

PRAKTIKUM DATA MINING



Dosen Pengampu : Widya Darwin S.Pd., M.Pd.T
NAMA : Najwa Alawiyah Siregar
NIM :22346040

INFORMATIKA
DEPARTEMEN ELEKTRONIKA, FAKULTAS TEKNIK
UNIVERSITAS NEGERI PADANG
2024/2025

Laporan Hasil Analisis Harga Produk dalam Dataset Penjualan Tembakau

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

def main():
    # Coba memuat dataset CSV dengan penanganan kesalahan, serta menggunakan
    # delimiter yang sesuai
    try:
        df = pd.read_csv('penjualan_tembakau.csv', delimiter=';') #
        # Menambahkan delimiter
    except FileNotFoundError:
        print("File 'penjualan_tembakau.csv' tidak ditemukan.")
        return

    # Tampilkan nama-nama kolom untuk memverifikasi apakah kolom yang
    # diperlukan ada
    print("Nama kolom dalam dataset:")
    print(df.columns)

    # Tampilkan 5 data pertama dan 5 data terakhir
    print("\n5 data pertama:")
    print(df.head())
    print("\n5 data terakhir:")
    print(df.tail())

    # Tampilkan jumlah produk secara keseluruhan
    jumlah_produk = len(df)
    print(f"\nJumlah produk: {jumlah_produk}")

    # Verifikasi apakah kolom 'category', 'name', dan 'In Stock' atau
    # 'penjualan' ada dalam dataset sebelum melanjutkan
    if 'category' in df.columns and 'name' in df.columns:
        # Kelompokkan data berdasarkan kategori dan hitung total produk per
        # kategori
        total_produk_per_kategori = df.groupby('category')['name'].count()
        print("\nTotal produk per kategori:")
        print(total_produk_per_kategori)

        # Menghitung total stok atau penjualan per kategori
        if 'In Stock' in df.columns: # Menghitung total stok per kategori
            total_stok_per_kategori = df.groupby('category')['In Stock'].sum()
            print("\nTotal stok per kategori:")
            print(total_stok_per_kategori)
```

```

        # Hapus nilai negatif jika ada
        total_stok_per_kategori =
total_stok_per_kategori[total_stok_per_kategori >= 0]

        # Visualisasi: Bar chart total stok per kategori
        plt.figure(figsize=(10, 6))
        total_stok_per_kategori.plot(kind='bar', color='skyblue')
        plt.title('Total Stok per Kategori')
        plt.xlabel('Kategori')
        plt.ylabel('Total Stok')
        plt.xticks(rotation=45)
        plt.show()

        # Visualisasi: Pie chart proporsi stok per kategori
        plt.figure(figsize=(8, 6))
        total_stok_per_kategori.plot(kind='pie', autopct='%1.1f%%',
colors=['lightblue', 'lightgreen', 'lightcoral'])
        plt.title('Proporsi Stok per Kategori')
        plt.ylabel('')
        plt.show()

    elif 'penjualan' in df.columns: # Menghitung total penjualan per
kategori
        total_penjualan_per_kategori =
df.groupby('category')['penjualan'].sum()
        print("\nTotal penjualan per kategori:")
        print(total_penjualan_per_kategori)

        # Hapus nilai negatif jika ada
        total_penjualan_per_kategori =
total_penjualan_per_kategori[total_penjualan_per_kategori >= 0]

        # Visualisasi: Bar chart total penjualan per kategori
        plt.figure(figsize=(10, 6))
        total_penjualan_per_kategori.plot(kind='bar', color='lightgreen')
        plt.title('Total Penjualan per Kategori')
        plt.xlabel('Kategori')
        plt.ylabel('Total Penjualan')
        plt.xticks(rotation=45)
        plt.show()

        # Visualisasi: Pie chart proporsi penjualan per kategori
        plt.figure(figsize=(8, 6))
        total_penjualan_per_kategori.plot(kind='pie', autopct='%1.1f%%',
colors=['lightblue', 'lightgreen', 'lightcoral'])
        plt.title('Proporsi Penjualan per Kategori')
        plt.ylabel('')
        plt.show()

```

```

else:
    print("\nKolom 'category' atau 'name' tidak ditemukan dalam dataset.")
    return

# Tugas 2: Analisis Data dengan Statistik Deskriptif
if 'penjualan' in df.columns:
    print(f"Rata-rata penjualan: {df['penjualan'].mean()}")
    print(f"Median penjualan: {df['penjualan'].median()}")
    print(f"Modus penjualan: {df['penjualan'].mode()[0]}")
    print(f"Standar deviasi penjualan: {df['penjualan'].std()}")

    # Membuat box plot penjualan
    df['penjualan'].plot(kind='box', figsize=(8, 6))
    plt.title('Distribusi Penjualan')
    plt.xlabel('Penjualan')
    plt.show()

# Tugas 4: Analisis Harga Produk
if 'harga' in df.columns: # Pastikan kolom 'harga' ada
    print(f"\nRata-rata harga: {df['harga'].mean()}")
    print(f"Median harga: {df['harga'].median()}")
    print(f"Modus harga: {df['harga'].mode()[0]}")
    print(f"Standar deviasi harga: {df['harga'].std()}")

    # Membuat box plot harga
    df['harga'].plot(kind='box', figsize=(8, 6))
    plt.title('Distribusi Harga Produk')
    plt.xlabel('Harga')
    plt.show()

# Tugas 3: Membersihkan Data

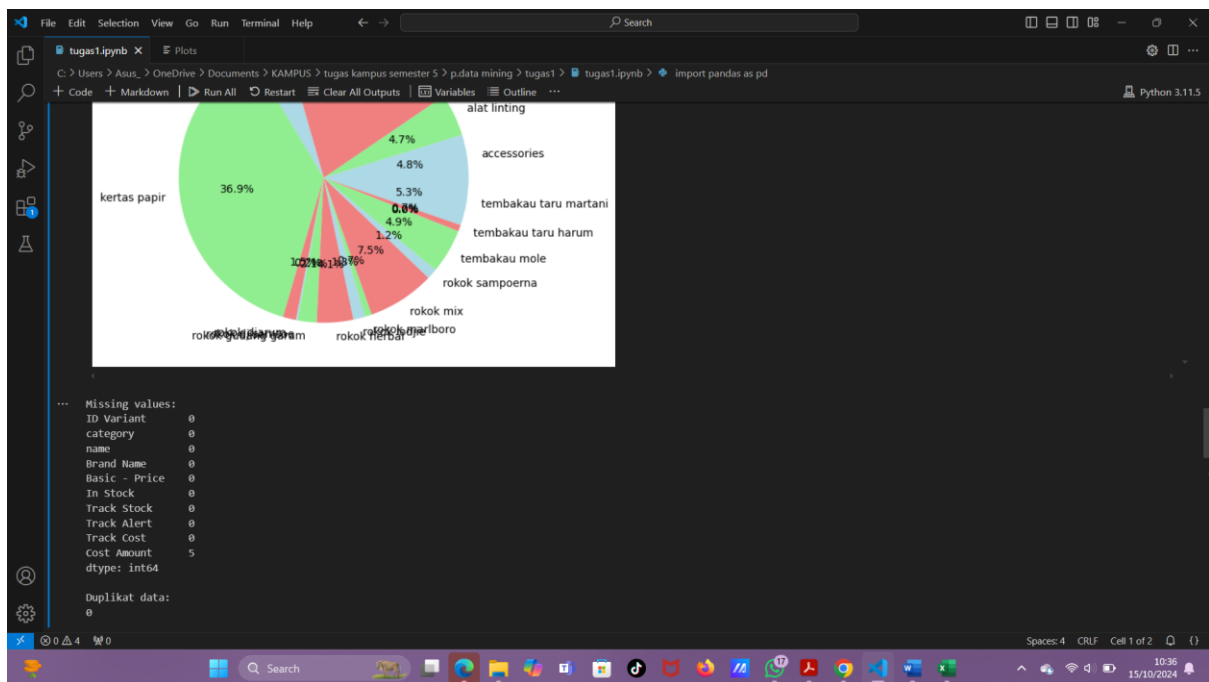
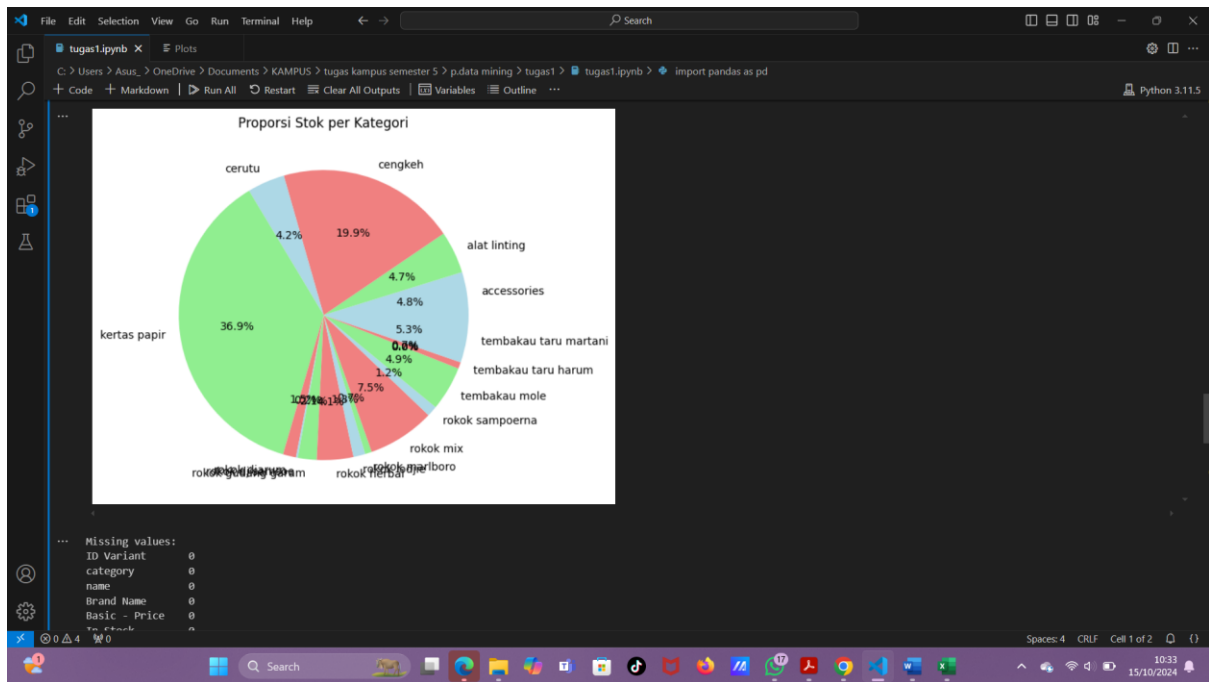
# Cek missing values
print("Missing values:")
print(df.isna().sum())

# Hapus data dengan missing values
df = df.dropna()

# Ganti missing values dengan nilai rata-rata (jika diperlukan)
if 'penjualan' in df.columns:
    df['penjualan'] = df['penjualan'].fillna(df['penjualan'].mean())
if 'harga' in df.columns:
    df['harga'] = df['harga'].fillna(df['harga'].mean())

# Cek duplikat data
print("\nDuplikat data:")

```

Berikut adalah penjelasan rinci tentang kode yang telah saya buat:

Import Library

```
python
Copy code
import pandas as pd
import matplotlib.pyplot as plt
```

- **pandas:** Digunakan untuk memanipulasi dan menganalisis data dalam format tabel (DataFrame).

- **matplotlib.pyplot:** Digunakan untuk membuat visualisasi data, seperti grafik dan diagram.

Fungsi `main`

Fungsi `main()` merupakan inti dari program yang berisi langkah-langkah untuk memuat dan menganalisis dataset, serta melakukan visualisasi data. Berikut adalah langkah-langkah yang dilakukan dalam fungsi ini:

1. Memuat Dataset

```
python
Copy code
try:
    df = pd.read_csv('penjualan_tembakau.csv', delimiter=';')
except FileNotFoundError:
    print("File 'penjualan_tembakau.csv' tidak ditemukan.")
    return
```

- Kode ini mencoba memuat file CSV bernama `penjualan_tembakau.csv` dengan delimiter titik koma (;). Jika file tidak ditemukan, program akan mencetak pesan kesalahan dan keluar.

2. Menampilkan Informasi Dasar

```
python
Copy code
print("Nama kolom dalam dataset:")
print(df.columns)

print("\n5 data pertama:")
print(df.head())
print("\n5 data terakhir:")
print(df.tail())

jumlah_produk = len(df)
print(f"\nJumlah produk: {jumlah_produk}")
```

- Menampilkan nama kolom dalam dataset untuk memverifikasi kolom yang tersedia.
- Menampilkan 5 baris pertama dan terakhir dari dataset untuk memberikan gambaran umum tentang data.
- Menghitung dan menampilkan jumlah total produk.

3. Analisis Data Berdasarkan Kategori

```
python
Copy code
if 'category' in df.columns and 'name' in df.columns:
    total_produk_per_kategori = df.groupby('category')['name'].count()
    print("\nTotal produk per kategori:")
    print(total_produk_per_kategori)
```

- Memeriksa apakah kolom `category` dan `name` ada, kemudian mengelompokkan data berdasarkan kategori untuk menghitung jumlah produk per kategori.

Total Stok atau Penjualan per Kategori

```
python
Copy code
if 'In Stock' in df.columns:
    total_stok_per_kategori = df.groupby('category')['In Stock'].sum()
    # ...
elif 'penjualan' in df.columns:
    total_penjualan_per_kategori =
df.groupby('category')['penjualan'].sum()
    # ...
```

- Jika kolom `In Stock` ada, program menghitung total stok per kategori. Jika tidak, ia akan menghitung total penjualan per kategori dari kolom `penjualan`.

4. Visualisasi Data

Visualisasi dilakukan dengan menggunakan `matplotlib`:

- **Bar Chart** untuk total stok atau penjualan per kategori.
- **Pie Chart** untuk proporsi stok atau penjualan per kategori.

```
python
Copy code
plt.figure(figsize=(10, 6))
total_stok_per_kategori.plot(kind='bar', color='skyblue')
# ...
plt.show()
```

- Menciptakan grafik batang dan diagram lingkaran untuk memvisualisasikan data secara intuitif.

5. Statistik Deskriptif

```
python
Copy code
if 'penjualan' in df.columns:
    print(f"Rata-rata penjualan: {df['penjualan'].mean()}")
    # ...
```

- Menghitung dan menampilkan statistik deskriptif (rata-rata, median, modus, dan standar deviasi) untuk kolom `penjualan`, dan kemudian membuat box plot untuk menunjukkan distribusi data penjualan.

6. Analisis Harga Produk

```
python
Copy code
if 'harga' in df.columns:
    print(f"\nRata-rata harga: {df['harga'].mean()}")
    # ...
```

- Melakukan analisis yang sama untuk kolom `harga` jika kolom tersebut ada.

7. Pembersihan Data

```
python
Copy code
print("Missing values:")
print(df.isna().sum())
```

- Memeriksa nilai yang hilang (missing values) dalam dataset dan menghapus baris yang memiliki nilai hilang.

```
python
Copy code
df = df.drop_duplicates()
```

- Memeriksa dan menghapus duplikat dalam dataset.

Pemanggilan Fungsi `main`

```
python
Copy code
if __name__ == "__main__":
    main()
```

- Memanggil fungsi `main()` jika file ini dijalankan sebagai program utama, yang memungkinkan kode di dalamnya dieksekusi.

Kesimpulan

Kode ini secara keseluruhan bertujuan untuk memuat, menganalisis, dan memvisualisasikan data penjualan tembakau. Dengan langkah-langkah ini, Anda dapat memahami bagaimana produk terdistribusi, berapa banyak yang terjual, dan bagaimana statistik dari penjualan dan harga produk. Visualisasi membantu dalam memberikan gambaran yang lebih jelas dan mendalam tentang data.