

# LUNG CANCER PREDICTION

G8 : RAINJERR X แมวอะซอบับคุณ

# ขั้นตอนการทำงาน

- 01 การเตรียมข้อมูล
- 02 การคลีนข้อมูล
- 03 การนำข้อมูลเข้าสู่ตัวโครงงาน
- 04 หาความสัมพันธ์ระหว่างปัจจัยต่างๆด้วย Pearson's Similarity
- 05 ประมวลผลข้อมูลเพื่อเตรียมนำเข้าสู่ตัวโมเดล
- 06 แบ่งข้อมูลเป็นชุดทดลองและชุดทดสอบ
- 07 นำข้อมูลชุดทดลองเข้าสู่โมเดล Logistic Regression
- 08 นำข้อมูลชุดทดสอบเข้าสู่โมเดล Logistic Regression
- 09 นำผลลัพธ์ที่ได้จากโมเดลมาทำ Confusion Matrix
- 10 สร้าง GUI ให้ user ตอบแบบสอบถาม



# การเตรียมข้อมูล

- Import libraries ที่ใช้ใน project และ import dataset

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.preprocessing import LabelEncoder

plt.style.use('fivethirtyeight')
colors=['#011f4b', '#03396c', '#005b96', '#6497b1', '#b3cde0']
sns.set_palette(sns.color_palette(colors))

## Loading Dataset
df = pd.read_csv('survey_lung_cancer.csv')
print(df.head)
```

- Output ที่ได้เป็นข้อมูลภายในไฟล์  
survey\_lung\_cancer.csv (309 rows x 16 columns)

```
<bound method NDFrame.head of
CHEST PAIN  LUNG_CANCER
1          M    74        2          1          1          1          2 ...          1          1          2          2          2          YES
2          F    59        1          1          1          2          1 ...          1          2          2          1          2          NO
3          M    63        2          2          2          1          1 ...          1          1          1          2          2          NO
4          F    63        1          2          1          1          1 ...          2          2          2          1          1          NO
..      ...    ...    ...      ...      ...      ...      ...      ...      ...      ...      ...      ...      ...      ...
304        F    56        1          1          1          2          2 ...          2          2          2          2          1          YES
305        M    70        2          1          1          1          1 ...          2          2          2          1          2          YES
306        M    58        2          1          1          1          1 ...          2          2          1          1          2          YES
307        M    67        2          1          2          1          1 ...          2          2          2          1          2          YES
308        M    62        1          1          1          2          1 ...          2          1          1          2          1          YES

[309 rows x 16 columns]>
```

- **หาสถิติของข้อมูลแต่ละรูปแบบ**

```
## Analysis numerical columns
print(df.describe())
```

## Output :

[illegible]

- สามารถทำออกมาเป็นตารางได้ดังนี้

	AGE	SMOKING	YELLOW FINGERS	ANXIETY	PEER PRESSURE	CHRONIC DISEASE	FATIGUE
count	309.000000	309.000000	309.000000	309.000000	309.000000	309.000000	309.000000
mean	62.673139	1.563107	1.569579	1.498382	1.501618	1.504854	1.673139
std	8.210301	0.496806	0.495938	0.500808	0.500808	0.500787	0.469827
min	21.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000
25%	57.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000
50%	62.000000	2.000000	2.000000	1.000000	2.000000	2.000000	2.000000
75%	69.000000	2.000000	2.000000	2.000000	2.000000	2.000000	2.000000
max	87.000000	2.000000	2.000000	2.000000	2.000000	2.000000	2.000000

[illegible]

4

- เช็คข้อมูลซ้ำกัน

```
# Check for duplicates in the dataset
print('Duplicated:',df.duplicated().sum())
```

**OUT :**

Duplicated: 33

- ลบข้อมูลซ้ำกัน

```
## Drop duplicates value
df.drop_duplicates(inplace=True)
print(df.shape)
```

**OUT :**

(276, 16)

(Before)

(309, 16)

- เช็คช่องข้อมูลที่เป็นช่องว่าง

```
# Check for null values
print(df.isnull().sum())
```

**OUT :**

ไม่มีข้อมูลที่เป็นช่องว่าง

```
GENDER      0
AGE          0
SMOKING      0
YELLOW_FINGERS  0
ANXIETY      0
PEER_PRESSURE  0
CHRONIC_DISEASE  0
FATIGUE      0
ALLERGY      0
WHEEZING     0
ALCOHOL_CONSUMING  0
COUGHING     0
SHORTNESS_OF_BREATH  0
SWALLOWING_DIFFICULTY  0
CHEST_PAIN   0
LUNG_CANCER  0
dtype: int64
```

- ENCODING COLUMNS ที่ไม่เป็นตัวเลข

มีข้อมูลเป็น M/F และ YES/NO ดังนั้นจึงต้องเปลี่ยนข้อมูลเป็นตัวเลขเพื่อให้ง่ายต่อการคำนวณ

```
## Encoding LUNG_CANCER and GENDER column
encoder = LabelEncoder()
df['LUNG_CANCER']=encoder.fit_transform(df['LUNG_CANCER'])
df['GENDER']=encoder.fit_transform(df['GENDER'])
print(df.head)
```

**OUT :**

```
found without NaNs: head of
GENDER  AGE  SMOKING  YELLOW_FINGERS  ANXIETY  PEER_PRESSURE  CHRONIC_DISEASE  ...  MALLING  ALCOHOL_CONSUMING  COUGHING  SHORTNESS_OF_BREATH  SWALLOWING_DIFFICULTY  CHEST_PAIN  LUNG_CANCER
0      1   60      1      2      2      1      1      1      2      2      2      2      2      2      2      1
1      1   76      2      1      1      1      2      1      1      2      2      2      2      2      2      1
2      0   59      1      1      1      2      1      1      2      1      2      2      2      2      2      0
3      1   63      2      2      2      1      1      1      1      2      1      1      2      2      2      0
4      0   63      1      2      1      1      1      1      2      1      2      2      1      1      1      0
...
279     0   59      1      2      2      2      1      1      2      1      2      1      2      1      1      1
280     0   59      2      1      1      1      2      1      1      1      2      1      1      1      1      0
281     1   55      2      1      1      1      1      1      1      1      2      1      1      2      2      0
282     1   46      1      2      2      1      1      1      1      1      1      2      2      2      2      0
283     1   69      1      2      2      1      1      1      2      2      2      2      2      2      2      1
[284 rows x 16 columns]
```

# การคลีนข้อมูล

# การนำข้อมูลเข้าสู่ตัวโครงงาน



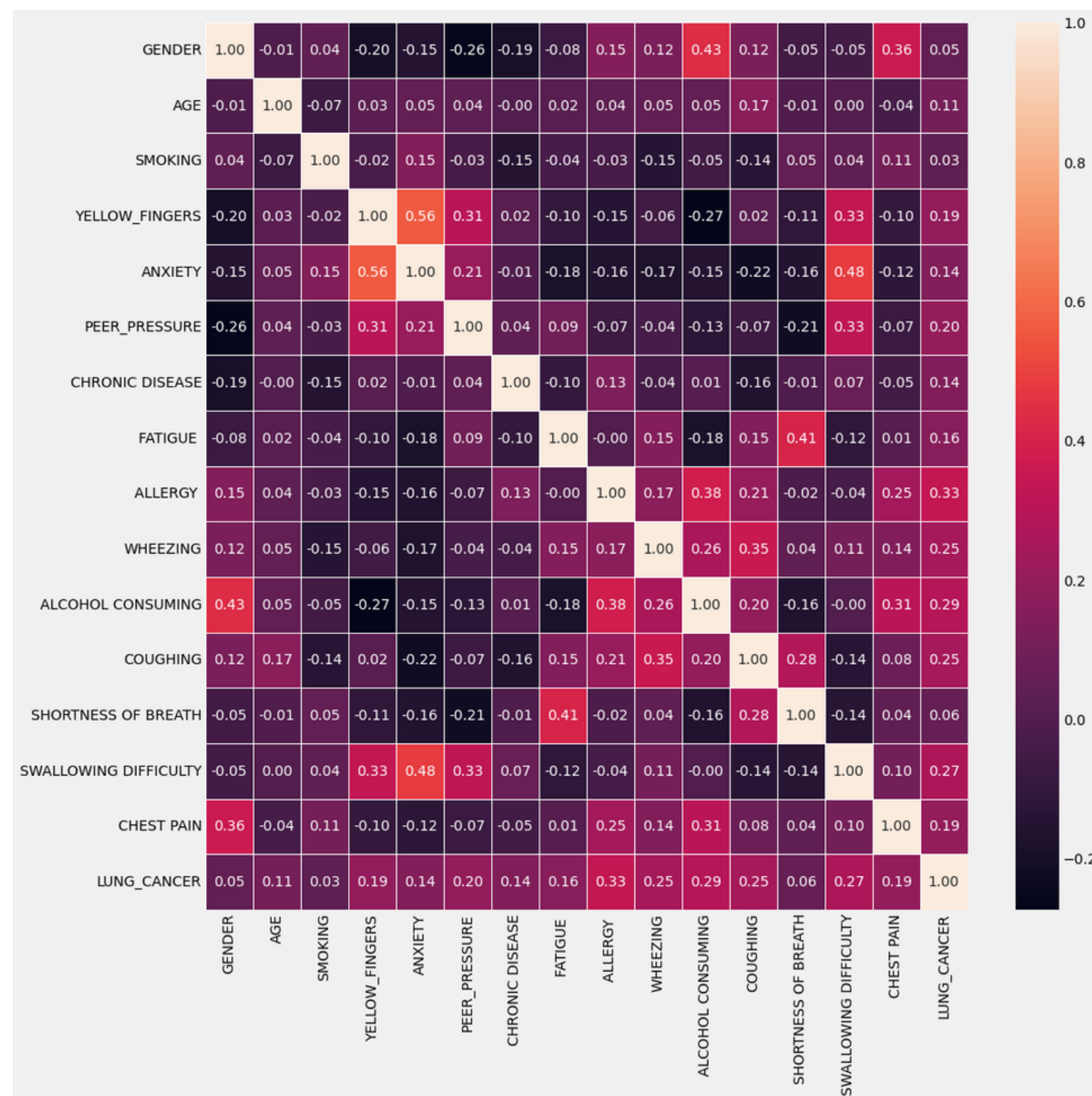
หลังจาก Clean ข้อมูลแล้ว จะเหลือข้อมูลอยู่ทั้งหมด 276 ชุดข้อมูล และ อยู่ในรูปเมทริกซ์จำนวน 15 ชุด

	GENDER	AGE	SPOKING	YELLOW_FINGERS	ANXIETY	FEEL_PRESSURE	CHRONIC_DISEASE	FATIGUE	ALLERGY	WHEEZING	ALCOHOL_CONSUMING	COUGHING	SHORTNESS_OF_BREATH	SWALLOWING_DIFFICULTY	CHEST_PAIN	LUNG_CANCER
0	1	69	1	2	2	1	1	2	1	2	2	2	2	2	2	1
1	1	74	2	1	1	1	2	2	2	1	1	1	2	2	2	1
2	0	59	1	1	1	2	1	2	1	2	1	2	2	1	2	0
3	1	63	2	2	2	1	1	1	1	1	2	1	1	2	2	0
4	0	69	1	2	1	1	1	1	1	2	1	2	2	1	1	0
5	0	75	1	3	1	1	2	2	2	2	1	2	2	1	1	1
6	1	52	2	1	1	1	1	2	1	2	2	2	2	1	2	1
7	0	51	2	2	2	2	1	2	2	1	1	1	2	2	1	1
8	0	68	2	1	2	1	1	2	1	1	1	1	1	1	1	0
9	1	53	2	2	2	2	2	1	2	1	2	1	1	2	2	1
10	0	61	2	2	2	2	2	2	1	2	1	2	2	2	1	1
11	1	72	1	1	1	1	2	2	2	2	2	2	2	1	2	1
12	0	66	2	1	1	1	1	2	1	1	1	1	2	1	1	0
13	1	68	2	1	1	1	1	2	2	2	2	2	2	1	2	1
14	1	69	2	1	1	1	1	1	2	2	2	2	1	1	2	0

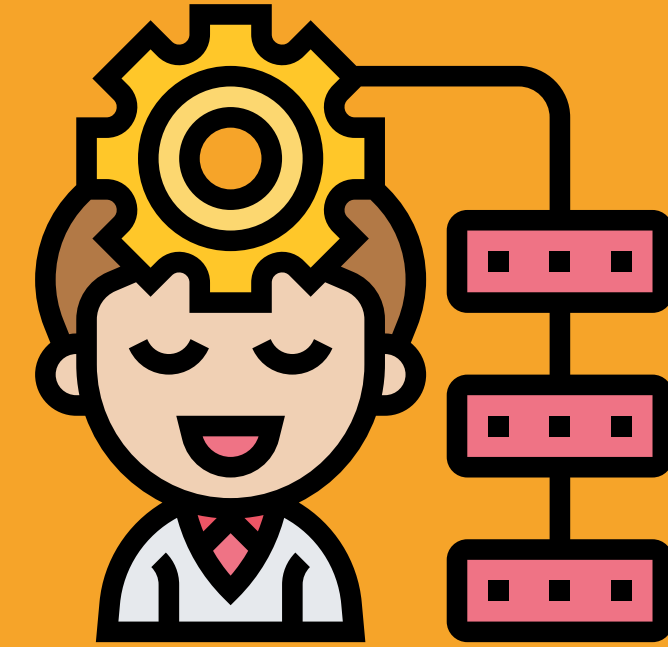


# หาความสัมพันธ์ระหว่างปัจจัยต่างๆด้วย PEARSON'S SIMILARITY

$$CORR(x, y) = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}}$$



# ประมวลผลข้อมูลเพื่อ เตรียมนำเข้าสู่ตัวโมเดล



	GENDER	AGE	SMOKING	YELLOW_FINGERS	ANXIETY	PEER_PRESSURE	CHRONIC_DISEASE	FATIGUE	ALLERGY	WHEEZING
0	1	69	0	1	1	0	0	1	0	1
1	1	74	1	0	0	0	1	1	1	0
2	0	59	0	0	0	1	0	1	0	1
3	1	63	1	1	1	0	0	0	0	0
4	0	63	0	1	0	0	0	0	0	1

WHEEZING	ALCOHOL_CONSUMING	COUGHING	SHORTNESS_OF_BREATH	SWALLOWING_DIFFICULTY	CHEST_PAIN
1	1	1	1	1	1
0	0	0	1	1	1
1	0	1	1	0	1
0	1	0	0	1	1
1	0	1	1	0	0



# แบ่งข้อมูลเป็นชุดทดลองและชุดทดสอบ

```
X_over,y_over=RandomOverSampler().fit_resample(X,y)
```

IN :

```
# Train Test Split
X_train,X_test,y_train,y_test = train_test_split(X_over,y_over,random_state=42,stratify=y_over)
```

```
print(X_train.head())
```

OUT :

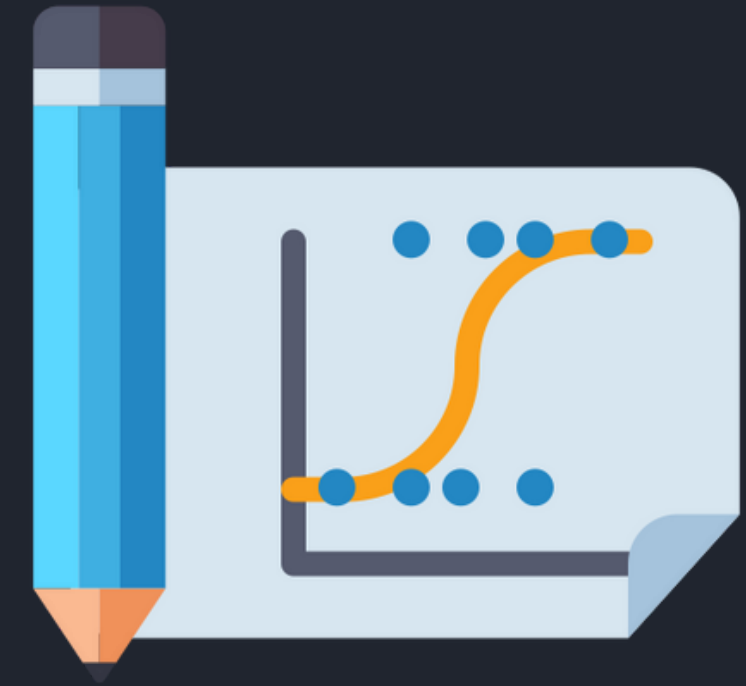
	GENDER	AGE	SMOKING	YELLOW_FINGERS	ANXIETY	PEER_PRESSURE	CHRONIC_DISEASE	FATIGUE	ALLERGY	WHEEZING	ALCOHOL_CONSUMING	COUGHING	SHORTNESS OF BREATH	SWALLOWING DIFFICULTY	CHEST PAIN
321	1	63	1	0	0	1	0	0	0	0	0	1	1	0	0
336	1	55	0	1	0	0	0	1	0	1	1	1	1	0	0
334	1	59	0	1	1	0	0	0	0	0	0	0	0	1	1
151	0	64	1	1	0	1	1	0	0	0	0	0	0	0	0
314	1	69	1	0	0	1	0	0	0	0	0	0	0	0	1

```
print(X_test.head())
```

OUT :

	GENDER	AGE	SMOKING	YELLOW_FINGERS	ANXIETY	PEER_PRESSURE	CHRONIC_DISEASE	FATIGUE	ALLERGY	WHEEZING	ALCOHOL_CONSUMING	COUGHING	SHORTNESS OF BREATH	SWALLOWING DIFFICULTY	CHEST PAIN
129	0	61	1	0	0	0	1	1	1	0	0	0	1	0	0
227	0	49	0	1	1	0	0	0	0	0	0	1	0	0	0
133	0	56	1	1	1	0	0	1	1	0	0	0	1	0	1
100	1	64	1	0	0	0	0	1	1	1	1	1	1	0	1
184	1	55	1	0	0	0	0	1	0	0	0	0	0	0	0

# นำข้อมูลชุดทดลองเข้าสู่โมเดล **LOGISTIC REGRESSION**



**IN :**

```
# Logistic Regression  
param_grid={'C':[0.001,0.01,0.1,1,10,100], 'max_iter':[50,75,100,200,300,400,500,700]}  
log=RandomizedSearchCV(LogisticRegression(solver='lbfgs'),param_grid,cv=5)  
log.fit(X_train,y_train)  
log.score(X_train, y_train)
```

**OUT :**

```
0.927170868347339
```

10

## นำข้อมูลชุดทดสอบเข้าสู่โมเดล **LOGISTIC REGRESSION**

**IN :**

```
y_pred_log = log.predict(X_test)
print(y_pred_log)
```

**OUT :**

```
[0 0 1 1 0 1 1 1 1 1 1 0 0 1 1 0 1 0 0 1 0 1 0 1 0 0 0 1 0 0 1 0 1 0 0 0 1
0 0 1 0 0 1 0 0 1 0 1 1 1 1 0 1 1 1 0 0 1 0 0 1 0 0 1 0 0 0 0 1 1 1 0 1 0
0 0 0 0 1 1 1 0 1 0 0 1 1 1 1 0 0 1 0 1 1 0 0 1 1 0 1 0 0 0 0 0 0 0 1 0 0 0
1 0 1 0 0 1 0 0]
```



**IN :**

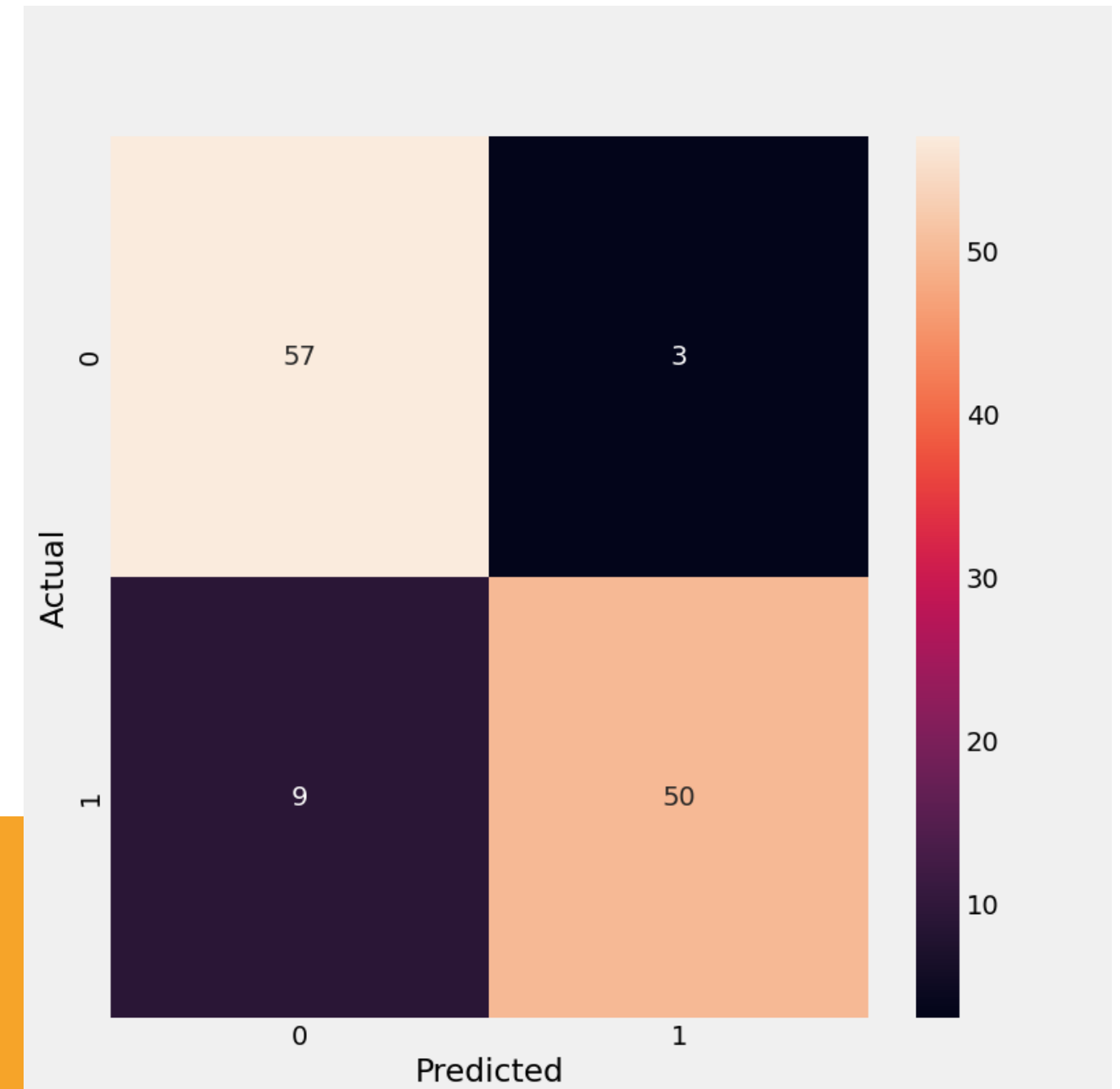
```
log.score(X_test,y_test)
```

**OUT :**

```
0.8907563025210085
```

# นำผลลัพธ์ที่ได้จากโมเดลมาทำ **CONFUSION MATRIX**

```
#Confusion Matrix
confusion_log=confusion_matrix(y_test,log.predict(X_test))
plt.figure(figsize=(8,8))
sns.heatmap(confusion_log,annot=True)
plt.xlabel("Predicted")
plt.ylabel("Actual")
plt.show()
```



# วิธีนำเข้าข้อมูลส่วนสร้างระบบและทดสอบระบบ

```
## Model Building
# Logistic Regression
param_grid={'C':[0.001,0.01,0.1,1,10,100], 'max_iter':[50,75,100,200,300,400,500,700]}
log=RandomizedSearchCV(LogisticRegression(solver='lbfgs'),param_grid,cv=5)
log.fit(X_train,y_train)

y_pred_log = log.predict(X_test)

#Confusion Matrix
confusion_log=confusion_matrix(y_test,log.predict(X_test))
plt.figure(figsize=(8,8))
sns.heatmap(confusion_log,annot=True)
plt.xlabel("Predicted")
plt.ylabel("Actual")
plt.show()
```

# ตัวอย่างข้อมูลทดสอบและผลการทดสอบ

```
print(X_test.head())
```

	GENDER	AGE	SMOKING	YELLOW_FINGERS	ANXIETY	PEER_PRESSURE	CHRONIC DISEASE	FATIGUE	ALLERGY	WHEEZING	ALCOHOL CONSUMING	COUGHING	SHORTNESS OF BREATH	SWALLOWING DIFFICULTY	CHEST PAIN
129	0	61	1	0	0	0	1	1	1	0	0	0	1	0	0
227	0	49	0	1	1	0	0	0	0	0	0	1	0	0	0
133	0	56	1	1	1	0	0	1	1	0	0	0	1	0	1
100	1	64	1	0	0	0	0	1	1	1	1	1	1	0	1
184	1	55	1	0	0	0	0	1	0	0	0	0	0	0	0

```
print(y_test.values)
```

```
[1 1 1 1 1 1 1 1 1 1 1 1 0 0 1 1 0 1 0 0 1 1 1 0 1 0 0 0 0 0 1 0 1 1 0 0 1
 0 0 1 0 0 1 0 0 1 0 1 1 1 1 0 1 1 1 1 0 1 0 0 1 0 0 1 0 0 0 0 0 1 1 0 0 0
 0 0 0 1 0 1 1 0 1 0 0 1 1 1 1 0 0 1 0 1 1 1 0 1 1 0 1 0 0 1 0 0 1 1 0 0 0
 1 0 1 0 0 1 0 0]
```





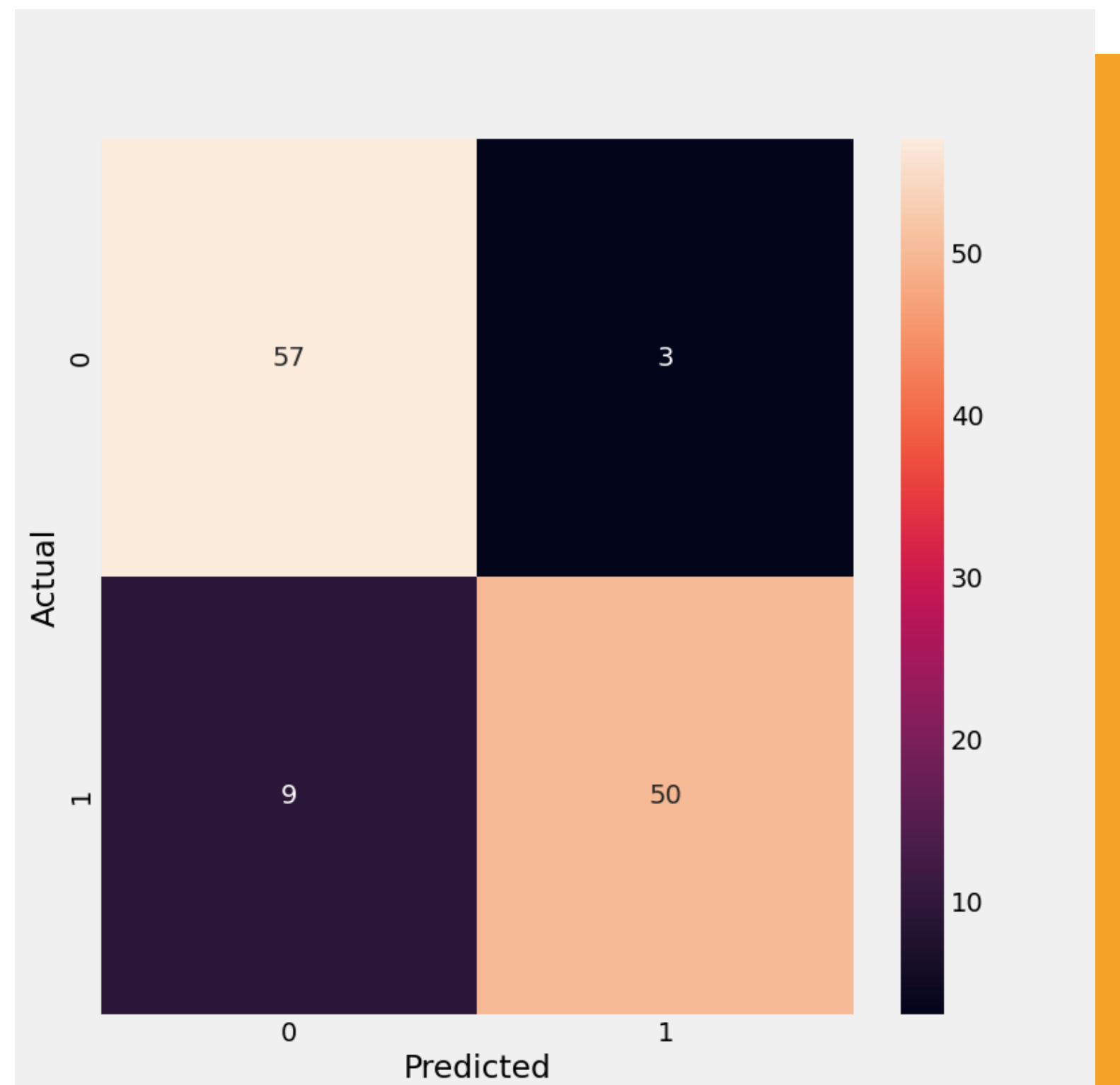
# ตัวอย่างข้อมูลทดสอบและผลการทดสอบ

## Confusion Matrix

	Actually Positive (1)	Actually Negative (0)
Predicted Positive (1)	True Positives (TPs)	False Positives (FPs)
Predicted Negative (0)	False Negatives (FNs)	True Negatives (TNs)

```
log.score(X_test,y_test)
```

0.8907563025210085



# แนวทางในการพัฒนาต่อ

สามารถใช้หลักการทางคณิตศาสตร์อื่นๆในการทำนายได้ เช่น **SUPPORT VECTOR MACHINE** หรือ **LGBM CLASSIFIER**

ซึ่งอาจจะมีค่าความแม่นยำในการทำนายมากกว่าใช้ **LOGISTIC REGRESSION**



LCP Application

### Lung Cancer Prediction Application


Input Your Data

Gender  
☐ Male ☒ Female

Age

Smoking <input type="checkbox"/> Yes	Chronic Disease <input type="checkbox"/> Yes	Alcohol Consuming <input type="checkbox"/> Yes
Yellow Fingers <input type="checkbox"/> Yes	Fatigue <input type="checkbox"/> Yes	Coughing <input type="checkbox"/> Yes
Anxiety <input type="checkbox"/> Yes	Allergy <input type="checkbox"/> Yes	Shortness of Breath <input type="checkbox"/> Yes
Peer Pressure <input type="checkbox"/> Yes	Wheezing <input type="checkbox"/> Yes	Swallowing Difficulty <input type="checkbox"/> Yes
Chest Pain <input type="checkbox"/> Yes	<input type="button" value="Predict"/>	

Lung Cancer :



LCP Application

### Lung Cancer Prediction Application


Input Your Data

Gender  
☒ Male ☐ Female

Age

Smoking <input checked="" type="checkbox"/> Yes	Chronic Disease <input type="checkbox"/> Yes	Alcohol Consuming <input checked="" type="checkbox"/> Yes
Yellow Fingers <input type="checkbox"/> Yes	Fatigue <input checked="" type="checkbox"/> Yes	Coughing <input checked="" type="checkbox"/> Yes
Anxiety <input checked="" type="checkbox"/> Yes	Allergy <input checked="" type="checkbox"/> Yes	Shortness of Breath <input checked="" type="checkbox"/> Yes
Peer Pressure <input type="checkbox"/> Yes	Wheezing <input checked="" type="checkbox"/> Yes	Swallowing Difficulty <input type="checkbox"/> Yes
Chest Pain <input checked="" type="checkbox"/> Yes	<input type="button" value="Predict"/>	

Lung Cancer : Yes



**MODEL ACCURACY : 89 %**

# MEMBERS

## G8 : RAINJERR X แมวอะชอบจับคุณ



บุริศ เสรีวัตตนะ  
64010462



ภัทรภรณ์ จันเดชา  
64010659



วัทธิกร เจริญกัลป์  
64010801



วิรุฬ สำเภาทอง  
64010815



# THANK YOU

G8 : RAINJERR X แมวอะชอบจับคุณ