbagai jenis pelancong.
- age\_member bisa penting untuk mengidentifikasi tren atau preferensi di antara kelompok usia yang berbeda, yang dapat mempengaruhi perilaku perjalanan.



**INFLATORS**  
INFORMATION INOVATORS



# Normalisasi data



```
from sklearn.preprocessing import MinMaxScaler
scaler = MinMaxScaler()

data_clustering_scaling = data_clustering[['flight_count', 'seg_km_sum', 'age_member']]
data_clustering_scaling = pd.DataFrame(scaler.fit_transform(data_clustering_scaling), columns=data_clustering_scaling.columns)
data_clustering[['flight_count_scaled', 'seg_km_sum_scaled', 'age_member_scaled']] = data_clustering_scaling
data_clustering
```

	member_no	ffp_tier	flight_count	seg_km_sum	age_member	flight_count_scaled	seg_km_sum_scaled	age_member_scaled
0	54993	6	210	580717	2708	0.985782	1.000000	0.762044
1	28065	6	140	293678	2599	0.654028	0.505403	0.726562
2	55106	6	135	283712	2617	0.630332	0.488230	0.732422
3	21189	5	23	281336	2049	0.099526	0.484136	0.547526
4	39546	6	152	309928	1818	0.710900	0.533403	0.472331
...	...	...	...	...	...	...	...	...

- Normalisasi Data dalam fitur `flight_count`, `seg_km_sum`, dan `age_member`. karena K-means sensitif terhadap skala fitur. Jika fitur-fitur memiliki skala yang sangat berbeda, fitur yang memiliki nilai besar dapat mendominasi hasil clustering.
- Dengan menormalisasi data, semua fitur akan berada dalam rentang yang sama, yang membuat algoritma clustering lebih efektif dan hasilnya lebih akurat.



# Elbow Method

```
[ ] from sklearn.cluster import KMeans
import matplotlib.pyplot as plt

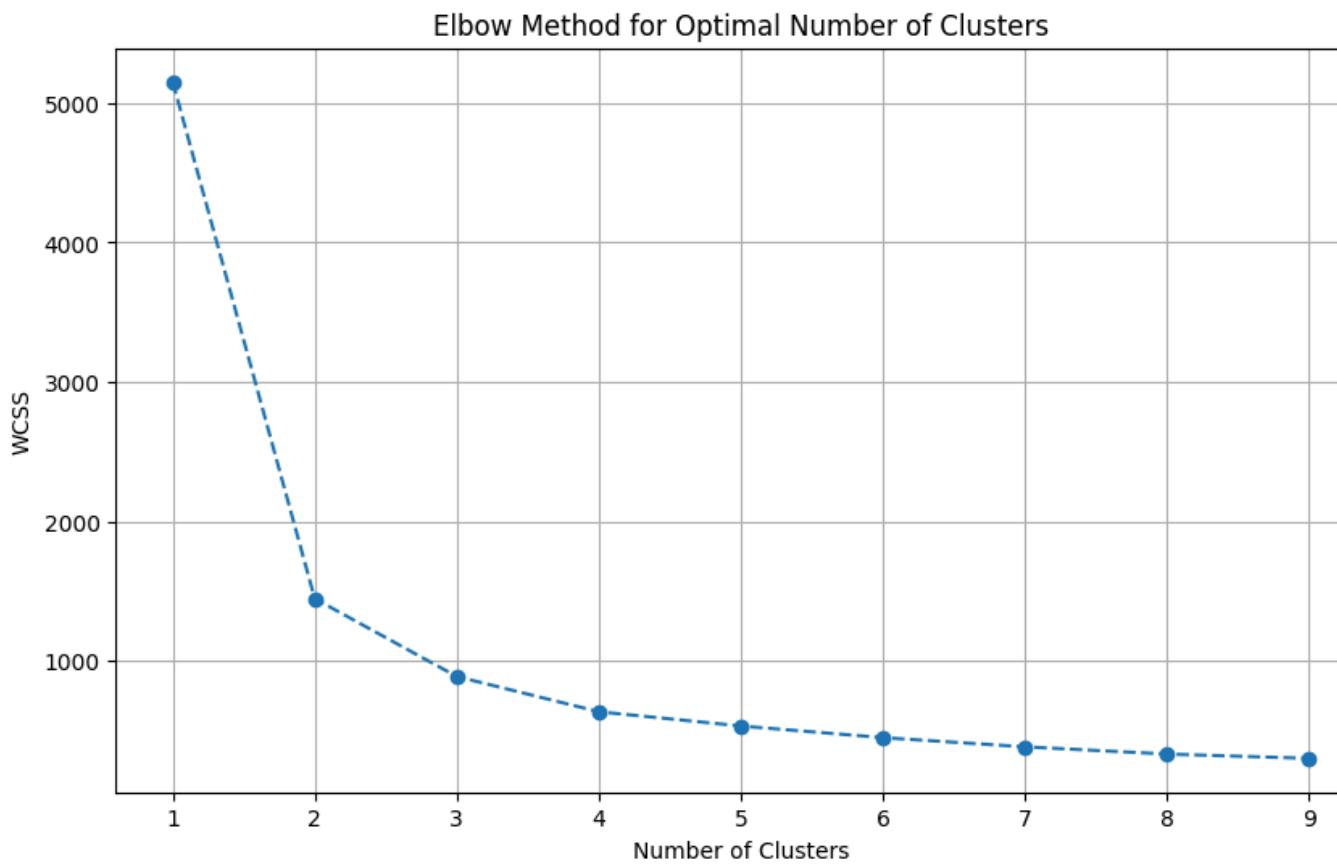
# Menghasilkan data sintetis
X = data_clustering[['flight_count_scaled', 'seg_km_sum_scaled', 'age_member_scaled']]

# Rentang jumlah cluster yang akan diuji
range_n_clusters = range(1, 10)
wcss = []

for n_clusters in range_n_clusters:
    # Inisialisasi KMeans dengan jumlah cluster tertentu
    kmeans = KMeans(n_clusters=n_clusters, random_state=42)
    kmeans.fit(X)

    # Hitung WCSS (Within-Cluster Sum of Squares)
    wcss.append(kmeans.inertia_)

# Plot hasil untuk Elbow method
plt.figure(figsize=(10, 6))
plt.plot(range_n_clusters, wcss, marker='o', linestyle='--')
plt.title('Elbow Method for Optimal Number of Clusters')
plt.xlabel('Number of Clusters')
plt.ylabel('WCSS')
plt.xticks(range_n_clusters)
plt.grid(True)
plt.show()
```



Sesuai dengan interpretasi dari grafik diatas maka kita pilih jumlah clusternya 2 ( $k=2$ ) karena titik 2 karena ini adalah titik di mana menambahkan cluster lain tidak menghasilkan pengurangan WCSS (Within-Cluster Sum of Squares) yang signifikan, yang menunjukkan bahwa 2 cluster mungkin merupakan pilihan yang optimal.

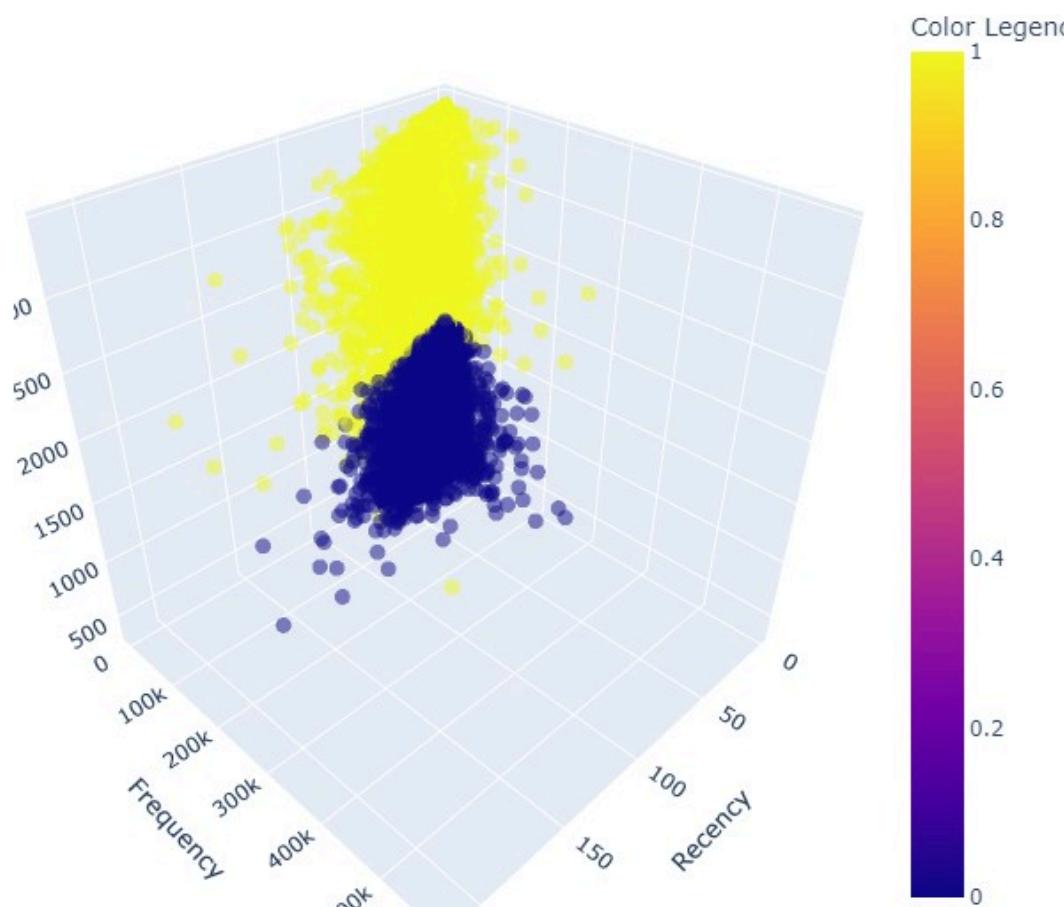


**INFLVATORS**  
INFORMATION INOVATORS



# Elbow Method

Customer Segmentation



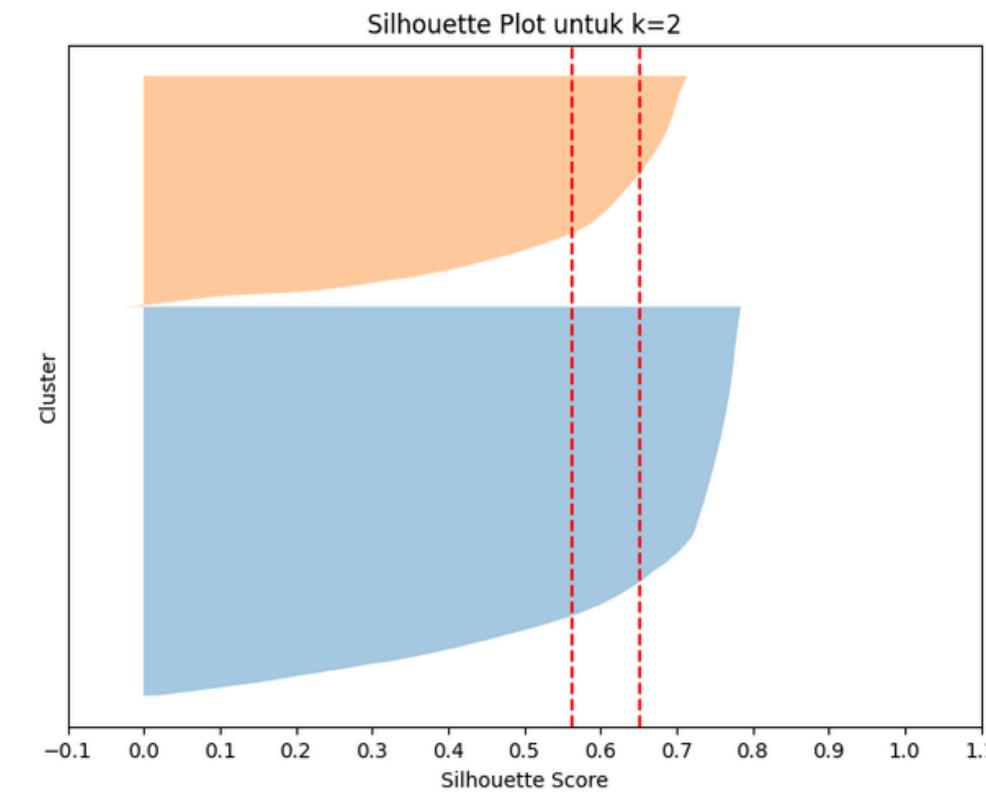
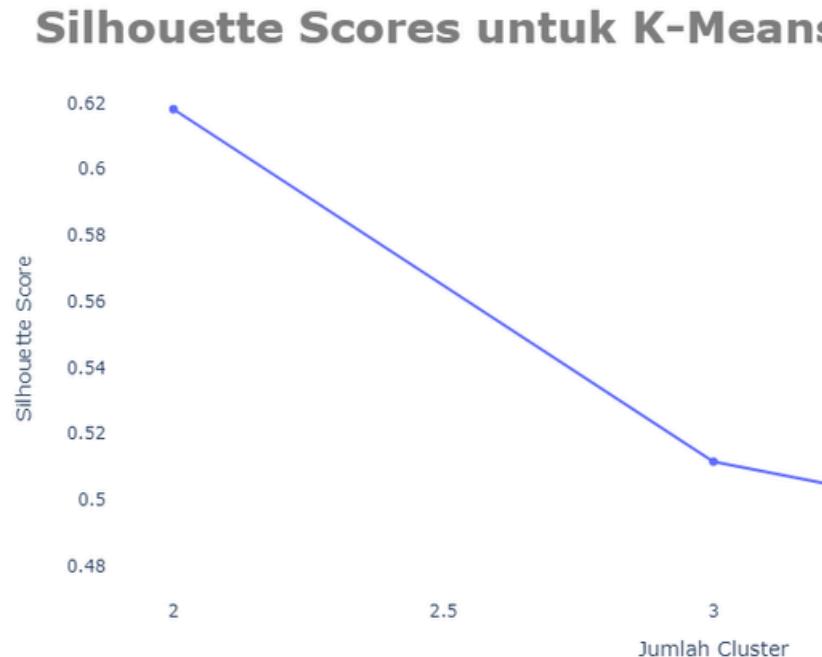
- Sumbu X (Recency/Kerecencian): Sumbu ini menunjukkan seberapa baru seorang pelanggan melakukan pembelian.
- Sumbu Y (Frequency/Frekuensi): Sumbu ini menunjukkan seberapa sering seorang pelanggan melakukan pembelian.
- Sumbu Z (Monetary Value/Nilai Uang): Sumbu ini menunjukkan jumlah uang yang dihabiskan oleh pelanggan.
- Bar warna di sebelah kanan menunjukkan intensitas atau kepadatan dari cluster. Warna berkisar dari biru tua (nilai lebih rendah) hingga kuning (nilai lebih tinggi). Setiap warna mewakili cluster yang berbeda, dengan kuning biasanya menunjukkan kepadatan yang lebih tinggi atau pengelompokan titik yang lebih rapat.
- Plot ini menampilkan dua cluster pelanggan yang berbeda. Cluster dalam warna biru tua menunjukkan satu kelompok dengan perilaku atau karakteristik pelanggan baru, sementara cluster kuning menunjukkan pelanggan lama.



**INFLVATORS**  
INFORMATION INOVATORS



# Evaluasi Cluster



- Pada  $K = 2$ , nilai Silhouette Score adalah yang tertinggi, sekitar 0.62. Ini menunjukkan bahwa dua cluster mungkin adalah jumlah cluster yang paling optimal, karena objek dalam setiap cluster lebih baik dipisahkan dari objek di cluster lain.
- Saat jumlah cluster meningkat menjadi  $K = 3$  dan  $K = 4$ , Silhouette Score menurun secara signifikan menjadi sekitar 0.51 dan 0.48, menunjukkan bahwa penambahan cluster mengurangi efektivitas clustering. Ini bisa berarti bahwa dengan menambahkan lebih banyak cluster, beberapa objek menjadi lebih dekat ke batas cluster lain, mengurangi kualitas clustering.
- Dari Gambar kedua kita bisa melihat bahwa cluster terpisah dengan baik (tidak ada irisannya) ini menunjukkan bahwa  $k=2$  merupakan jumlah cluster yang cukup optimal. Tiap kluster juga masing masing memiliki skor yang cukup tinggi dimana keduanya diatas 0.55
- Dari plot ini, dapat disimpulkan bahwa  $K = 2$  adalah jumlah cluster yang paling optimal berdasarkan Silhouette Score. Dengan menambah jumlah cluster menjadi 3 atau 4, kinerja clustering justru menurun, yang bisa menunjukkan bahwa data tersebut lebih baik dikelompokkan ke dalam dua cluster.





# **INTERPRETASI CLUSTER + REKOMENDASI BISNIS**



# Konsektual Customer sesuai Cluster

```
data_clustering['customer_type'] = data_clustering['cluster'].apply(lambda x: 'pelanggan baru'
                                         if x == 0 else 'pelanggan lama'
                                         if x == 1 else 'other')

data_clustering
```

**Membuat Kolom baru untuk menampilkan pada Data Untuk membedakan Pelanggan Baru dan Pelanggan Lama**

1. Pelanggan Baru = 0
2. Pelanggan lama = 1

	member_no	ffp_tier	flight_count	seg_km_sum	age_member	flight_count_scaled	seg_km_sum_scaled	age_member_scaled	cluster	customer_type
0	54993	6	210	580717	2708	0.985782	1.000000	0.762044	1	pelanggan lama
1	28065	6	140	293678	2599	0.654028	0.505403	0.726562	1	pelanggan lama
2	55106	6	135	283712	2617	0.630332	0.488230	0.732422	1	pelanggan lama
3	21189	5	23	281336	2049	0.099526	0.484136	0.547526	1	pelanggan lama
4	39546	6	152	309928	1818	0.710900	0.533403	0.472331	1	pelanggan lama
...	...	...	...	...	...	...	...	...	...	...
62983	18375	4	2	1134	1048	0.000000	0.001320	0.221680	0	pelanggan baru
62984	36041	4	4	8016	1486	0.009479	0.013178	0.364258	0	pelanggan baru
62985	45690	4	2	2594	2925	0.000000	0.003836	0.832682	1	pelanggan lama
62986	61027	4	2	3934	420	0.000000	0.006145	0.017253	0	pelanggan baru
62987	61340	4	2	4222	409	0.000000	0.006641	0.013672	0	pelanggan baru



# Konsektual Customer sesuai Cluster

	member_no	ffp_tier	flight_count	seg_km_sum	age_member	flight_count_scaled	seg_km_sum_scaled	age_member_scaled	cluster	customer_type
0	54993	6	210	580717	2708	0.985782	1.000000	0.762044	1	pelanggan lama
1	28065	6	140	293678	2599	0.654028	0.505403	0.726562	1	pelanggan lama
2	55106	6	135	283712	2617	0.630332	0.488230	0.732422	1	pelanggan lama
3	21189	5	23	281336	2049	0.099526	0.484136	0.547526	1	pelanggan lama
4	39546	6	152	309928	1818	0.710900	0.533403	0.472331	1	pelanggan lama
...	...	...	...	...	...	...	...	...	...	...
62983	18375	4	2	1134	1048	0.000000	0.001320	0.221680	0	pelanggan baru
62984	36041	4	4	8016	1486	0.009479	0.013178	0.364258	0	pelanggan baru
62985	45690	4	2	2594	2925	0.000000	0.003836	0.832682	1	pelanggan lama
62986	61027	4	2	3934	420	0.000000	0.006145	0.017253	0	pelanggan baru
62987	61340	4	2	4222	409	0.000000	0.006641	0.013672	0	pelanggan baru

Membuat Kolom baru untuk menampilkan pada Data Untuk membedakan Pelanggan Baru dan Pelanggan Lama

1. Pelanggan Baru = 0 = Pelanggan yang memiliki age\_number yang lebih rendah dan juga jarak kilometer penerbangan yang rendah menandakan bahwa ia masih baru untuk penggunaan penerbangan ini
2. Pelanggan lama = 1 = Pelanggan yang memiliki age number yang tinggi, memilik jarak kilometer yang jauh menandakan bahwa ia sudah sering menggunakan penerbangan dan juga dengan jarak yang jauh.



# Rekomendasi Bussines

## 1. Untuk Cluster Pelanggan Lama (Cluster 1)

### Identifikasi Karakteristik Pelanggan Setia:

**Frekuensi Tinggi:** Pelanggan di cluster ini sering menggunakan layanan penerbangan, mungkin melakukan perjalanan secara berkala untuk keperluan bisnis atau pribadi.

**Loyalitas Tinggi:** Mereka telah menjadi anggota program untuk waktu yang lama, menunjukkan tingkat loyalitas yang kuat terhadap layanan maskapai.

## 2. Untuk Cluster Pelanggan Baru (Cluster 0)

### Identifikasi Karakteristik Pelanggan Baru:

**Frekuensi Rendah:** Pelanggan ini baru mulai menggunakan layanan maskapai dan mungkin hanya melakukan sedikit perjalanan.

**Fleksibilitas dan Eksplorasi:** Sebagai pelanggan baru, mereka mungkin masih dalam tahap eksplorasi, mencari penawaran yang menarik dan layanan yang memberikan nilai terbaik.



# Rekomendasi Bussines

## 1. Untuk Cluster Pelanggan Lama (Cluster 1)

### Program Loyalitas Tingkat Lanjut (Kategori 1):

- **Penawaran Khusus Upgrade Gratis:** Untuk meningkatkan kepuasan dan retensi, tawarkan kepada pelanggan setia upgrade gratis ke kelas yang lebih tinggi (misalnya, dari ekonomi ke bisnis) pada penerbangan tertentu. Ini tidak hanya meningkatkan pengalaman mereka tetapi juga memperkuat hubungan dengan maskapai penerbangan.
- **Akses Eksklusif ke Lounge Bandara:** Berikan akses gratis atau berlangganan ke lounge bandara yang nyaman. Fasilitas ini akan memberikan pengalaman lebih santai dan nyaman sebelum penerbangan, yang akan dihargai oleh pelanggan setia.

### Lakukan juga Personalized Offers:

- **Komunikasi Personal:** Kirimkan penawaran yang dipersonalisasi berdasarkan preferensi perjalanan mereka, seperti diskon untuk tujuan favorit mereka atau penawaran khusus pada waktu-waktu tertentu dalam setahun yang mereka sering bepergian.
- **Event Eksklusif:** Undang pelanggan setia ke acara eksklusif yang diadakan oleh maskapai, seperti peluncuran rute baru atau acara networking untuk pebisnis. Ini memberikan nilai tambah dan membuat mereka merasa dihargai oleh pihak maskapai.



**INNOVATORS**  
INFORMATION INOVATORS



# Rekomendasi Bussines

## 2. Untuk Cluster Pelanggan Baru (Cluster 0)

### Program Incentif Awal:

- **Diskon untuk Penerbangan Kedua:** Untuk mendorong mereka melakukan perjalanan lebih sering, tawarkan diskon atau penawaran khusus pada penerbangan kedua. Misalnya, jika mereka baru saja melakukan penerbangan pertama, berikan diskon 20% untuk penerbangan kedua dalam periode waktu tertentu.
- **Penawaran Paket:** Tawarkan paket penerbangan yang menggabungkan beberapa penerbangan dengan harga diskon, sehingga mereka merasa mendapatkan nilai lebih dari setiap pembelian tiket.

### Engagement Berkelanjutan:

- **Edukasi tentang Program Loyalitas:** Buat kampanye yang mendidik pelanggan baru tentang manfaat dari program loyalitas maskapai. Gunakan email, media sosial, dan aplikasi seluler untuk menyampaikan informasi tentang cara mengumpulkan poin, menebusnya, dan keuntungan menjadi anggota.
- **Follow-Up Email:** Kirimkan email follow-up dengan penawaran yang relevan setelah penerbangan pertama mereka. Misalnya, jika mereka terbang untuk liburan, tawarkan diskon untuk penerbangan liburan berikutnya atau layanan tambahan seperti asuransi perjalanan atau bagasi tambahan.





**SEKIAN TERIMAKASIH**

