

## **Machine Learning**



**OLEH:**

**NAMA : Fathiyaturohmah**

**NIM : 0110224060**

**E-mail : 0110224060@student.nurulfikri.ac.id**

**Sekolah Tinggi Teknologi Terpadu Nurul Fikri**  
**Program Studi Teknik Informatika**  
**2025**

## 1. Hasil Praktikum Kelas

Dataset: 500\_Person\_Gender\_Height\_Weight\_Index

### a. Membaca dataset

Kode ini membaca file CSV menjadi DataFrame df dan menampilkan 5 baris pertama sebagai gambaran data.

```
[6]
✓ 2 d # membaca file csv menggunakan pandas
import pandas as pd

df = pd.read_csv( '/content/gdrive/MyDrive/Praktikum2/Data/500_Person_Gender_Height_Weight_Index.csv')
df
```

	Gender	Height	Weight	Index
0	Male	174	96	4
1	Male	189	87	2
2	Female	185	110	4
3	Female	195	104	3
4	Male	149	61	3
...	...	...	...	...
495	Female	150	153	5

### b. Informasi Dataset

Menampilkan jumlah baris, kolom, tipe data, dan jumlah nilai non-null.

```
# Mencari info data pada file (tipe datanya, non nul count data, nama kolom)
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 500 entries, 0 to 499
Data columns (total 4 columns):
 #   Column  Non-Null Count  Dtype  
---  -
 0   Gender  500 non-null    object  
 1   Height  500 non-null    int64   
 2   Weight  500 non-null    int64   
 3   Index   500 non-null    int64   
dtypes: int64(3), object(1)
memory usage: 15.8+ KB
```

### c. Statistik Deskriptif

Menghitung Nilai nilai sentral (mean, median, modus)

```
# Menghitung mean semua kolom numerik
df['Height'].mean()
```

```
np.float64(169.944)
```

```
# Menghitung median semua kolom numerik
df['Height'].median()
```

```
170.5
```

```
# Mencari modus (hati-hati karena bisa lebih dari satu)
df['Height'].mode()
```

```
Height
0      188
dtype: int64
```

```
# Menghitung Variansi & Standard Deviasi
df.var(numeric_only=True)
```



0

Height	268.149162
Weight	1048.633267
Index	1.836168

dtype: float64

```
# Menghitung Standard Deviasi
df.std(numeric_only=True)
```



0

Height	16.375261
Weight	32.382607
Index	1.355053

## Menghitung Kuartil



```
# Hitung kuartil pertama (Q1)
q1 = df['Height'].quantile(0.25)
print("Q1 : ", q1)

# Hitung kuartil ketiga (Q3)
q3 = df['Height'].quantile(0.75)
print("Q3 : ", q3)

# Hitung IQR (Interquartile Range)
iqr = q3 - q1
print("IQR : ", iqr)
```



```
Q1 : 156.0
Q3 : 184.0
IQR : 28.0
```

## Menghitung statistik Deskriptif Otomatis

```
# Untuk membuat statistika deskripsi pada type data int
df.describe()
```



	Height	Weight	Index
count	500.000000	500.000000	500.000000
mean	169.944000	106.000000	3.748000
std	16.375261	32.382607	1.355053
min	140.000000	50.000000	0.000000
25%	156.000000	80.000000	3.000000
50%	170.500000	106.000000	4.000000
75%	184.000000	136.000000	5.000000
max	199.000000	160.000000	5.000000



## Menghitung Korelasi

```
# Menghitung matriks korelasi untuk semua kolom numerik
correlation_matrix = df.corr(numeric_only=True)

# Menampilkan matriks korelasi
print("Matriks Korelasi:")
print(correlation_matrix)
```

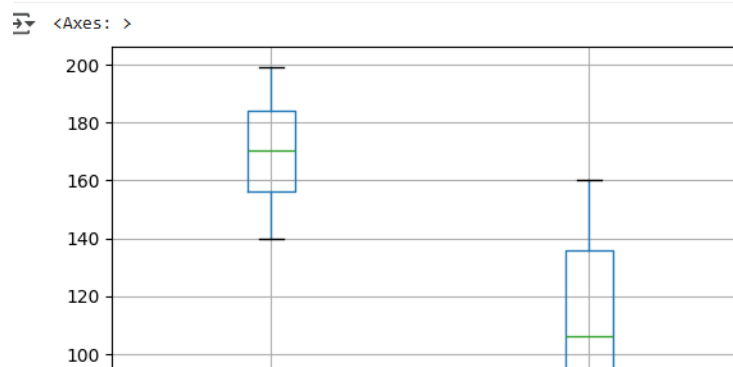
```
⇒ Matriks Korelasi:
      Height  Weight  Index
Height  1.000000  0.000446 -0.422223
Weight  0.000446  1.000000  0.804569
Index   -0.422223  0.804569  1.000000
```

### d. Visualisasi Data

- Boxplot: menunjukkan distribusi dan outlier
- Histogram: distribusi frekuensi data
- Scatter plot: hubungan antar variabel

```
import pandas as pd
import numpy as np

df.boxplot(column=['Height', 'Weight'])
```



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

# Ambil data Height
data_height = df["Height"]

# Buat histogram
n, bins, patches = plt.hist(data_height, bins=5, color='pink', edgecolor='black')

# Tambahkan label
plt.title('Histogram Nilai')
plt.xlabel('Height')
plt.ylabel('Frekuensi')

# Tampilkan rentang frekuensi di sumbu x
bin_centers = 0.5 * (bins[:-1] + bins[1:])
plt.xticks(bin_centers, ['{:.0f}-{:.0f}'.format(bins[i], bins[i+1]) for i in range(len(bins)-1)])

# Tampilkan histogram
plt.show()
```

```

df2 = pd.DataFrame(data)

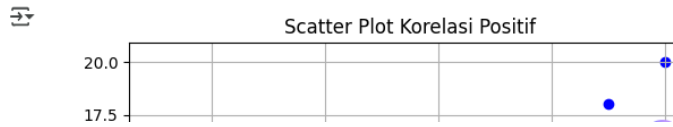
# Buat scatter plot
plt.scatter(df2['Nilai1'], df2['Nilai2'], color='blue', marker='o')

# Tambahkan label
plt.title('Scatter Plot Korelasi Positif')
plt.xlabel('Nilai1')
plt.ylabel('Nilai2')

# Tambahkan grid
plt.grid(True)

# Tampilkan plot
plt.show()

```



```

}

df3 = pd.DataFrame(data)

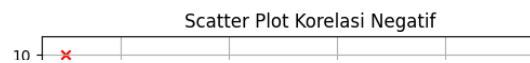
# Buat scatter plot
plt.scatter(df3['Nilai1'], df3['Nilai2'], color='red', marker='x')

# Tambahkan label
plt.title('Scatter Plot Korelasi Negatif')
plt.xlabel('Nilai1')
plt.ylabel('Nilai2')

# Tambahkan grid
plt.grid(True)

# Tampilkan plot
plt.show()

```



## 2. Tugas Praktikum Mandiri

Dataset: day.csv

### a. Membaca dataset

Membaca dataset day.csv untuk dianalisis.

```
import pandas as pd

path = "/content/drive/MyDrive/Pekan2 ML/Tugas Praktikum Mandiri/data/day.csv"
df = pd.read_csv(path)
df.head()
```

	instant	dteday	season	yr	mnth	holiday	weekday	workingday	weathersit	temp	atemp	hum	windspeed	casual	registered
0	1	2011-01-01	1	0	1	0	6	0	2	0.344167	0.363625	0.805833	0.169915	73	53
1	2	2011-01-02	1	0	1	0	0	0	2	0.363478	0.353739	0.696087	0.169915	73	53
2	3	2011-01-03	1	0	1	0	1	1	1	0.196364	0.189405	0.437273	0.169915	73	53
3	4	2011-01-04	1	0	1	0	2	1	1	0.200000	0.212122	0.590435	0.169915	73	53
4	5	2011-01-05	1	0	1	0	3	1	1	0.226957	0.229270	0.436957	0.169915	73	53

### b. Membagi dataset

- Train: 80% data untuk melatih model
- Validation: 10% dari train untuk evaluasi saat training
- Test: 20% untuk uji performa model

```
from sklearn.model_selection import train_test_split

# Split pertama: Train 80% - Test 20%
train, test = train_test_split(df, test_size=0.2, random_state=42)

# Split kedua: dari Train -> Validation 10%
train, val = train_test_split(train, test_size=0.1, random_state=42)

# Jumlah data tiap set
print("Jumlah Data Training:", len(train))
print("Jumlah Data Validation:", len(val))
print("Jumlah Data Testing:", len(test))

# 5 baris pertama
print("\nTraining Data:\n", train.head())
print("\nValidation Data:\n", val.head())
print("\nTesting Data:\n", test.head())
```

Jumlah Data Training: 525  
Jumlah Data Validation: 59  
Jumlah Data Testing: 147

Training Data:

instant	dteday	season	yr	mnth	holiday	weekday	workingday	weathersit	temp	atemp	hum	windspeed	casual	registered
657	2012-10-19	4	1	10	0	5	1	2	0.563333	0.537896	0.815000	0.134954	753	4671
163	2011-06-13	2	0	6	0	1	1	1	0.635000	0.601654	0.494583	0.305350	863	4157
305	2011-11-02	4	0	11	0	3	1	1	0.377500	0.390133	0.718750	0.082092	370	3816
111	2011-04-22	2	0	4	0	5	1	1	0.336667	0.321954	0.729583	0.219521	177	1506
538	2012-06-22	3	1	6	0	5	1	1	0.777500	0.724121	0.573750	0.182842	964	4859

instant	cnt
657	5424
163	5020
305	4186
111	1683
538	5823

```

Validation Data:
  instant  dteday  season  yr  mnth  holiday  weekday  workingday \
325    326 2011-11-22     4   0    11         0         2         1
410    411 2012-02-15     1   1     2         0         3         1
92     93 2011-04-03     2   0     4         0         0         0
47     48 2011-02-17     1   0     2         0         4         1
508    509 2012-05-23     2   1     5         0         3         1

  weathersit  temp  atemp  hum  windspeed  casual  registered
325         3  0.416667  0.421696  0.962500  0.118792      69      1538
410         1  0.348333  0.351629  0.531250  0.181600     141     4028
92          1  0.378333  0.378767  0.480000  0.182213    1651     1598
47          1  0.435833  0.428658  0.505000  0.230104     259     2216
508         2  0.621667  0.584612  0.774583  0.102000     766     4494

      cnt
325  1607
410  4169
92   3249
47   2475
508  5260

```

### 3. Kesimpulan

Dari praktikum ini, saya belajar bahwa statistik deskriptif membantu memahami data melalui angka-angka penting seperti rata-rata, median, dll. Visualisasi seperti boxplot, histogram, dan scatter plot memudahkan saya melihat pola dan hubungan antar variabel. Sedangkan pembagian dataset menjadi training, validation, dan testing penting agar model Machine Learning bisa dilatih dengan baik dan diuji secara adil.

### 4. Link Github

[https://github.com/fathiyaturohmah/Fathiyaturohmah\\_0110224060\\_LaporanPraktikumMandiri2.git](https://github.com/fathiyaturohmah/Fathiyaturohmah_0110224060_LaporanPraktikumMandiri2.git)