

Analisis Data Penjualan Menggunakan Python

By Eka Rahmi Syamsuddin



Project ini bertujuan untuk menerapkan pengetahuan yang diperoleh dari kelas "Python for Data Analyst" dalam menganalisis dataset penjualan dari Kaggle. Melalui proyek ini, akan dilakukan pembersihan data, manipulasi data, dan pembuatan visualisasi untuk mengungkap wawasan tentang tren dan temuan utama dalam data penjualan. Analisis data penjualan penting untuk mengidentifikasi pola pembelian pelanggan, mengevaluasi performa produk, dan merumuskan strategi bisnis yang efektif berdasarkan data.

Dataset yang digunakan diunduh dari Kaggle, yang berisi informasi detail tentang transaksi penjualan seperti Order ID, Product, Quantity Ordered, Price Each, Order Date, Purchase Address, Month, Sales, City, dan Hour.

Dataset ini terdiri dari beberapa kolom yang mencakup informasi penting tentang setiap transaksi penjualan. Data telah dibersihkan dari nilai yang hilang dan telah disesuaikan tipe datanya untuk analisis lebih lanjut.

```
Beberapa baris pertama dari dataset:
  Unnamed: 0  Order ID  Product  Quantity Ordered  ...  Month  Sales  City  Hour
0           0    295665  Macbook Pro Laptop      1  ...   12  1700.00  New York City  0
1           1    295666  LG Washing Machine      1  ...   12   600.00  New York City  7
2           2    295667  USB-C Charging Cable      1  ...   12    11.95  New York City  18
3           3    295668  27in FHD Monitor      1  ...   12   149.99  San Francisco  15
4           4    295669  USB-C Charging Cable      1  ...   12    11.95  Atlanta      12

[5 rows x 11 columns]
```


Pembersihan Data: Menghapus data yang tidak lengkap dan melakukan penyesuaian tipe data untuk memastikan keakuratan analisis.

Manipulasi Data: Menerapkan teknik manipulasi data untuk menghitung total penjualan per produk, tren penjualan per bulan, dan pola pembelian berdasarkan waktu.

Visualisasi Data: Membuat visualisasi untuk memberikan wawasan yang lebih mendalam seperti Total penjualan per produk, Tren penjualan per bulan, Pola pembelian berdasarkan waktu (jam), dll.

*Pembersihan Data

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 185950 entries, 0 to 185949
Data columns (total 11 columns):
#   Column              Non-Null Count  Dtype  
---  -
0   Unnamed: 0          185950 non-null int64  
1   Order ID            185950 non-null int64  
2   Product             185950 non-null object 
3   Quantity Ordered    185950 non-null int64  
4   Price Each          185950 non-null float64 
5   Order Date          185950 non-null object 
6   Purchase Address    185950 non-null object 
7   Month               185950 non-null int64  
8   Sales               185950 non-null float64 
9   City                185950 non-null object 
10  Hour                185950 non-null int64  
dtypes: float64(2), int64(5), object(4)
memory usage: 15.6+ MB
```

```
df = df.drop(['Unnamed: 0'], axis=1)
df.head()
```

	Order ID	Product	Quantity Ordered	Price Each	Order Date	Purchase Address	Month	Sales	City	Hour
0	295665	Macbook Pro Laptop	1	1700.00	12/30/2019 0:01	136 Church St, New York City, NY 10001	12	1700.00	New York City	0
1	295666	LG Washing Machine	1	600.00	12/29/2019 7:03	562 2nd St, New York City, NY 10001	12	600.00	New York City	7
2	295667	USB-C Charging Cable	1	11.95	12/12/2019 18:21	277 Main St, New York City, NY 10001	12	11.95	New York City	18
3	295668	27in FHD Monitor	1	149.99	12/22/2019 15:13	410 6th St, San Francisco, CA 94016	12	149.99	San Francisco	15
4	295669	USB-C Charging Cable	1	11.95	12/18/2019 12:38	43 Hill St, Atlanta, GA 30301	12	11.95	Atlanta	12

```
df.shape
```

```
(185950, 10)
```

Data telah dibersihkan dengan jumlah data sebanyak 185.950 baris dan 10 kolom

*Hasil Analisis

```
jumlah_produk = df['Product'].nunique()
jumlah_produk
```

19

Jumlah Produk yang dijual berdasarkan jenis sebanyak 19 item

```
pd.DataFrame(df['Quantity Ordered'].describe())
```

Quantity Ordered	
count	185950.000000
mean	1.124383
std	0.442793
min	1.000000
25%	1.000000
50%	1.000000
75%	1.000000
max	9.000000

Rata-rata pembelian dalam sekali transaksi adalah 1 item, paling sedikit membeli 1 item dan paling banyak adalah 9 item

Top 10 Pelanggan Berdasarkan Frekuensi Pembelian:

	count	Order ID	City
54069	5	160873	San Francisco
85018	4	196615	San Francisco
26661	4	178158	San Francisco
9536	4	304802	Los Angeles
76926	4	242936	Austin
59085	4	165665	Los Angeles
82541	4	194253	Austin
17491	4	312462	San Francisco
42662	4	193511	Austin
17429	4	312407	Seattle

Daftar 10 Pelanggan teratas yang paling sering membeli produk

*Hasil Analisis dan Visualisasi

```
total_sales_per_product = total_sales_per_product.sort_values(by='Sales', ascending=False)
```

```
# Plot bar chart
```

```
plt.figure(figsize=(20, 5))
```

```
plt.bar(total_sales_per_product['Product'], total_sales_per_product['Sales'], color='skyblue')
```

```
plt.title('Total Penjualan Produk')
```

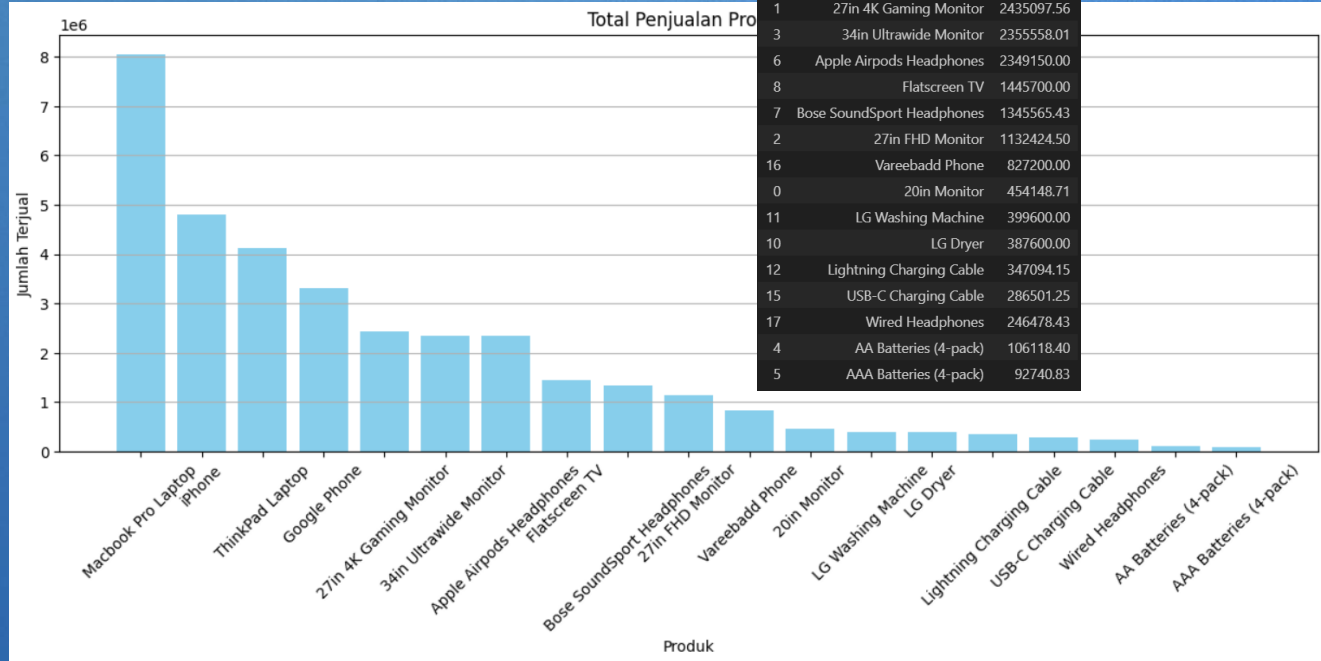
```
plt.xlabel('Produk')
```

```
plt.ylabel('Jumlah Terjual')
```

```
plt.xticks(rotation=45)
```

```
plt.grid(axis='y')
```

```
plt.show()
```



Macbook Pro Laptop memiliki total penjualan tertinggi, diikuti oleh iPhone dan ThinkPad Laptop. Produk dengan penjualan terendah adalah AAA Batteries (4-pack)

*Hasil Analisis dan Visualisasi

```
df = pd.read_csv('Sales Data.csv')
df['Order Date'] = pd.to_datetime(df['Order Date'])

# Filter data untuk rentang waktu yang diinginkan (Januari 2019 sampai Januari 2020)
start_date = '2019-01-01'
end_date = '2020-01-31'
df_filtered = df[(df['Order Date'] >= start_date) & (df['Order Date'] <= end_date)]

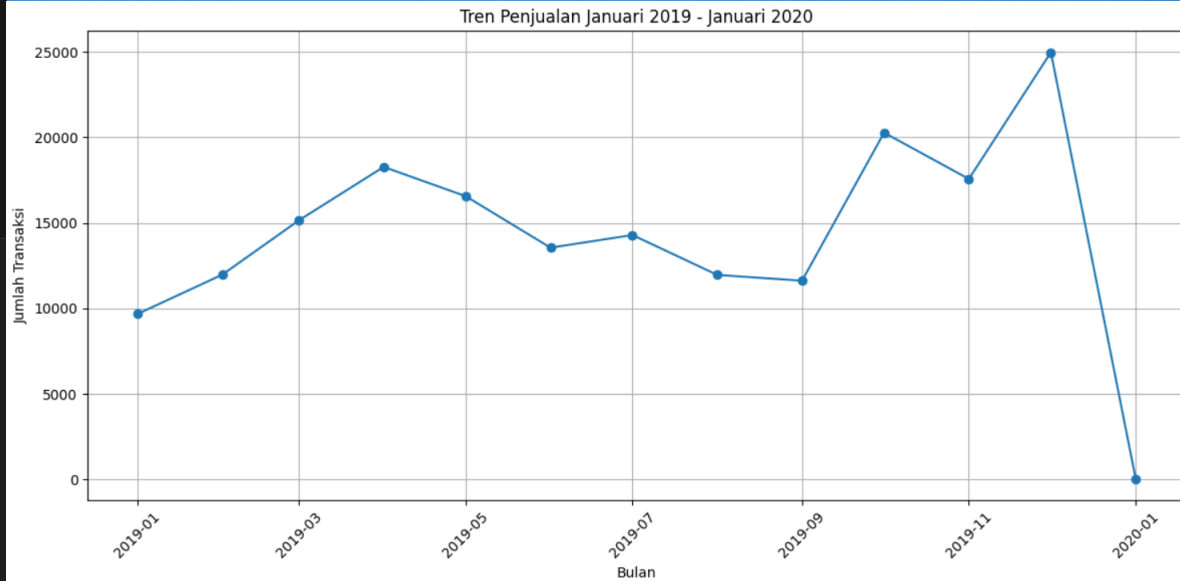
# Ubah kolom 'Order Date' menjadi periode bulan dan tahun saja
df_filtered['Bulan'] = df_filtered['Order Date'].dt.to_period('M')

# Hitung jumlah transaksi per bulan
transactions_per_month = df_filtered.groupby('Bulan').size().reset_index(name='Jumlah Transaksi')

# Ubah 'Bulan' dari Period ke Tanggal (ambil tanggal pertama setiap bulan)
transactions_per_month['Bulan'] = transactions_per_month['Bulan'].apply(lambda x: x.start_time)
transactions_per_month
```

✓ 2.7s

	Bulan	Jumlah Transaksi
0	2019-01-01	9675
1	2019-02-01	11975
2	2019-03-01	15153
3	2019-04-01	18279
4	2019-05-01	16566
5	2019-06-01	13554
6	2019-07-01	14293
7	2019-08-01	11961
8	2019-09-01	11621
9	2019-10-01	20282
10	2019-11-01	17573
11	2019-12-01	24984
12	2020-01-01	34



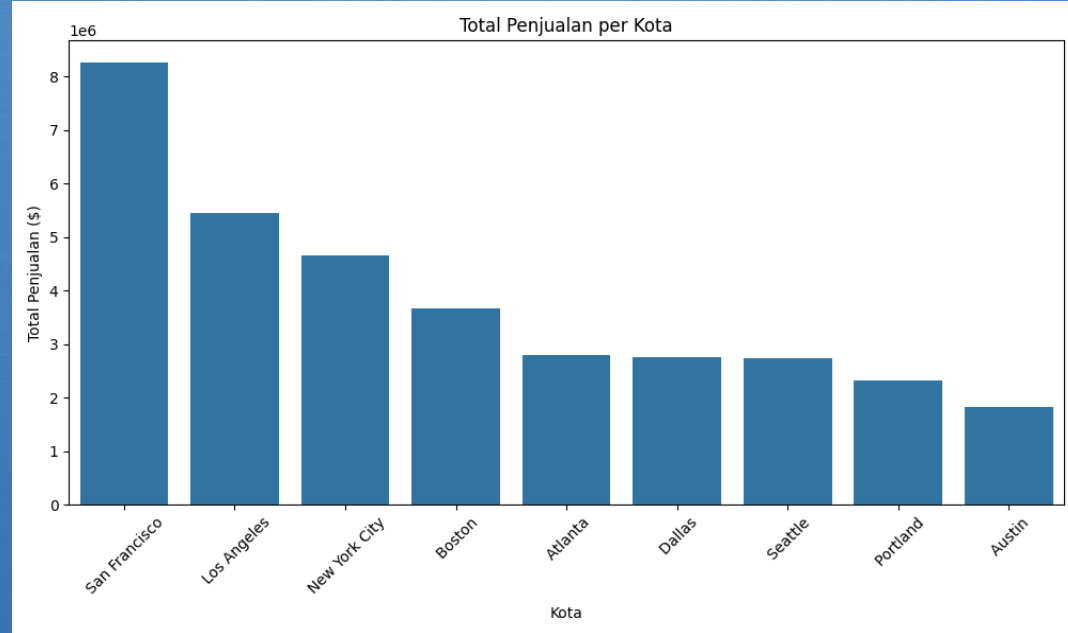
Grafik di atas menunjukkan bahwa penjualan mencapai puncaknya pada bulan Desember 2019 dan mengalami penurunan tajam pada Januari 2020,

*Hasil Analisis dan Visualisasi

```
# Groupby berdasarkan kota dan hitung total penjualan per kota
city_sales = df.groupby('City')['Sales'].sum().reset_index()
```

```
# Urutkan berdasarkan total penjualan dari yang terbesar ke terkecil
city_sales = city_sales.sort_values(by='Sales', ascending=False)
city_sales
```

	City	Sales
7	San Francisco	8262203.91
4	Los Angeles	5452570.80
5	New York City	4664317.43
2	Boston	3661642.01
0	Atlanta	2795498.58
3	Dallas	2767975.40
8	Seattle	2747755.48
6	Portland	2320490.61
1	Austin	1819581.75

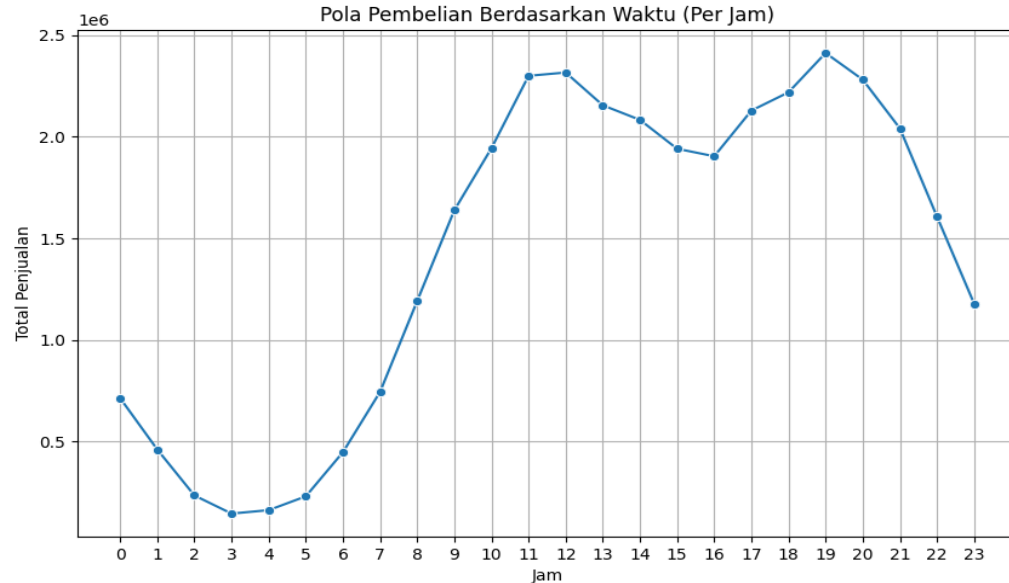


San Fransisco menjadi kota dengan total penjualan tertinggi, diikuti Los Angeles, dan New York City. Interpretasi ini mencerminkan sebaran aktivitas pembelian dalam dataset berdasarkan lokasi geografis.

*Hasil Analisis dan Visualisasi

```
# Groupby berdasarkan jam dan hitung Total Penjualan
hourly_sales = df.groupby('Hour').size().reset_index(name='Sales')
hourly_sales
```

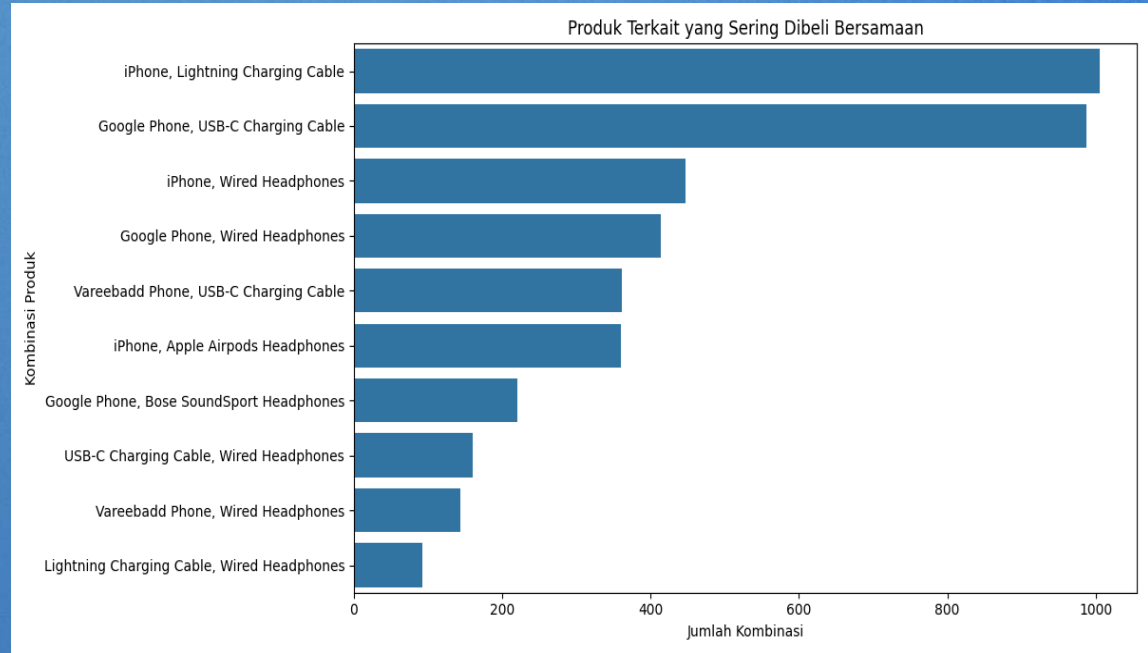
Hour	Counts
0	3910
1	2350
2	1243
3	831
4	854
5	1321
6	2482
7	4011
8	6256
9	8748
10	10944
11	12411
12	12587
13	12129
14	10984
15	10175
16	10384
17	10899
18	12280
19	12905
20	12228
21	10921
22	8822
23	6275



Grafik di atas menunjukkan bahwa pembelian paling banyak terjadi pada jam 18.00–20.00 dan pada jam 11.00–12.00, adapun aktivitas pembelian terendah yaitu terjadi pada dini hari.

*Hasil Analisis dan Visualisasi

```
('iPhone', 'Lightning Charging Cable') 1005  
( 'Google Phone', 'USB-C Charging Cable') 987  
( 'iPhone', 'Wired Headphones') 447  
( 'Google Phone', 'Wired Headphones') 414  
( 'Vareebadd Phone', 'USB-C Charging Cable') 361  
( 'iPhone', 'Apple AirPods Headphones') 360  
( 'Google Phone', 'Bose SoundSport Headphones') 220  
( 'USB-C Charging Cable', 'Wired Headphones') 160  
( 'Vareebadd Phone', 'Wired Headphones') 143  
( 'Lightning Charging Cable', 'Wired Headphones') 92
```



Dengan mengetahui produk yang sering dibeli bersamaan memungkinkan penawaran bundel, meningkatkan nilai penjualan, dan kepuasan pelanggan. Selain itu, pengoptimalan persediaan dapat dilakukan dengan memastikan stok yang cukup untuk produk-produk ini, mengurangi risiko kehabisan stok dan memaksimalkan penjualan. Pemahaman tentang pola pembelian ini membantu merancang strategi pemasaran yang lebih efektif dan efisien.



Terima Kasih