ปฏิบัติการครั้งที่ 3 กระบวนวิชา 229351 Statistical Learning for Data Science

คำชี้แจง

1. ให้เริ่มทำปฏิบัติการจาก colab notebook ที่กำหนดให้ จากนั้นบันทึกเป็นไฟล์ *.pdf

ในปฏิบัติการนี้เราจะฝึกการทำ PCA ด้วยสองวิธี คือ

1. หาด้วยการแยกส่วนประกอบของเมทริกซ์ผ่าน numpy
2. หาด้วยการใช้เครื่องมือที่มีมาให้ใน scikit-learn

ดาวน์โหลดข้อมูลสัตว์ต่างๆ 50 ชนิดดังนี้
ชนิดสัตว์: https://donlapark.pages.dev/229351/data/classes.txt
ตัวแปรต่างๆ: https://donlapark.pages.dev/229351/data/predicates.txt
ค่าของสัตว์แต่ละชนิด: https://donlapark.pages.dev/229351/data/predicate-matrix-continuous.txt

```
!pip install -q wget

import numpy as np
from matplotlib import pyplot as plt
from sklearn.decomposition import PCA
```

```
  Preparing metadata (setup.py) ... done
  Building wheel for wget (setup.py) ... done
```

```
# Download the files
!wget -O classes.txt https://donlapark.pages.dev/229351/data/classes.txt
!wget -O predicate-matrix-continuous.txt https://donlapark.pages.dev/229351/data/predicate-matrix-continuous.txt
```

```
--2025-07-23 13:00:08--  https://donlapark.pages.dev/229351/data/classes.txt
Resolving donlapark.pages.dev (donlapark.pages.dev)... 172.66.47.56, 172.66.44.200, 2606:4700:310c::ac42:2f38, ...
Connecting to donlapark.pages.dev (donlapark.pages.dev)|172.66.47.56|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 755 [text/plain]
Saving to: ‘classes.txt’

classes.txt         100%[===================>]     755  --.-KB/s    in 0s

2025-07-23 13:00:08 (10.9 MB/s) - ‘classes.txt’ saved [755/755]

--2025-07-23 13:00:09--  https://donlapark.pages.dev/229351/data/predicate-matrix-continuous.txt
Resolving donlapark.pages.dev (donlapark.pages.dev)... 172.66.47.56, 172.66.44.200, 2606:4700:310c::ac42:2f38, ...
Connecting to donlapark.pages.dev (donlapark.pages.dev)|172.66.47.56|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 29800 (29K) [text/plain]
Saving to: ‘predicate-matrix-continuous.txt’

predicate-matrix-co 100%[===================>]  29.10K  --.-KB/s    in 0s

2025-07-23 13:00:09 (109 MB/s) - ‘predicate-matrix-continuous.txt’ saved [29800/29800]
```

```
classes = np.genfromtxt('classes.txt',dtype='str')

classes[:5]
```

```
array([['1', 'antelope'],
       ['2', 'grizzly+bear'],
       ['3', 'killer+whale'],
       ['4', 'beaver'],
       ['5', 'dalmatian']], dtype='<U15')
```

```
data = np.genfromtxt('predicate-matrix-continuous.txt')

data[:5,:5]
```

```
array([[-1.  , -1.  , -1.  , -1.  , 12.34],
       [39.25,  1.39,  0.  , 74.14,  3.75],
       [83.4 , 64.79,  0.  ,  0.  ,  1.25],
```

```
       [19.38,  0.  ,  0.  , 87.81,  7.5 ],
       [69.58, 73.33,  0.  ,  6.39,  0.  ]])
```

Exercise 1

ใน code block ข้างล่างนี้ จงทำ PCA บนข้อมูลที่ได้มาให้เหลือเมทริกซ์ข้อมูลที่มีตัวแปรแค่ 2 ตัว โดยใช้ฟังก์ชัน `np.linalg.eigh`
ดังนั้น เมทริกซ์ที่ได้ต้องมีขนาด 50x2

$$\Sigma = \begin{pmatrix} \mathrm{var}(X_1) & \mathrm{cov}(X_1, X_2) & \cdots & \mathrm{cov}(X_1, X_{85}) \\ \mathrm{cov}(X_2, X_1) & \mathrm{var}(X_2) & \cdots & \mathrm{cov}(X_2, X_{85}) \\ \vdots & \vdots & \ddots & \vdots \\ \mathrm{cov}(X_{85}, X_1) & \mathrm{cov}(X_{85}, X_1) & \cdots & \mathrm{var}(X_{85}) \end{pmatrix}$$

```
# TODO: enter code here
import numpy as np
# 1. Center ข้อมูล สมมติว่า data_c คือ normalized (centered) data
data_c = data - np.mean(data, axis=0)

# 2. หา covariance matrix
cov_matrix = np.cov(data_c, rowvar=False)

# 3. Decompose the covariance matrix UDU^T
eigenvalues, eigenvectors = np.linalg.eigh(cov_matrix)

# 4. ดึง column ของ U ที่ประกอบไปด้วย eigenvector สองตัวที่มีค่า eigenvalue สูงที่สุด
sorted_idx = np.argsort(eigenvalues)[::-1]
top2_eigenvectors = eigenvectors[:, sorted_idx[:2]]

# 5. เอา data_c ไปทำ projection ทิศทางของ eigenvector ใน U
projected_data = data_c @ top2_eigenvectors

# Result: a (50,2) matrix
print(projected_data.shape)
```

```
(50, 2)
```

OPTIONAL

ใน code block ข้างล่างนี้ จงทำ PCA บนข้อมูลที่ได้มาให้เหลือเมทริกซ์ข้อมูลที่มีตัวแปรแค่ 2 ตัว โดยใช้ $scikit-learn$
พร้อมกับตรวจสอบว่าเมทริกซ์ที่ได้จากทั้งสองวิธีนี้มีค่าเท่ากัน (ต่างกันแค่ค่าบวกลบ)

```
from sklearn.decomposition import PCA

# ใช้ sklearn PCA
pca = PCA(n_components=2)
sklearn_projected = pca.fit_transform(data)

# ตรวจสอบว่าเหมือนกันหรือต่างกันแค่สัญญาณ(บวกลบ)
equal_check = np.allclose(np.abs(sklearn_projected), np.abs(projected_data))
print("Equal (up to sign):", equal_check)
```

```
Equal (up to sign): True
```

```
#TODO (optional): enter code here
```

Exercise 2

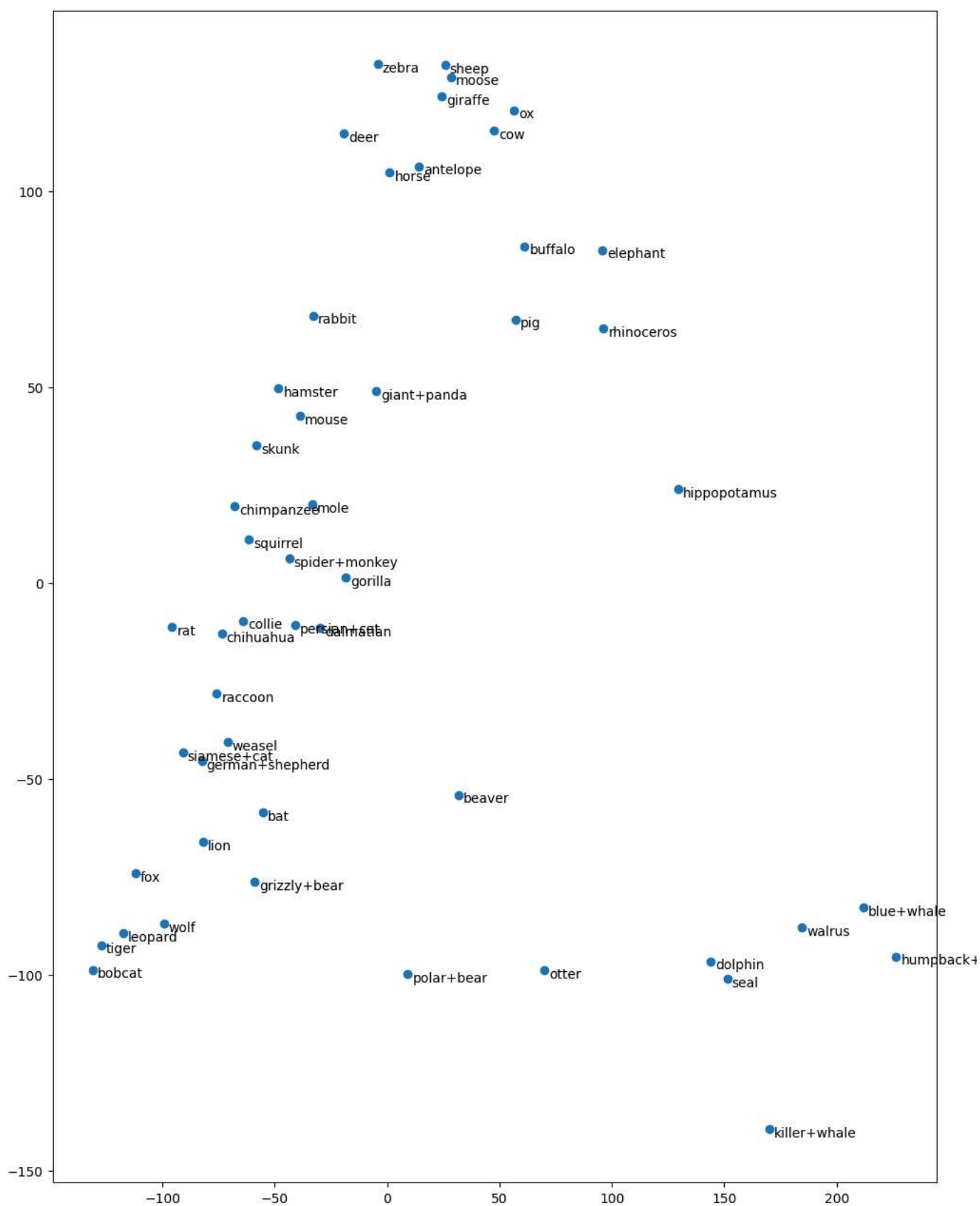
ทำการพล็อตจุดของข้อมูลที่ได้จาก PCA ข้างบนพร้อมกับใส่คำกำกับว่าจุดไหนเป็นของสัตว์ชนิดใดโดยใช้ชื่อสัตว์จาก classes.txt

```
# กำหนดขนาดของรูป
plt.figure(figsize=(12,16))

# จงเติม argument ที่เหมาะสมในวงเล็บข้างล่างนี้
# รูปแบบของฟังก์ชันคือ plt.scatter(numpy array ของ x-coordinate, numpy array ของ y-coordinate)
plt.scatter(projected_data[:,0], projected_data[:,1])#TODO: 1st column of your 50x2 matrix, #TODO: 2nd column of your 50x2 matrix )
```

```
# for loop เพื่อใส่คำกำกับ (annotate) ชื่อสัตว์ของแต่ละจุด วนให้ครบสัตว์ทุกชนิดที่อยู่ใน classes
# ใส่ numpy array ที่ผ่านการทำ PCA แล้วลงในตำแหน่งที่ระบุเพื่อบอกพิกัดที่ต้องวางคำกำกับ
for i in range(50):
  plt.annotate(classes[i,1],xy=(projected_data[i,0], projected_data[i,1])#TODO: YOUR_MATRIX[i,0],#TODO YOUR_MATRIX[i,1]
               ,xytext=(5, -8),textcoords='offset pixels')


plt.show()
```



Exercise 3

หากลุ่มสัตว์ต่างๆ ที่อยู่ใกล้กันมา 4 กลุ่ม แล้วลองอธิบายว่าสัตว์ในแต่ละกลุ่มมีอะไรที่เหมือนกัน

1. ตัวอย่าง: กลุ่มสัตว์ที่มุมขวาบน ตัวอย่างเช่น... มีลักษณะที่เหมือนกันคือ...
2. มุมขวาบน (สัตว์เลี้ยงลูกด้วยนมกินพืชขนาดใหญ่)ได้แก่ zebra, sheep, moose, giraffe, ox, cow, horse, antelope, deer มีลักษณะที่เหมือนกันคือ -เป็นสัตว์กินพืช -มีลักษณะกายภาพคล้ายกัน เช่น เดินสี่ขา, มีเขาและมีขน -มักพบในทุ่งหญ้าหรือป่า
3. มุมล่างขวา (สัตว์ทะเลขนาดใหญ่) ได้แก่ killer whale, humpback whale, blue whale, dolphin, seal, walrus มีลักษณะที่เหมือนกันคือ -เป็นสัตว์ทะเลทั้งหมด -ส่วนใหญ่เป็นสัตว์เลี้ยงลูกด้วยนมในทะเล -อาศัยในน้ำลึก/มหาสมุทร
4. กลางล่างซ้าย (สัตว์กินเนื้อ) ได้แก่ tiger, lion, wolf, leopard, bobcat, fox มีลักษณะที่เหมือนกันคือ -เป็นสัตว์กินเนื้อ -มีฟันเขี้ยวและกรงเล็บแหลม -เป็นนักล่า
5. กลาง (สัตว์เลี้ยงลูกด้วยนมขนาดเล็ก) ได้แก่ hamster, rabbit, mouse, skunk, mole, squirrel มีลักษณะที่เหมือนกันคือ -ขนาดตัวเล็ก -ส่วนใหญ่อยู่ในลำดับสัตว์ฟันแทะหรือเลี้ยงลูกด้วยนมขนาดเล็ก -พบในพื้นที่ป่าหรือในบ้าน