Tugas 1: Laporan Praktikum dan Tugas Mandiri

Aan Adriyana - 0110224014

[1] Teknik Informatika, STT Terpadu Nurul Fikri, Depok

*E-mail: 0110224014@student.nurulfikri.ac.id

Abstract. Laporan ini berisi tentang penjelasan untuk menganalisis statistik deskriptif seperti: melihat informasi umum data, menghitung nilai-nilai sentral (mean, median, modus), menghitung ukuran persebaran (variasi & standar deviasi), menghitung kuartil, menghitung statistik deskriptif otomatis dan menghitung korelasi. Kemudian, membuat visualisasi data seperti: boxplot, histogram dan scatter plot.

1. Analisis Statistik Deskriptif

1.1 Melihat Informasi Umum Data

Mengetahui informasi mengenai dataset denngan menerapkan method .info()

```
# Mencari info data pada file (tipe datanya, non null 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
```

1.2 Menghitung Nilai-Nilai Sentral (Mean, Median, Modus)

Menghitun nilai-nilai sentral dari kolom data Height (Tinggi) menggunakan Pustaka pandas di Python.

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
```

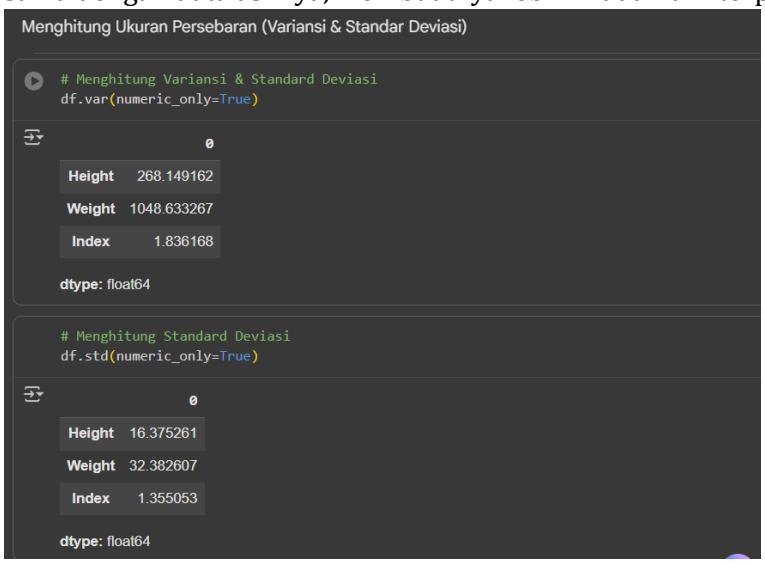
```
# Mencari modus (hati-hati karena bisa lebih dari satu)
df['Height'].mode()
```

	Height
0	188

dtype: int64

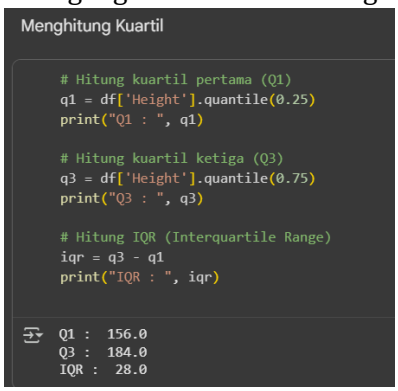
1.3 Menghitung ukuran persebaran (variasi & standar deviasi)

Variansi mengukur seberapa jauh setiap titik data tersebar dari nilai rata-ratanya. Nilai variansi yang besar menunjukkan data yang sangat tersebar. Standar Deviasi adalah akar kuadrat dari variansi. Ini adalah ukuran penyebaran yang paling umum karena nilainya memiliki satuan yang sama dengan data aslinya, membuatnya lebih mudah diinterpretasikan daripada variansi.



1.4 Menghitung Kuartil

Tujuan utama dari ketiga metrik ini adalah untuk memahami penyebaran (dispersion) data dan sering digunakan untuk mengidentifikasi *outlier* (data pencilan) dalam analisis statistik.



1.5 Menghitung Statistik Deskriptif Otomatis

Metrik-metrik ini memberikan ringkasan statistik yang cepat mengenai pusat, penyebaran, dan bentuk (min, max, Q1, Q2, Q3) dari distribusi data numerik.

Menghitung Statistik Deskriptif Otomatis

```
# Untuk membuat statistika deskripsi pada tipe 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

1.6 Menghitung Korelasi

Melakukan Analisis Statistik Deskriptif data secara komprehensif, mencakup pemeriksaan kualitas, pusat, dan penyebaran data, serta mengukur hubungan linier antar variabel. Hasilnya memberikan kesimpulan cepat tentang struktur dan kesehatan *dataset* untuk menginformasikan tahap pemodelan atau analisis data yang lebih mendalam.

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
```

2. Visualisasi Data

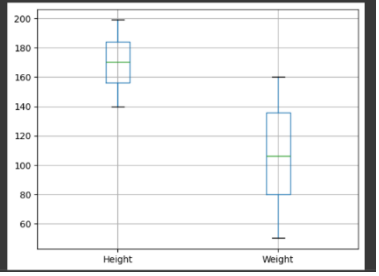
2.1 Boxplot

Visualisasi data yang menyajikan ringkasan lima angka statistik (minimum, Kuartil Pertama/Q1, Median/Q2, Kuartil Ketiga/Q3, dan maksimum) dari suatu kumpulan data untuk menunjukkan sebaran, kemiringan, dan adanya pencilan (outlier).

Boxplot

```
import pandas as pd
import numpy as np

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

```
<Axes: >
```

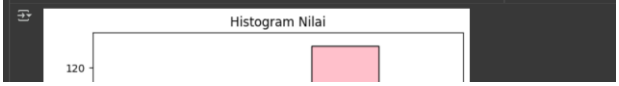


2.2 Histogram

Grafik yang memvisualisasikan distribusi data numerik dengan membaginya menjadi beberapa interval (bin) dan menghitung frekuensi (jumlah) data dalam setiap bin tersebut.

Histogram

```
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()
```

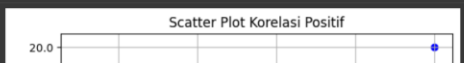


2.3 Scatter Plot

Grafik yang menampilkan titik-titik data menggunakan koordinat Kartesius untuk menunjukkan hubungan antara dua variabel numerik.

Scatter Plot

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

# Buat DataFrame contoh
data = {
    'Nilai1': [1, 2, 3, 4, 5, 6, 7, 8, 9, 10],
    'Nilai2': [2, 4, 6, 8, 10, 12, 14, 16, 18, 20]
}
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()
```




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

# Buat DataFrame contoh
data = {
    'Nilai1': [1, 2, 3, 4, 5, 6, 7, 8, 9, 10],
    'Nilai2': [10, 9, 8, 7, 6, 5, 4, 3, 2, 1]
}

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()
```

3. Tugas Mandiri

Membagi dataset day.csv menjadi tiga bagian yaitu training, validation, dan testing dengan proporsi 80%, 10% dan 20%. Setelah pembagian, menampilkan jumlah data serta 5 baris pertama dari masing-masing set sebagai bukti hasil pemrosesan.

```
import pandas as pd
from sklearn.model_selection import train_test_split

# Define the path to the data
path = "/content/drive/MyDrive/praktikum_ml/praktikum02"

# Baca dataset
df = pd.read_csv(path + '/data/day.csv')

# Split data menjadi training (80%) dan testing (20%)
train_df, test_df = train_test_split(df, test_size=0.2, random_state=42)

# Dari training, ambil 10% sebagai validation
train_df, val_df = train_test_split(train_df, test_size=0.1, random_state=42)

# Tampilkan jumlah data
print("Jumlah data total:", len(df))
print("Jumlah data training:", len(train_df))
print("Jumlah data validation:", len(val_df))
print("Jumlah data testing:", len(test_df))

print("\n5 data teratas Training:")
print(train_df.head())

print("\n5 data teratas Validation:")
print(val_df.head())

print("\n5 data teratas Testing:")
print(test_df.head())
```

```
Jumlah data total: 731
Jumlah data training: 525
Jumlah data validation: 59
Jumlah data testing: 147

5 data teratas Training:
     instant      dteday  season  yr  mnth  holiday  weekday  workingday  \
657      658  2012-10-19       4   1    10        0        5           1
163      164  2011-06-13       2   0     6        0        1           1
305      306  2011-11-02       4   0    11        0        3           1
111      112  2011-04-22       2   0     4        0        5           1
538      539  2012-06-22       3   1     6        0        5           1

     weathersit      temp     atemp       hum  windspeed  casual  registered  \
657           2  0.563333  0.537896  0.815000   0.134954     753        4671
163           1  0.635000  0.601654  0.494583   0.305350     863        4157
305           1  0.377500  0.390133  0.718750   0.082092     370        3816
111           2  0.336667  0.321954  0.729583   0.219521     177        1506
538           1  0.777500  0.724121  0.573750   0.182842     964        4859

      cnt
657  5424
```

Link Colab:

https://colab.research.google.com/drive/1Qr7rjycgdD0L3kKadAiavsVOL3pibE_a?usp=sharing

Referensi:

https://medium.com/@mirdhasucianandа/data-cleaning-menggunakan-python-154ee86ca8afr.
https://www.revou.co/kosakata/box-plot
https://pythongeeks.org/python-
histogram/#:~:text=Ringkasan,hingga%20tingkat%20lanjut%20sesuai%20kebutuhan.
https://sis.binus.ac.id/2019/10/29/dasar-scatter-plot-pada-tibco-spotfire-x/