

Modul Praktikum Data Mining



Tim Penyusun:

Dr. Rakhmat Arianto, S.ST., M.Kom

Ir. Rudy Ariyanto, ST., M.Cs

Prof. Dr. Eng. Rosa Andrie Asmara, ST., MT

Jurusan Teknologi Informasi

Sistem Informasi Bisnis

Politeknik Negeri Malang

Februari 2025

DAFTAR ISI

DAFTAR ISI	2
JOBSHEET 2 Pengumpulan Data.....	3
Pendahuluan	3
Tujuan Praktikum.....	3
Visual Objectives	Error! Bookmark not defined.
Peralatan yang dibutuhkan	3
Praktikum	3
Pengumpulan Data Secara Manual	Error! Bookmark not defined.
Pengumpulan Data Menggunakan API.....	Error! Bookmark not defined.
Latihan.....	Error! Bookmark not defined.
Tugas Praktikum	Error! Bookmark not defined.

JOBSHEET 5

Menentukan Objek Data

Pendahuluan

Modul ini menjelaskan proses Menentukan Objek Data dengan menggunakan metode Correlation dan implementasi dari metode Sampling Slovin yang dilengkapi dengan langkah-langkah detail serta gambar untuk memudahkan pemahaman.

Tujuan Praktikum

Setelah menyelesaikan praktikum ini, mahasiswa mampu:

- Memahami tentang Metode Correlation.
- Memahami tentang Metode Sampling Slovin.

Peralatan yang dibutuhkan

Beberapa peralatan yang dibutuhkan dalam menyelesaikan praktikum ini adalah:

- Aplikasi Microsoft Excel
- Google Colab
- Google Drive
- Koneksi Internet
- Browser Web

Praktikum

Implementasi Metode Correlation Menggunakan Ms. Excel

Lakukan praktikum sesuai tahapan berikut:

- a. Buka aplikasi web browser
- b. Unduh file contoh praktikum pada [Hitung Korelasi.xlsx](#)
- c. Pada file Excel tersebut telah terdapat data sebagai berikut:

ID	Luas Tanah (m ²)	Jumlah Kamar	Jarak ke Pusat Kota (km)	Harga (Juta Rupiah)
1	100	3	5	500
2	150	4	10	450
3	80	2	2	400
4	120	3	8	350
5	200	5	15	700
6	90	2	7	300
7	130	3	12	380
8	110	3	6	480
9	140	4	9	420
10	95	2	4	320

- d. Berdasarkan data tersebut, maka jumlah keseluruhan data adalah 10 data

ID	Luas Tanah (m ²)	Jumlah Kamar	Jarak ke Pusat Kota (km)	Harga (Juta Rupiah)
1	100	3	5	500
2	150	4	10	450
3	80	2	2	400
4	120	3	8	350
5	200	5	15	700
6	90	2	7	300
7	130	3	12	380
8	110	3	6	480
9	140	4	9	420
10	95	2	4	320

- e. Hitung korelasi dari setiap kolom terhadap Kolom **Harga** dengan menggunakan rumus:

$$r_{xy} = \frac{n \sum x_i y_i - (\sum x_i)(\sum y_i)}{\sqrt{(n \sum x_i^2 - (\sum x_i)^2)(n \sum y_i^2 - (\sum y_i)^2)}}$$

Dimana:

- r = nilai korelasi
- x = variabel x
- y = variabel y

- f. Hitung Korelasi antara **Harga** dengan **Luas Tanah** maka:

➤ X = Luas Tanah dan Y = Harga

- g. Buat tabel dataset sesuai dengan kebutuhan dari rumus Correlation

ID	x	y	xy	x ²	y ²
1	100	500	50000	10000	250000
2	150	450	67500	22500	202500
3	80	400	32000	6400	160000
4	120	350	42000	14400	122500
5	200	700	140000	40000	490000
6	90	300	27000	8100	90000
7	130	380	49400	16900	144400
8	110	480	52800	12100	230400
9	140	420	58800	19600	176400
10	95	320	30400	9025	102400
Total	1215	4300	549900	159025	1968600

ID	x	y	xy	x ²	y ²
1	100	500	50000	10000	250000
2	150	450	67500	22500	202500
3	80	400	32000	6400	160000
4	120	350	42000	14400	122500
5	200	700	140000	40000	490000
6	90	300	27000	8100	90000
7	130	380	49400	16900	144400
8	110	480	52800	12100	230400
9	140	420	58800	19600	176400
10	95	320	30400	9025	102400
Total	1215	4300	549900	159025	1968600

h. Hitung nilai r sesuai dengan rumus, maka akan didapatkan nilai korelasinya $r = 0,743321541$

Maka r Harga dengan Luas Tanah	
r atas	274500
r bawah	369288.3697
r	0.7433215409

i. Interpretasi dari nilai r tersebut adalah

- Harga dengan Luas Tanah memiliki korelasi positif yang sangat erat
- Harga akan naik jika luas tanah semakin besar

LATIHAN

1. Salin data dari Ms. Excel ke Google SpreadSheet
2. Lakukan perhitungan untuk korelasi dari:
 - a. Harga dengan Jumlah Kamar

ID	x (Jumlah Kamar)	y (Harga)	xy	x ²	y ²	N	10
1	3	500	1500	9	250000		
2	4	450	1800	16	202500		
3	2	400	800	4	160000		
4	3	350	1050	9	122500		
5	5	700	3500	25	490000		
6	2	300	600	4	90000		
7	3	380	1140	9	144400		
8	3	480	1440	9	230400		
9	4	420	1680	16	176400		
10	2	320	640	4	102400		
Total	31	4300	14150	105	1968600		

b. Harga dengan Jarak Ke Pusat Kota

ID	x (Jarak)	y (Harga)	xy	x ²	y ²
1	5	500	2500	25	250000
2	10	450	4500	100	202500
3	2	400	800	4	160000
4	8	350	2800	64	122500
5	15	700	10500	225	490000
6	7	300	2100	49	90000
7	12	380	4560	144	144400
8	6	480	2880	36	230400
9	9	420	3780	81	176400
10	4	320	1280	16	102400
Total	78	4300	35700	744	1968600

3. Interpretasikan setiap hasil korelasi yang didapatkan!

a. Jumlah Kamar

Maka r Harga dengan Jumlah Kamar					
r atas	8200		intepretasi hasil		
r bawah	10317.17015		Harga dengan Luas Tanah memiliki korelasi positif yang sangat erat		
r	0.7947915831		Harga akan naik jika jumlah kamar semakin banyak		

b. Jarak ke Pusat Kota

Maka r Harga dengan Jarak Pusat Kota					
r atas	21600		intepretasi hasil		
r bawah	40271.28009		Harga dengan Jarak ke Pusat Kota memiliki korelasi positif yang sangat lemah		
r	0.5363623891		Jarak ke pusat kota bukan merupakan faktor yang signifikan dalam menentukan harga properti dalam dataset ini		

4. Kumpulkan hasil pengerjaan dengan mengirimkan linknya

https://docs.google.com/spreadsheets/d/1Ji8-YJRzvlxipQmxCosLVy_0knIKZAqleO5mgHxi2ZA/edit?gid=0#gid=0

Implementasi Metode Correlation menggunakan Python

- Buka Web Browser
- Masuk pada Google Colab
- Ketikkan perintah berikut ini

```
import pandas as pd

# Data dalam bentuk dictionary
data = {
    'Luas Tanah (m²)': [100, 150, 80, 120, 200, 90, 130, 110, 140, 95],
    'Jumlah Kamar': [3, 4, 2, 3, 5, 2, 3, 3, 4, 2],
    'Jarak ke Pusat Kota (km)': [5, 10, 2, 8, 15, 7, 12, 6, 9, 4],
    'Harga (Juta Rupiah)': [500, 450, 400, 350, 700, 300, 380, 480, 420, 320]
}

# Membuat DataFrame
df = pd.DataFrame(data)

# Menghitung matriks korelasi
correlation_matrix = df.corr()

# Menampilkan korelasi antara fitur dan target
print(correlation_matrix['Harga (Juta Rupiah)'])
```

```
import pandas as pd

# Data dalam bentuk dictionary
data = {
    'Luas Tanah (m²)': [100, 150, 80, 120, 200, 90, 130, 110, 140, 95],
    'Jumlah Kamar': [3, 4, 2, 3, 5, 2, 3, 3, 4, 2],
    'Jarak ke Pusat Kota (km)': [5, 10, 2, 8, 15, 7, 12, 6, 9, 4],
    'Harga (Juta Rupiah)': [500, 450, 400, 350, 700, 300, 380, 480, 420, 320]
}

# Membuat DataFrame
df = pd.DataFrame(data)

# Menghitung matriks korelasi
correlation_matrix = df.corr()

# Menampilkan korelasi antara fitur dan target
print(correlation_matrix['Harga (Juta Rupiah)'])
```

- Maka hasil yang didapatkan

Luas Tanah (m²)	0.743322	Luas Tanah (m²)	0.743322
Jumlah Kamar	0.794792	Jumlah Kamar	0.794792
Jarak ke Pusat Kota (km)	0.536362	Jarak ke Pusat Kota (km)	0.536362
Harga (Juta Rupiah)	1.000000	Harga (Juta Rupiah)	1.000000
Name: Harga (Juta Rupiah), dtype: float64		Name: Harga (Juta Rupiah), dtype: float64	

- Pada kode berikutnya, ketikkan kode berikut:

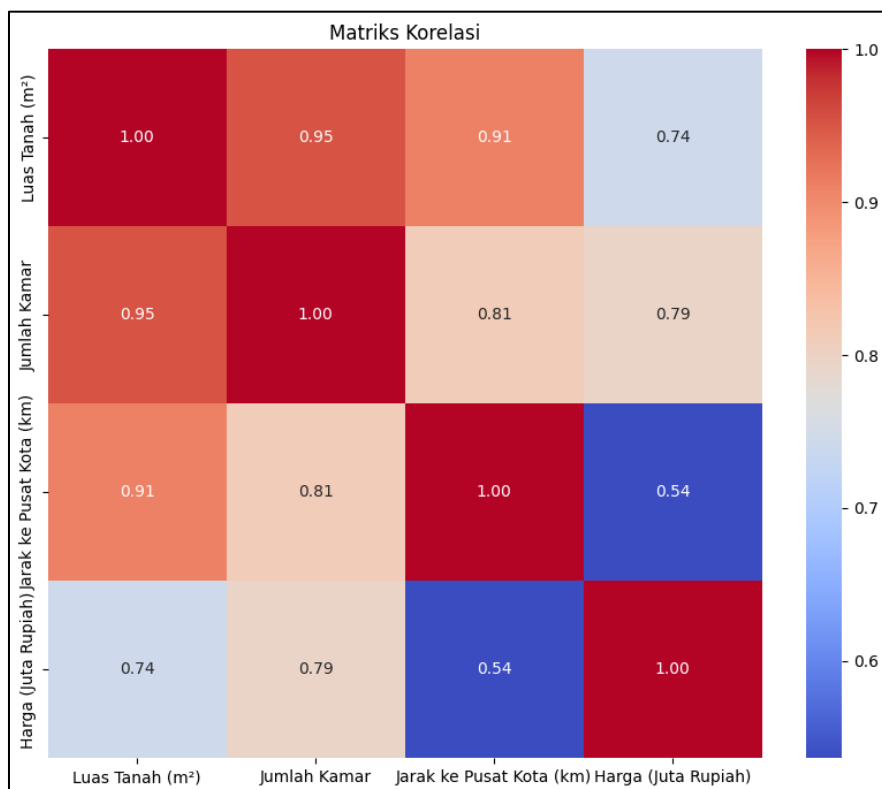
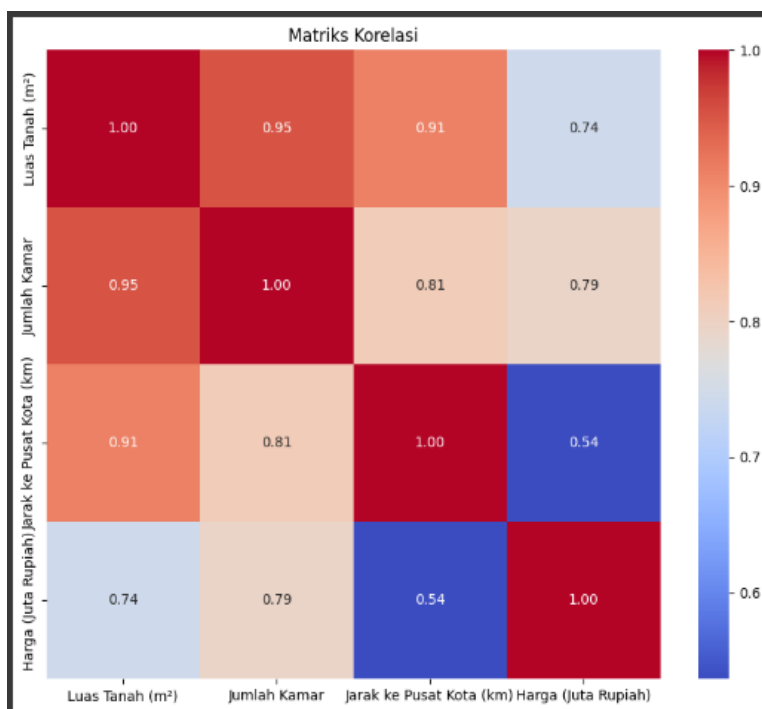
```
import matplotlib.pyplot as plt
import seaborn as sns

# Visualisasi matriks korelasi menggunakan heatmap
plt.figure(figsize=(10, 8))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', fmt=".2f")
plt.title('Matriks Korelasi')
plt.show()
```

```
import matplotlib.pyplot as plt
import seaborn as sns

# Visualisasi matriks korelasi menggunakan heatmap
plt.figure(figsize=(10, 8))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', fmt=".2f")
plt.title('Matriks Korelasi')
plt.show()
```

f. Maka hasil yang didapatkan adalah:



LATIHAN

1. Tambahkan satu kolom pada data menggunakan Python dengan nama “Gangguan Listrik”
2. Buatlah data “Gangguan Listrik” dengan nilai antara 1 sampai dengan 5 yang menunjukkan frekuensi terjadi gangguan listrik
3. Isikan kolom Gangguan Listrik dengan sifat **Korelasi Negatif**
4. Ulangi perhitungan Korelasi dengan kolom yang baru!

Jawaban:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

# Data awal
data = {
    'ID': [1, 2, 3, 4, 5, 6, 7, 8, 9, 10],
    'Luas_Tanah': [100, 150, 80, 120, 200, 90, 130, 110, 140, 95],
    'Jumlah_Kamar': [3, 4, 2, 3, 5, 2, 3, 3, 4, 2],
    'Jarak_ke_Pusat_Kota': [5, 10, 2, 8, 15, 7, 12, 6, 9, 4],
    'Harga': [500, 450, 400, 350, 700, 300, 380, 480, 420, 320]
}

df = pd.DataFrame(data)

# 1. Tambahkan satu kolom pada data menggunakan Python dengan nama "Gangguan Listrik"
# 2. Buatlah data "Gangguan Listrik" dengan nilai antara 1 sampai dengan 5 yang menunjukkan frekuensi terjadi gangguan listrik
# 3. Isikan kolom Gangguan Listrik dengan sifat Korelasi Negatif

# Normalisasi harga ke skala 1-5 dan balik nilainya untuk mendapatkan korelasi negatif
max_harga = df['Harga'].max()
min_harga = df['Harga'].min()

# Rumus untuk menghasilkan nilai 1-5 dengan korelasi negatif terhadap harga
df['Gangguan_Listrik'] = 6 - ((df['Harga'] - min_harga) / (max_harga - min_harga) * 4 + 1).round()

print("Data dengan kolom Gangguan Listrik:")
print(df)

# 4. Ulangi perhitungan Korelasi dengan kolom yang baru!

# Menghitung korelasi baru
new_correlation = df.corr()
print("\nKorelasi setelah menambahkan kolom Gangguan Listrik:")
print(new_correlation)

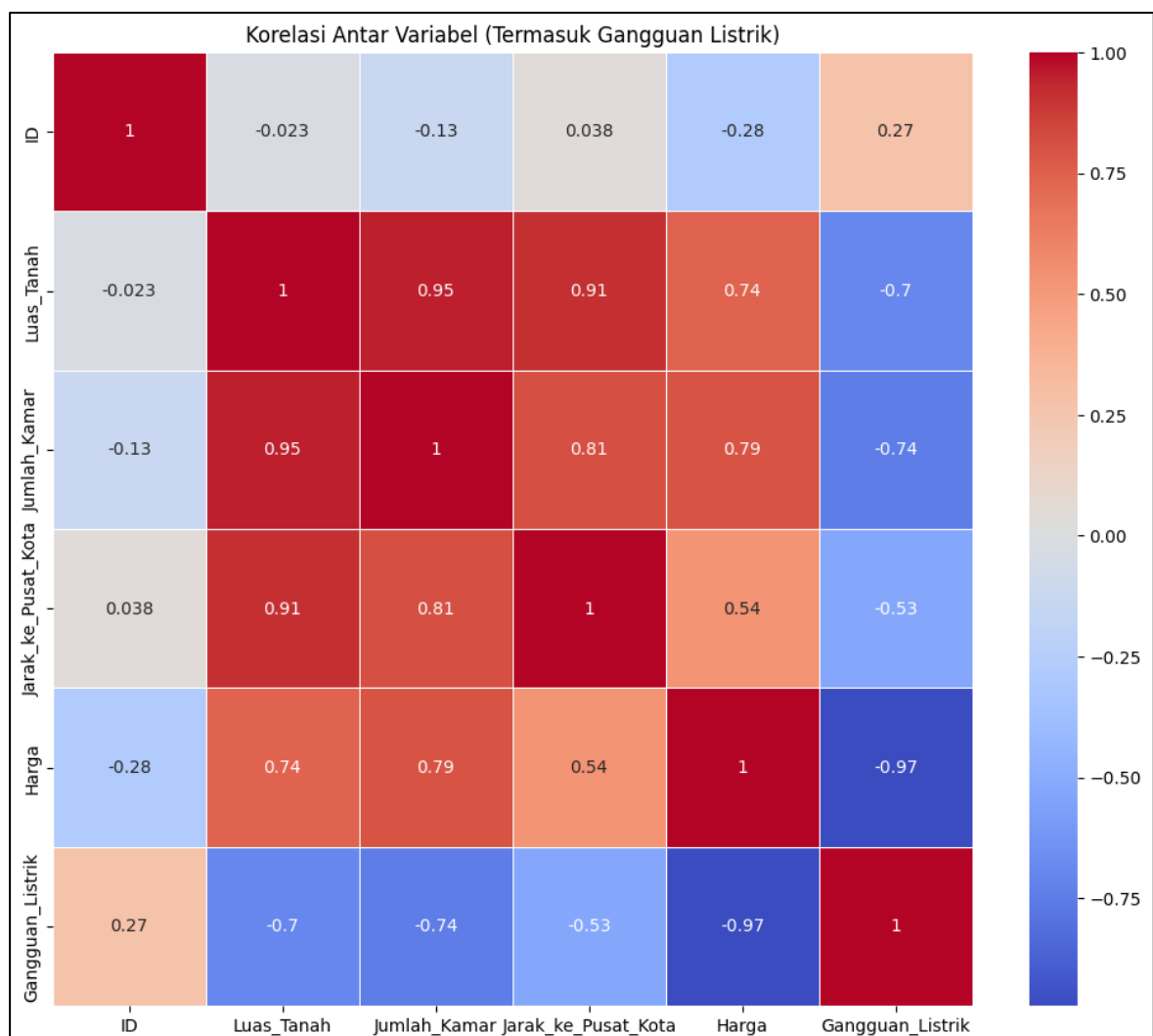
# Visualisasi korelasi baru
plt.figure(figsize=(12, 10))
sns.heatmap(new_correlation, annot=True, cmap='coolwarm', linewidths=0.5)
plt.title('Korelasi Antar Variabel (Termasuk Gangguan Listrik)')
plt.show()
```

Data dengan kolom Gangguan Listrik:

	ID	Luas_Tanah	Jumlah_Kamar	Jarak_ke_Pusat_Kota	Harga	Gangguan_Listrik
0	1	100	3	5	500	3.0
1	2	150	4	10	450	4.0
2	3	80	2	2	400	4.0
3	4	120	3	8	350	4.0
4	5	200	5	15	700	1.0
5	6	90	2	7	300	5.0
6	7	130	3	12	380	4.0
7	8	110	3	6	480	3.0
8	9	140	4	9	420	4.0
9	10	95	2	4	320	5.0

Korelasi setelah menambahkan kolom Gangguan Listrik:

	ID	Luas_Tanah	Jumlah_Kamar	Jarak_ke_Pusat_Kota	Harga	Gangguan_Listrik
ID	1.000000	-0.023198	-0.129165	0.037818	-0.276966	0.269029
Luas_Tanah	-0.023198	1.000000	0.952716	0.911754	0.743322	-0.701318
Jumlah_Kamar	-0.129165	0.952716	1.000000	0.811754	0.794792	-0.741999
Jarak_ke_Pusat_Kota	0.037818	0.911754	0.811754	1.000000	0.536362	-0.533250
Harga	-0.276966	0.743322	0.794792	0.536362	1.000000	-0.972585
Gangguan_Listrik	0.269029	-0.701318	-0.741999	-0.533250	-0.972585	1.000000



Implementasi Metode Sampling Slovin Menggunakan Python

- Buka Google Colabs
- Ketikkan perintah berikut:

```
import pandas as pd
# Parameter Slovin
N = 100 # Ukuran populasi (100)
e = 0.05 # Tingkat kesalahan 5%

# Hitung ukuran sampel
n = slovin_sample_size(N, e)
print(f"Ukuran sampel yang dibutuhkan: {n}")

# Buat DataFrame dummy dengan 100 data
data = {'value': range(1, 101)}
df = pd.DataFrame(data)

# Ambil sampel acak dari DataFrame
sample_df = df.sample(n=n, random_state=42) # random_state untuk hasil yang konsisten

# Tampilkan sampel
print("Sampel yang diambil:")
sample_df
```

```
import pandas as pd

# Fungsi untuk menghitung ukuran sampel menggunakan rumus Slovin
def slovin_sample_size(N, e):
    return int(N / (1 + N * (e ** 2)))

# Parameter Slovin
N = 100 # Ukuran populasi (100)
e = 0.05 # Tingkat kesalahan 5%

# Hitung ukuran sampel
n = slovin_sample_size(N, e)
print(f"Ukuran sampel yang dibutuhkan: {n}")

# Buat DataFrame dummy dengan 100 data
data = {'value': range(1, 101)}
df = pd.DataFrame(data)

# Ambil sampel acak dari DataFrame
sample_df = df.sample(n=n, random_state=42) # random_state untuk hasil yang konsisten

# Tampilkan sampel
print("Sampel yang diambil:")
sample_df
```

- Maka akan menghasilkan sebagai berikut:

Ukuran sampel yang dibutuhkan: 80
Sampel yang diambil:

value	
83	84
53	54
70	71
45	46
44	45
...	...
57	58
75	76
32	33
94	95
59	60

80 rows x 1 columns

Ukuran sampel yang dibutuhkan: 80
Sampel yang diambil:

value	
83	84
53	54
70	71
45	46
44	45
...	...
57	58
75	76
32	33
94	95
59	60

80 rows x 1 columns

LATIHAN

1. Buatlah data dengan menggunakan 5 kolom dan 1000 baris

```
import pandas as pd
import numpy as np

# Membuat DataFrame dengan 5 kolom dan 1000 baris
np.random.seed(42) # Untuk hasil yang konsisten
data = {
    'ID': range(1, 1001), # ID unik untuk setiap baris
    'Usia': np.random.randint(18, 60, 1000), # Usia antara 18 - 60 tahun
    'Pendapatan': np.random.randint(3000000, 15000000, 1000), # Pendapatan dalam rupiah
    'Jumlah_Anggota_Keluarga': np.random.randint(1, 6, 1000), # Anggota keluarga antara 1 - 5
    'Pengeluaran_Bulanan': np.random.randint(1000000, 10000000, 1000) # Pengeluaran bulanan
}

df = pd.DataFrame(data)

# Menampilkan 5 baris pertama dari dataset
print(df.head())
```

	ID	Usia	Pendapatan	Jumlah_Anggota_Keluarga	Pengeluaran_Bulanan
0	1	56	11514716	5	1312023
1	2	46	14548718	5	7270016
2	3	32	14618627	1	8320583
3	4	25	13096300	2	8822636
4	5	38	8525633	4	4177902

2. Lakukan metode Sampling Slovin dengan menggunakan Python

```
# Fungsi untuk menghitung ukuran sampel menggunakan Rumus Slovin
def slovin_sample_size(N, e):
    return int(N / (1 + N * (e ** 2)))

# Menentukan ukuran sampel dengan tingkat kesalahan 5%
N = len(df) # Jumlah populasi (1000)
e = 0.05 # Tingkat kesalahan 5%
n = slovin_sample_size(N, e)

print(f"Ukuran sampel yang dibutuhkan: {n}")

# Mengambil sampel acak dari DataFrame
sample_df = df.sample(n=n, random_state=42) # random_state agar hasil tetap sama

# Menampilkan 5 baris pertama dari sampel
print(sample_df.head())
```

ID	Usia	Pendapatan	Jumlah_Anggota_Keluarga	Pengeluaran_Bulanan	
521	522	45	11197295	2	4948401
737	738	34	14779534	5	9725977
740	741	22	12734816	4	1296063
660	661	39	14254507	5	5151923
411	412	43	10904294	4	1291464

3. Jelaskan masing-masing baris perintah Python yang tersusun!

1. Membuat Data:

- `np.random.seed(42)`: Menetapkan seed untuk menghasilkan angka acak yang sama setiap kali kode dijalankan.
- `data = {...}`: Membuat dictionary dengan 5 kolom yang berisi data acak.
- `df = pd.DataFrame(data)`: Mengubah dictionary menjadi DataFrame Pandas.
- `print(df.head())`: Menampilkan 5 baris pertama dari dataset.

2. Menghitung Ukuran Sampel Menggunakan Rumus Slovin:

- `def slovin_sample_size(N, e)`: Fungsi untuk menghitung ukuran sampel berdasarkan Rumus Slovin.
- `N = len(df)`: Menghitung jumlah populasi dari DataFrame.
- `e = 0.05`: Menentukan tingkat kesalahan sebesar 5%.
- `n = slovin_sample_size(N, e)`: Menghitung ukuran sampel yang dibutuhkan.
- `print(f"Ukuran sampel yang dibutuhkan: {n}")`: Menampilkan jumlah sampel yang diperlukan.

3. Mengambil Sampel:

- `df.sample(n=n, random_state=42)`: Mengambil sampel acak sebanyak n dari DataFrame.
- `print(sample_df.head())`: Menampilkan 5 baris pertama dari data sampel.