


Class period 19

บทที่ 7 การแสดงผลการเปรียบเทียบข้อมูล (ต่อ)

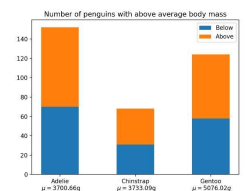
Stacked bar chart and histogram

1




Stacked Bar Chart

- เป็นกราฟแท่งที่สามารถใช้ในการเปรียบเทียบข้อมูลโดยรวมของแต่ละตัวแปร และภายในแท่งกราฟแต่ละแท่งจะแสดงสัดส่วนของข้อมูลของแต่ละกลุ่ม




2



ขั้นตอนการสร้าง Stacked Bar Chart

- ขั้นตอนการสร้างเหมือนกับ Grouped bar chart
- เตรียมข้อมูลแต่ละกลุ่มที่ต้องการและปรับค่าไว้ในตัวแปรเพื่อนำมาสร้างกราฟ (จะใช้ข้อมูล b4lunch, b4midnight ที่เตรียมไว้ก่อนหน้านี้)
- กำหนดตัวแปร labels ให้กับชื่อแท่งกราฟ (จะใช้ ตัวแปร labels ที่เตรียมไว้ก่อนหน้านี้)
- นำตัวแปรที่เตรียมไว้มาสร้างกราฟ

3



เตรียมตัวแปรที่ใช้สร้างกราฟ

```

import matplotlib.pyplot as plt
width = 0.35
fig, ax = plt.subplots()

```

4



สร้างกราฟ


```

ax.bar(labels, b4lunch, width, label='before lunch',color = '#fc9700')
ax.bar(labels, b4midnight, width, bottom=b4lunch, label='before midnight',color = '#19038a')

```

- Parameter: bottom ใช้ในการกำหนดจุดเริ่มต้นที่ต้องการให้วาดกราฟต่อ เช่น
- กำหนด bottom=b4lunch ในการวาดกราฟของข้อมูล b4midnight
- หมายถึงให้วาดกราฟของข้อมูล b4midnight ต่อบนกราฟของข้อมูล b4lunch

5



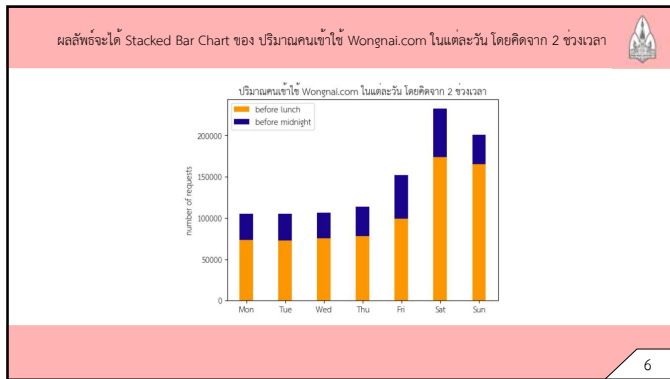
กำหนดส่วนประกอบกราฟต่างๆ

```

ax.set_ylabel('number of requests')
ax.set_title('ปริมาณคนเข้าใช้ Wongnai.com ในแต่ละวัน โดยคิดจาก 2 ช่วงเวลา')
ax.legend()
plt.show()

```

6



7

การสร้างกราฟแท่งแนวนอน

- โดยจะใช้ `ax.barh` ในการสร้างกราฟแท่งแนวนอน เช่น
- `ax.barh(labels, b4lunch, width, label='before lunch',color = '#fc9700')`
- `ax.barh(labels, b4midnight, width, left=b4lunch, label='before midnight',color = '#19038a')`

8

ตัวอย่างการสร้างกราฟแท่งแนวนอน

```

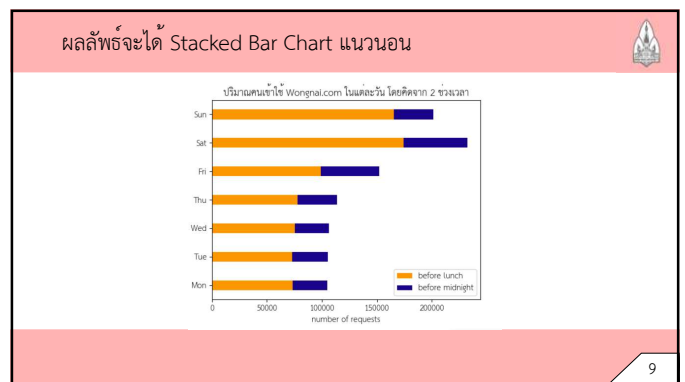
width = 0.35
fig, ax = plt.subplots()

ax.barh(labels, b4lunch, width, label='before lunch',color = '#fc9700')
ax.barh(labels, b4midnight, width, left=b4lunch, label='before midnight',color = '#19038a')

ax.set_xlabel('number of requests')
ax.set_title('ปริมาณคนเข้าใช้ Wongnai.com ในแต่ละวัน โดยคิดจาก 2 ช่วงเวลา')
ax.legend()
plt.show()

```

9



10

Quiz (15 นาที)

- โดยให้วาด Stacked Bar Chart ที่เปรียบเทียบปริมาณคนใช้งาน Wongnai.com สองช่วงเวลา โดยให้กราฟแสดงสัดส่วนของปริมาณคนใช้งานในแต่ละวันด้วย

11



12

การสร้าง Histogram



- สามารถสร้างได้ด้วยคำสั่ง
- `plt.hist('ตัวแปรข้อมูลที่ต้องการสร้างกราฟ', 'ช่วงของข้อมูลหรือจำนวนแท่ง(int)')`

12

13

ตัวอย่างข้อมูลที่ random



- `import numpy as np`
- `from matplotlib import pyplot as plt`
- `np.random.seed(2021)`
- `mu, sigma = 100, 15`
- `X = mu + sigma * np.random.randn(10000)`
- random มาจาก normal distribution ที่มี mean = 100 และ stdev = 15 เก็บไว้ในตัวแปร X

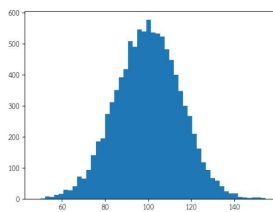
13

14

สร้างกราฟ Histogram ของข้อมูลที่ random



- `plt.hist(X, 50);`



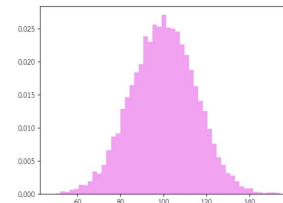
14

15

เพิ่ม Parameter ต่างๆทำให้กราฟด้วยสวยงาม



- `plt.hist(X, 50, density = True, facecolor = 'violet', alpha = 0.75);`



15

16