

Code part 1

```
# Import library ต่างๆ เช่น pandas and matplotlib

import pandas as pd

from sklearn.preprocessing import MinMaxScaler, StandardScaler, OrdinalEncoder

from sklearn.model_selection import train_test_split

from sklearn.manifold import TSNE

from sklearn.preprocessing import StandardScaler

import matplotlib.pyplot as plt

import seaborn as sns

from matplotlib.ticker import FuncFormatter

import os


# Read data file from csv.

df = pd.read_excel('/kaggle/input/dataset-transportation/Data_Rail_clean.xlsx')


# clean ช่องว่างจาก text

df.columns = df.columns.str.strip()


# clean เปลี่ยน name header month

df = df.rename(columns={'Month.1': 'Month'})


# check string ทุก row and column และลบช่องว่างออก

df = df.applymap(lambda x: x.strip() if isinstance(x, str) else x)


# Drop rows where Year is 2025

df = df[df['Year'] != 2025]


# Confirm removal

year_counts = df['Year'].value_counts().sort_index()

year_counts
```

```
# import os เพื่อทำการ save image output
os.makedirs("images", exist_ok=True)
```

Graph 1

สรุปได้วิเคราะห์ Travel Mode Bangkok and Provincial

```
grouped_abs = df.groupby(['Year', 'Travel Mode'])['Quantity'].sum().unstack().fillna(0)

grouped_pct = grouped_abs.div(grouped_abs.sum(axis=1), axis=0) * 100

plt.figure(figsize=(20, 10), dpi=100)
ax = plt.gca()

grouped_abs.plot(kind='barh', stacked=True, ax=ax, title='Annual Travel Mode Summary in Bangkok and Provincial')

for i, year in enumerate(grouped_pct.index):
    left = 0
    for mode in grouped_pct.columns:
        width = grouped_abs.loc[year, mode]
        pct = grouped_pct.loc[year, mode]
        if pct > 2:
            ax.text(left + width / 2, i, f"{width * 1e-6:.0f} M", ha='center', va='center', fontsize=11)
            left += width

for i, year in enumerate(grouped_abs.index):
    total = grouped_abs.loc[year].sum()
    ax.text(total + 10_000_000, i, f"{total * 1e-6:.0f} M", va='center', fontsize=11, fontweight='bold')

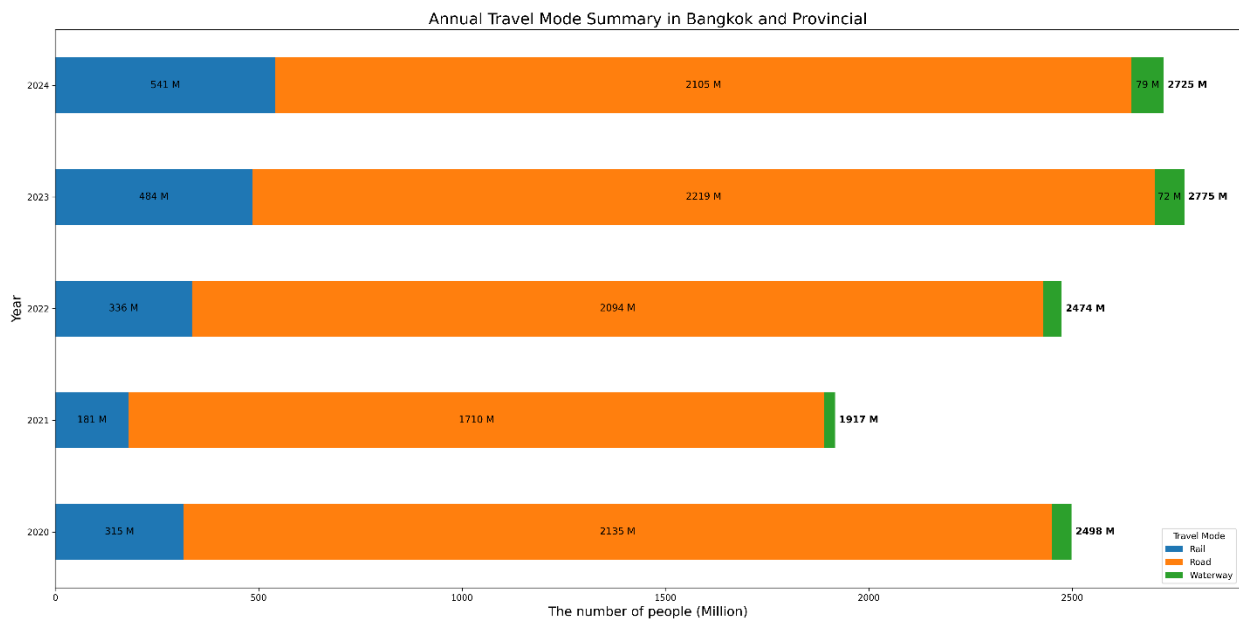
ax.set_xlabel('The number of people (Million)', fontsize=15)
ax.set_ylabel('Year', fontsize=15)
ax.set_title('Annual Travel Mode Summary in Bangkok and Provincial', fontsize=18)

ax.xaxis.set_major_formatter(FuncFormatter(lambda x, _: f'{x * 1e-6:.0f}'))

plt.tight_layout()
plt.savefig("images/travel_mode_quantity.png", dpi=300)
plt.show()
```

1. `grouped_abs = df.groupby(['Year', 'Travel Mode'])['Quantity'].sum().unstack().fillna(0)`
 - รวมจำนวนคน (Quantity) แยกตาม Year และ Travel Mode
 - เติมค่าว่าง (Null) ด้วย 0 โดยใช้คำสั่ง `.fillna(0)`
2. `grouped_pct = grouped_abs.div(grouped_abs.sum(axis=1), axis=0) * 100`
 - คำนวณ percent การใช้แต่ละโหมดต่อยอดรวมของปีนั้น ๆ
3. `plt.figure(figsize=(20, 10), dpi=100)`
 - สร้างกรอบกราฟใหม่ ขนาด 20 x 10 ความชัด dpi=100

4. `grouped_abs.plot(kind='barh', stacked=True, ax=ax, ...)`
 - plot graph **กราฟแท่งแนวนอนแบบ stacked** แสดงจำนวนคนแต่ละโหมดต่อปี
5. วาดรูปแต่ละปีใหม่อีกครั้ง for i, year in enumerate(grouped_abs.index):
6. `total = grouped_abs.loc[year].sum()` >> คำนวณยอดรวมทั้งหมดของปีนั้นๆ
7. แปลงค่าแกน X ให้แสดงหน่วยเป็น จาก หน่วย "Billion" "Million"
8. `plt.savefig("images/travel_mode_quantity.png", dpi=300)` → บันทึกเป็นภาพ PNG
9. `plt.show()` → แสดงผลกราฟ



Graph 2

สรุปโค้ตวิเคราะห์ Travel Mode Bangkok

```
grouped_abs = df_bangkok.groupby(['Year', 'Travel Mode'])['Quantity'].sum().unstack().fillna(0)
grouped_pct = grouped_abs.div(grouped_abs.sum(axis=1), axis=0) * 100

plt.figure(figsize=(20, 10), dpi=100)
ax = plt.gca()

grouped_abs.plot(kind='barh', stacked=True, ax=ax, title='Annual Travel Mode Summary in Bangkok Areas')

# Add percent labels on the bars
for i, year in enumerate(grouped_pct.index):
    left = 0
    for mode in grouped_pct.columns:
        width = grouped_abs.loc[year, mode]
        pct = grouped_pct.loc[year, mode]
        if pct > 5:
            ax.text(left + width / 2, i, f"{width * 1e-6:.0f} M", ha='center', va='center', fontsize=11)
            left += width

for i, year in enumerate(grouped_abs.index):
    total = grouped_abs.loc[year].sum()
    ax.text(total + 10_000_000, i, f"{total * 1e-6:.0f}M", va='center', fontsize=11, fontweight='bold', color='black')

ax.xaxis.set_major_formatter(FuncFormatter(lambda x, _: f'{x * 1e-6:.0f}'))

ax.set_xlabel('The number of people (Million)', fontsize=15)

ax.set_ylabel('Year', fontsize=15)

ax.set_title('Annual Travel Mode Summary in Bangkok Areas', fontsize=18)
plt.tight_layout()
plt.savefig("images/travel_mode_quantit2.png", dpi=300)
plt.show()
```

1. Filter data ให้เหลือเฉพาะ Bangkok เพื่อดู แนวโน้มของคนเดินทางใน กรุงเทพมหานคร
`df[df['Purpose'] == 'Bangkok']`
2. `grouped_abs = df_bangkok.groupby(['Year', 'Travel Mode'])['Quantity'].sum().unstack().fillna(0)`
 - จัดกลุ่มตาม Year และ Travel Mode
 - รวมจำนวนผู้เดินทาง (Quantity) ต่อปี/โหมด
 - ใช้ `.unstack()` เพื่อแปลง Travel Mode เป็น column
 - ถ้าไม่มีข้อมูลบางช่อง ให้เติม 0 ด้วย `.fillna(0)`
3. `grouped_pct = grouped_abs.div(grouped_abs.sum(axis=1), axis=0) * 100`
 - คำนวณสัดส่วน (%) การใช้แต่ละ Travel Mode ต่อปี

4. grouped_abs.plot(kind='barh', stacked=True, ax=ax, title='Annual Travel Mode Summary in Bangkok Areas')
5. ใส่ตัวเลขของบนแต่ละแท่ง โดยคำสั่ง for ทำการวนลูป แต่ละปี โดยใช้ Traveler mode และ มีการใช้ condition สัดส่วน > 2% ของข้อมูลจึงจะแสดงบนกราฟ

```
for i, year in enumerate(grouped_pct.index):
    left = 0
    for mode in grouped_pct.columns:
        width = grouped_abs.loc[year, mode]
        pct = grouped_pct.loc[year, mode]
        if pct > 2:
            ax.text(left + width / 2, i, f"{width * 1e-6:.0f} M", ha='center', va='center', fontsize=11)
            left += width
```

6. แสดงยอดรวมของแต่ละปี (รวมทุก Traveler mode) และ ax.text ปรับแกน x และ label ของแกน x ให้สวยงาม

```
for i, year in enumerate(grouped_abs.index):
    total = grouped_abs.loc[year].sum()
    ax.text(total + 10_000_000, i, f"{total * 1e-6:.0f}M", va='center',
            fontsize=11, fontweight='bold', color='black')
```



Graph 3

สรุปโค้ดวิเคราะห์ Travel Mode Bangkok

```
plt.figure(figsize=(15, 10), dpi=100)
ax = plt.gca()

color_map = {
    'BTS': '#66bb6a',    # train
    'MRT': '#1e88e5',    # apl
    'ARL': '#e57373',    # mrt
    'Train': '#7986cb'   # bts
}
colors = [color_map[col] for col in grouped_pct.columns]

grouped_pct.plot(kind='bar', stacked=True, ax=ax, color=colors)
for i, year in enumerate(grouped_pct.index):
    bottom = 0
    for rtype in grouped_pct.columns:
        height = grouped_pct.loc[year, rtype]
        if height > 2:
            ax.text(i, bottom + height / 2, f"{height:.1f}%", ha='center', va='center', fontsize=10)
            bottom += height

ax.set_ylabel('Percentage (%)', fontsize=15)
ax.set_xlabel('Year', fontsize=15)
ax.set_ylim(0, 100)
ax.set_xticklabels(grouped_pct.index, rotation=0, fontsize=11)
ax.legend(
    loc='upper center',
    bbox_to_anchor=(0.5, -0.08),
    ncol=4,
    fontsize=10,
    columnspacing=1.5,
    handletextpad=1.2,
    frameon=False
)

ax.set_title('Annual Passenger Share by Rail Type in Bangkok', fontsize=18)
plt.tight_layout()
plt.savefig("images/rail_type_share_percent4.png", dpi=300)
plt.show()
```

โค้ดนี้สร้างกราฟแท่งแนวดิ่งแบบ stacked (ซ้อนกัน) เพื่อแสดงว่าในแต่ละปี ผู้โดยสารในกรุงเทพฯ เลือกใช้ระบบรางประเภทต่าง ๆ เช่น BTS, MRT, ARL และรถไฟ (Train) ในสัดส่วนเท่าไรเมื่อเทียบกับเป็นเปอร์เซ็นต์ 100%.

เริ่มต้นด้วยการสร้างพื้นที่กราฟ โดยกำหนดขนาดให้กว้าง 15 นิ้ว สูง 10 นิ้ว และใช้ DPI ที่คมชัดระดับ 100 แล้วดึงแกนกราฟออกมาเก็บไว้ในตัวแปร ax เพื่อใช้ควบคุมรายละเอียดการ plot graph จากนั้นใช้ grouped_pct.plot() เพื่อวาดกราฟแท่งแนวดิ่งแบบซ้อน โดยข้อมูลในแต่ละปีจะมีแท่งแสดงเปอร์เซ็นต์ของแต่ละระบบรางแบบ stacked รวมกันให้ครบ 100% พอได้แท่งกราฟแล้ว โปรแกรมจะวนซ้ำทุกปี เพื่อใส่ตัวเลขเปอร์เซ็นต์ไว้ตรงกลางของแต่ละแท่งย่อยในปีนั้น โดยจะใส่เฉพาะเมื่อค่าสูงกว่า 2% หลังจากนั้นจะตั้งชื่อแกน y ให้เป็น "Percentage (%)" และแกน x เป็น "Year" พร้อมจำกัดขอบเขตแกน y ให้

อยู่ระหว่าง 0 ถึง 100 สุดท้ายจะตั้งชื่อกราฟว่า "Annual Passenger Share by Rail Type in Bangkok" จากนั้นจัดระยะช่องไฟ
ทั้งหมดด้วย tight_layout() แล้วเซฟกราฟเป็นภาพ .png เก็บไว้

