เอกสารอธิบายการทดลองที่ 1 ตอนที่ 2

พื้นฐานการอ่านไฟล์ข้อมูล การแก้ปัญหาข้อมูลหาย การปรับช่วงค่าของข้อมูล การปรับลดมิติข้อมูล และแสดงผลข้อมูลในเชิงกราฟ

ตอนที่ 2: การทดลองการลดมิติของข้อมูลด้วยค่า Principle Component Analysis

2.1 เตรียมชุดข้อมูล feature 3 ค่า $\boldsymbol{X}$(accelerateX, accelerateY, accelerate

- ปรับให้เป็น zero mean: โดยจะทำการวนแต่ละแถว ซึ่งค่าของ column นั้นๆ จะถูกลบกับค่าเฉลี่ยของ column นั้นๆ

```
for column in df.columns :
    df[column] = df.apply(lambda row: (row[column] - df[column].mean()), axis=1)
df
```

uts	accelerateX	accelerateY	accelerateZ
2018-11-18 08:18:40+07:00	-0.172321	-0.350530	-0.030806
2018-11-18 08:19:00+07:00	-0.046506	0.083251	-0.044434
2018-11-18 08:19:20+07:00	-0.045278	0.121703	-0.105454
2018-11-18 08:19:40+07:00	0.015941	-0.023881	-0.246696
2018-11-18 08:20:00+07:00	-0.053794	-0.215708	-0.006878
...	...	...	...
2018-11-18 16:07:40+07:00	0.011875	-0.098047	-0.038007
2018-11-18 16:08:00+07:00	-0.032467	-0.156361	-0.279064
2018-11-18 16:08:20+07:00	-0.045278	0.121703	-0.105454
2018-11-18 16:08:40+07:00	-0.021343	-0.171536	-0.270933
2018-11-18 16:09:00+07:00	-0.029858	-0.167563	-0.256762

- คำนวณค่า covariance matrix ของชุดข้อมูล Xnorm จากสูตรจะได้

```
covariance_matrix = np.dot(df.T, df) / (len(df) -1)
covariance_matrix
```

```
array([[ 0.02336014,  0.00138956, -0.00136398],
       [ 0.00138956,  0.02562116,  0.00397339],
       [-0.00136398,  0.00397339,  0.02035056]])
```

2.2 คำนวณค่า eigenvalue / eigenvector จาก covariance matrix ที่คำนวณได้จากข้อ 2.1

หา eigenvalue และ eigenvector จาก ฟังก์ชัน np.linalg.eig โดยจะใส่พารามิเตอร์เป็น covariance matrix ก่อนหน้า

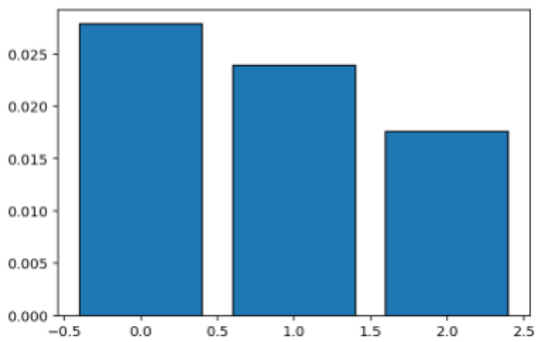
```
# คำนวณค่า eigenvalue / eigenvector จาก covariance matrix
eigenvalue, eigenvecter = np.linalg.eig(covariance_matrix)
print('eigenvalue:', eigenvalue)
print('eigenvecter:', eigenvecter)
```

```
eigenvalue: [0.01761319 0.02388339 0.02783528]
eigenvecter: [[-0.30933198  0.94069151  0.13933128]
 [ 0.46520554  0.02190334  0.88493167]
 [-0.82939589 -0.33855535  0.4443903 ]]
```

2.3 แสดงกราฟ Eigen Space (Eigenvalue, Eigenvector)

- แสดงกราฟแท่ง (Bar graph) ของค่า Eigenvalue ที่จัดเรียงค่าจากมากไปน้อย

```
plt.bar(np.arange(3), height=np.sort(eigenvalue)[::-
1], width=0.8, align='center', edgecolor='k')
```



- แสดงปรับขนาดของ Eigenvector ด้วยค่า Eigenvalue

```
ev1 = eigenvecter[: ,0] * np.sqrt(eigenvalue[0])
ev2 = eigenvecter[: ,1] * np.sqrt(eigenvalue[1])
ev3 = eigenvecter[: ,2] * np.sqrt(eigenvalue[2])
print(ev1, ev2, ev3)
```

```
[-0.04105289  0.0617396  -0.11007301] [ 0.14537683  0.003385   -0.0523212 ] [0.0232459  0.14764119 0.07414167]
```

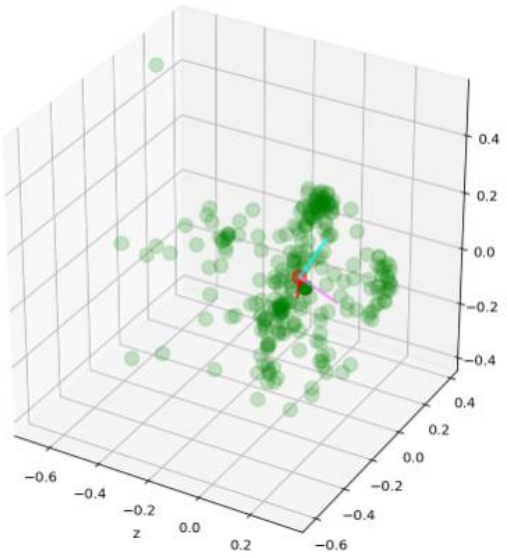
- แสดงกราฟความสัมพันธ์ของ feature และ eigen vector

```
fig = plt.figure(figsize=(35, 8))

ax = fig.add_subplot(141, projection='3d')
ax.plot(df['accelerateX'], df['accelerateY'], df['accelerateZ'], 'o', markersize=10
, color='green', alpha=0.2)
ax.plot([df['accelerateX'].mean()], [df['accelerateY'].mean()], [df['accelerateZ'].
mean()], 'o', markersize=10, color='red', alpha=0.5)

ax.plot([0, ev1[0]], [0, ev1[1]], [0, ev1[2]], color='red', alpha=0.8, lw=2)
ax.plot([0, ev2[0]], [0, ev2[1]], [0, ev2[2]], color='violet', alpha=0.8, lw=2)
ax.plot([0, ev3[0]], [0,  ev3[1]], [0, ev3[2]], color='cyan', alpha=0.8, lw=2)

ax.set_xlabel('x')
ax.set_xlabel('y')
ax.set_xlabel('z')
ax.view_init(30, -60)
plt.show()
```



2.4 ทำ PCA เพื่อลดข้อมูลจาก 3D -> 2D

- ลดมิติของข้อมูลจาก 3D features $\boldsymbol{x}$ (accelerateX, accelerateY, accelerateZ) ลงเหลือ 2D โดย เลือก eigenvector 2 vector แรก ที่สัมพันธ์กับ eigenvalue ที่มีค่าสูงสุด 2 อันดับแรก

```
x_pca =np.dot(np.take(eigenvecter,np.argsort(eigenvalue)[1:].tolist(),axis=0),df.T)
df_x_pca = pd.DataFrame(x_pca)
df_x_pca
```

	0	1	2	3	4	5	6	7	8	9	.	314	315	316
0	-0.115104	-0.059133	-0.111717	-0.211416	-0.035837	-0.100907	-0.021379	-0.056646	-0.325859	-0.256297	.	-0.111717	-0.253443	-0.244777
1	0.247906	-0.009360	-0.050512	-0.114766	0.114589	0.156625	0.446962	-0.168643	0.581331	0.270601	.	-0.050512	-0.044624	-0.032609

2 rows × 317 columns

- แสดงภาพ X_PCA ด้วย sns.heatmap()

```
sns.heatmap(df_x_pca)
```