SVM- Support Vector Machine

```
import seaborn as sns
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score,classification_report
import matplotlib.pyplot as plt
import numpy as np
```

```
data=pd.read_csv('survey lung cancer.csv')
```

```
data
```

	GENDER	AGE	SMOKING	YELLOW_FINGERS	ANXIETY	PEER_PRESSURE	CHRONIC DISEASE	FATIGUE	ALLERGY	WHEEZING	ALCOHOL CONSUMING	COUGHING	SHO OF
0	M	69	1	2	2	1	1	2	1	2	2	2	
1	M	74	2	1	1	1	2	2	2	1	1	1	
2	F	59	1	1	1	2	1	2	1	2	1	2	
3	M	63	2	2	2	1	1	1	1	1	2	1	
4	F	63	1	2	1	1	1	1	1	2	1	2	
...	...	...	...	...	...	...	...	...	...	...	...	...	...
304	F	56	1	1	1	2	2	2	1	1	2	2	
305	M	70	2	1	1	1	1	2	2	2	2	2	
306	M	58	2	1	1	1	1	1	2	2	2	2	
307	M	67	2	1	2	1	1	2	2	1	2	2	
308	M	62	1	1	1	2	1	2	2	2	2	1	

309 rows × 16 columns

Next steps: Generate code with data | New interactive sheet

```
data.head()
```

	GENDER	AGE	SMOKING	YELLOW_FINGERS	ANXIETY	PEER_PRESSURE	CHRONIC DISEASE	FATIGUE	ALLERGY	WHEEZING	ALCOHOL CONSUMING	COUGHING	SHORT OF BR
0	M	69	1	2	2	1	1	2	1	2	2	2	
1	M	74	2	1	1	1	2	2	2	1	1	1	
2	F	59	1	1	1	2	1	2	1	2	1	2	
3	M	63	2	2	2	1	1	1	1	1	2	1	
4	F	63	1	2	1	1	1	1	1	2	1	2	

Next steps: Generate code with data | New interactive sheet

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 309 entries, 0 to 308
Data columns (total 16 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   GENDER                 309 non-null    object
 1   AGE                    309 non-null    int64
 2   SMOKING                309 non-null    int64
 3   YELLOW_FINGERS         309 non-null    int64
 4   ANXIETY                309 non-null    int64
 5   PEER_PRESSURE          309 non-null    int64
 6   CHRONIC DISEASE        309 non-null    int64
 7   FATIGUE                309 non-null    int64
 8   ALLERGY                309 non-null    int64
 9   WHEEZING               309 non-null    int64
 10  ALCOHOL CONSUMING      309 non-null    int64
 11  COUGHING               309 non-null    int64
 12  SHORTNESS OF BREATH    309 non-null    int64
 13  SWALLOWING DIFFICULTY  309 non-null    int64
 14  CHEST PAIN             309 non-null    int64
 15  LUNG_CANCER            309 non-null    object
```

```
dtypes: int64(14), object(2)
memory usage: 38.8+ KB
```

```
data.describe()
```

	AGE	SMOKING	YELLOW_FINGERS	ANXIETY	PEER_PRESSURE	CHRONIC DISEASE	FATIGUE	ALLERGY	WHEEZING	ALCOHO CONSUMI
count	309.000000	309.000000	309.000000	309.000000	309.000000	309.000000	309.000000	309.000000	309.000000	309.0000(
mean	62.673139	1.563107	1.569579	1.498382	1.501618	1.504854	1.673139	1.556634	1.556634	1.5566:
std	8.210301	0.496806	0.495938	0.500808	0.500808	0.500787	0.469827	0.497588	0.497588	0.4975
min	21.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.0000(
25%	57.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.0000(
50%	62.000000	2.000000	2.000000	1.000000	2.000000	2.000000	2.000000	2.000000	2.000000	2.0000(
75%	69.000000	2.000000	2.000000	2.000000	2.000000	2.000000	2.000000	2.000000	2.000000	2.0000(
max	87.000000	2.000000	2.000000	2.000000	2.000000	2.000000	2.000000	2.000000	2.000000	2.0000(

```
data.isnull()
```

	GENDER	AGE	SMOKING	YELLOW_FINGERS	ANXIETY	PEER_PRESSURE	CHRONIC DISEASE	FATIGUE	ALLERGY	WHEEZING	ALCOHOL CONSUMING	COUGHING	SI OI
0	False	False	False	False	False	False	False	False	False	False	False	False	
1	False	False	False	False	False	False	False	False	False	False	False	False	
2	False	False	False	False	False	False	False	False	False	False	False	False	
3	False	False	False	False	False	False	False	False	False	False	False	False	
4	False	False	False	False	False	False	False	False	False	False	False	False	
...	...	...	...	...	...	...	...	...	...	...	...	...	...
304	False	False	False	False	False	False	False	False	False	False	False	False	
305	False	False	False	False	False	False	False	False	False	False	False	False	
306	False	False	False	False	False	False	False	False	False	False	False	False	
307	False	False	False	False	False	False	False	False	False	False	False	False	
308	False	False	False	False	False	False	False	False	False	False	False	False	

309 rows × 16 columns

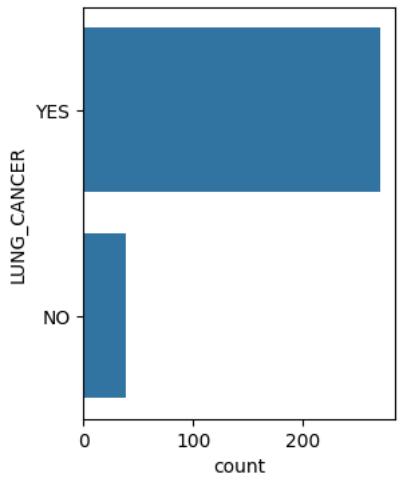
```
data.isnull().sum()
```

	0
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

```
plt.figure(figsize=(3,4))
sns.countplot(data['LUNG_CANCER'])
```

```
<Axes: xlabel='count', ylabel='LUNG_CANCER'>
```




```
data=data.drop('ANXIETY',axis=1)
```

```
data
```

	GENDER	AGE	SMOKING	YELLOW_FINGERS	PEER_PRESSURE	CHRONIC DISEASE	FATIGUE	ALLERGY	WHEEZING	ALCOHOL CONSUMING	COUGHING	SHORTNESS OF BREATH	S D
0	M	69	1	2	1	1	2	1	2	2	2	2	
1	M	74	2	1	1	2	2	2	1	1	1	2	
2	F	59	1	1	2	1	2	1	2	1	2	2	
3	M	63	2	2	1	1	1	1	1	2	1	1	
4	F	63	1	2	1	1	1	1	2	1	2	2	
...	...	...	...	...	...	...	...	...	...	...	...	...	
304	F	56	1	1	2	2	2	1	1	2	2	2	
305	M	70	2	1	1	1	2	2	2	2	2	2	
306	M	58	2	1	1	1	1	2	2	2	2	1	
307	M	67	2	1	1	1	2	2	1	2	2	2	
308	M	62	1	1	2	1	2	2	2	2	1	1	

309 rows × 15 columns

Next steps: Generate code with data | New interactive sheet

```
data.nunique()
```

	0
GENDER	2
AGE	39
SMOKING	2
YELLOW_FINGERS	2
PEER_PRESSURE	2
CHRONIC DISEASE	2
FATIGUE	2
ALLERGY	2
WHEEZING	2
ALCOHOL CONSUMING	2
COUGHING	2
SHORTNESS OF BREATH	2
SWALLOWING DIFFICULTY	2
CHEST PAIN	2
LUNG_CANCER	2

dtype: int64

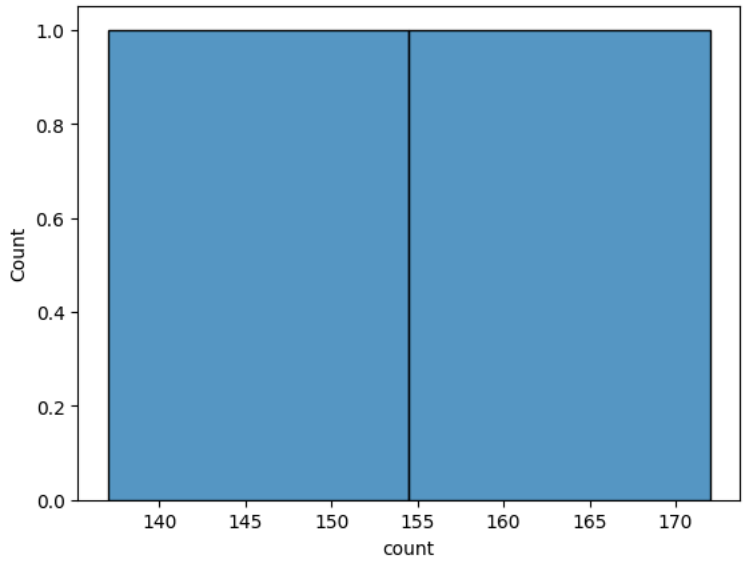
```
data['WHEEZING'].value_counts()
```

WHEEZING	count
2	172
1	137

dtype: int64

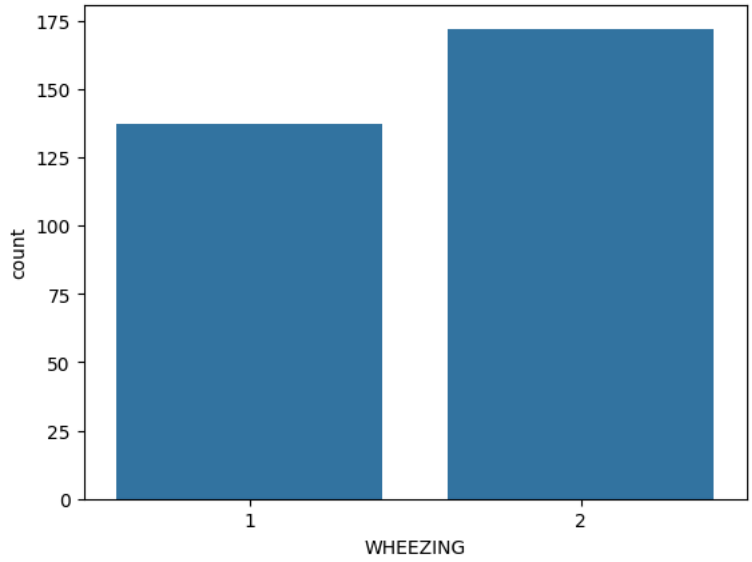
```
sns.histplot(data['WHEEZING'].value_counts())
```

```
<Axes: xlabel='count', ylabel='Count'>
```



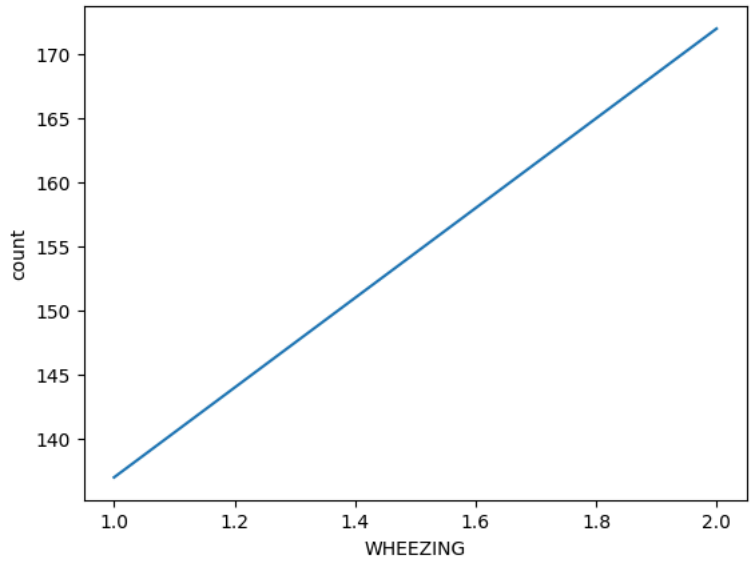

```
sns.barplot(data['WHEEZING'].value_counts())
```

```
<Axes: xlabel='WHEEZING', ylabel='count'>
```




```
sns.lineplot(data['WHEEZING'].value_counts())
```

```
<Axes: xlabel='WHEEZING', ylabel='count'>
```




```
data.columns
```

```
Index(['GENDER', 'AGE', 'SMOKING', 'YELLOW_FINGERS', 'PEER_PRESSURE',
       'CHRONIC DISEASE', 'FATIGUE ', 'ALLERGY ', 'WHEEZING',
       'ALCOHOL CONSUMING', 'COUGHING', 'SHORTNESS OF BREATH',
       'SWALLOWING DIFFICULTY', 'CHEST PAIN', 'LUNG_CANCER'],
      dtype='object')
```

```
x=[['Age'], 'YELLOW_FINGERS', 'COUGHING', 'ALLERGY', 'CHEST PAIN', 'LUNG_CANCER']
y=['Outcome']
```

```
data.notnull()
```

	GENDER	AGE	SMOKING	YELLOW_FINGERS	PEER_PRESSURE	CHRONIC DISEASE	FATIGUE	ALLERGY	WHEEZING	ALCOHOL CONSUMING	COUGHING	SHORTNESS OF BREATH
0	True	True	True	True	True	True	True	True	True	True	True	True
1	True	True	True	True	True	True	True	True	True	True	True	True
2	True	True	True	True	True	True	True	True	True	True	True	True
3	True	True	True	True	True	True	True	True	True	True	True	True
4	True	True	True	True	True	True	True	True	True	True	True	True
...	...	...	...	...	...	...	...	...	...	...	...	...
304	True	True	True	True	True	True	True	True	True	True	True	True
305	True	True	True	True	True	True	True	True	True	True	True	True
306	True	True	True	True	True	True	True	True	True	True	True	True
307	True	True	True	True	True	True	True	True	True	True	True	True
308	True	True	True	True	True	True	True	True	True	True	True	True

309 rows × 15 columns

```
data.fillna(4)
```

	GENDER	AGE	SMOKING	YELLOW_FINGERS	PEER_PRESSURE	CHRONIC DISEASE	FATIGUE	ALLERGY	WHEEZING	ALCOHOL CONSUMING	COUGHING	SHORTNESS OF BREATH	S D
0	M	69	1	2	1	1	2	1	2	2	2	2	
1	M	74	2	1	1	2	2	2	1	1	1	2	
2	F	59	1	1	2	1	2	1	2	1	2	2	
3	M	63	2	2	1	1	1	1	1	2	1	1	
4	F	63	1	2	1	1	1	1	2	1	2	2	
...	...	...	...	...	...	...	...	...	...	...	...	...	
304	F	56	1	1	2	2	2	1	1	2	2	2	
305	M	70	2	1	1	1	2	2	2	2	2	2	
306	M	58	2	1	1	1	1	2	2	2	2	1	
307	M	67	2	1	1	1	2	2	1	2	2	2	
308	M	62	1	1	2	1	2	2	2	2	1	1	

309 rows × 15 columns

```
data.replace()
```

```
/tmp/ipython-input-295544180.py:1: FutureWarning: DataFrame.replace without 'value' and with non-dict-like 'to_replace' is depr
  data.replace()
```

	GENDER	AGE	SMOKING	YELLOW_FINGERS	PEER_PRESSURE	CHRONIC DISEASE	FATIGUE	ALLERGY	WHEEZING	ALCOHOL CONSUMING	COUGHING	SHORTNESS OF BREATH	S D
0	M	69	1	2	1	1	2	1	2	2	2	2	
1	M	74	2	1	1	2	2	2	1	1	1	2	
2	F	59	1	1	2	1	2	1	2	1	2	2	
3	M	63	2	2	1	1	1	1	1	2	1	1	
4	F	63	1	2	1	1	1	1	2	1	2	2	
...	...	...	...	...	...	...	...	...	...	...	...	...	
304	F	56	1	1	2	2	2	1	1	2	2	2	
305	M	70	2	1	1	1	2	2	2	2	2	2	
306	M	58	2	1	1	1	1	2	2	2	2	1	
307	M	67	2	1	1	1	2	2	1	2	2	2	
308	M	62	1	1	2	1	2	2	2	2	1	1	

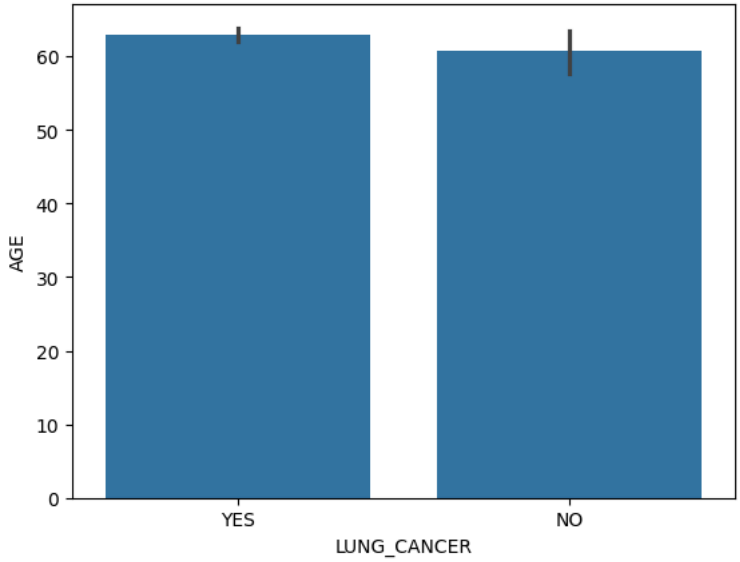
309 rows × 15 columns

```
sns.scatterplot(data['LUNG_CANCER'])
```

```
<Axes: ylabel='LUNG_CANCER'>
```

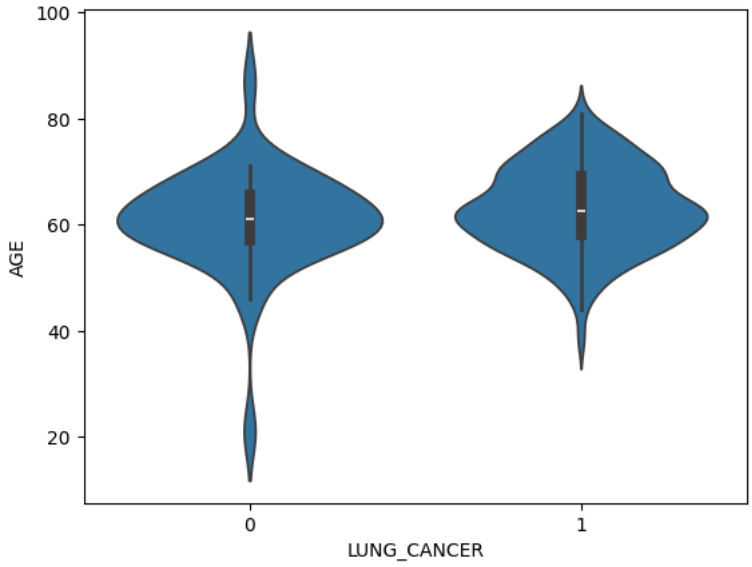

```
sns.barplot(x='LUNG_CANCER',y='AGE',data=data)
```

```
<Axes: xlabel='LUNG_CANCER', ylabel='AGE'>
```



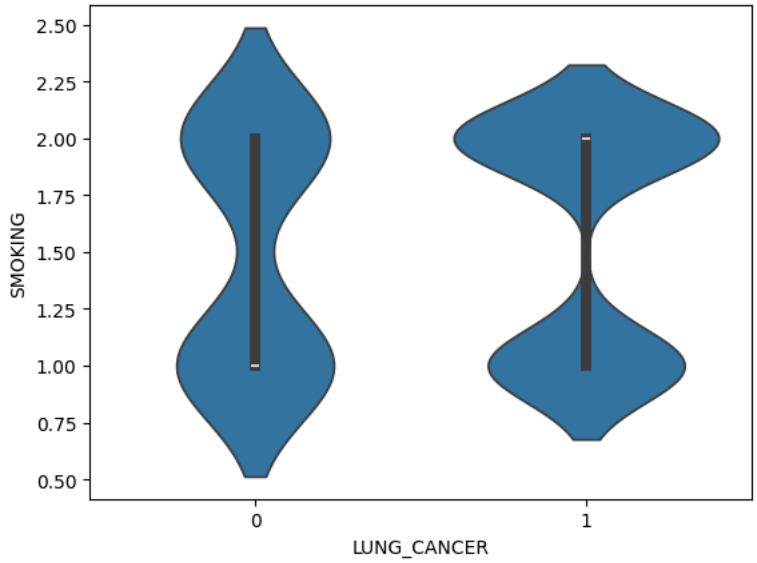

```
sns.violinplot(x='LUNG_CANCER', y='AGE',data=data)
```

```
<Axes: xlabel='LUNG_CANCER', ylabel='AGE'>
```



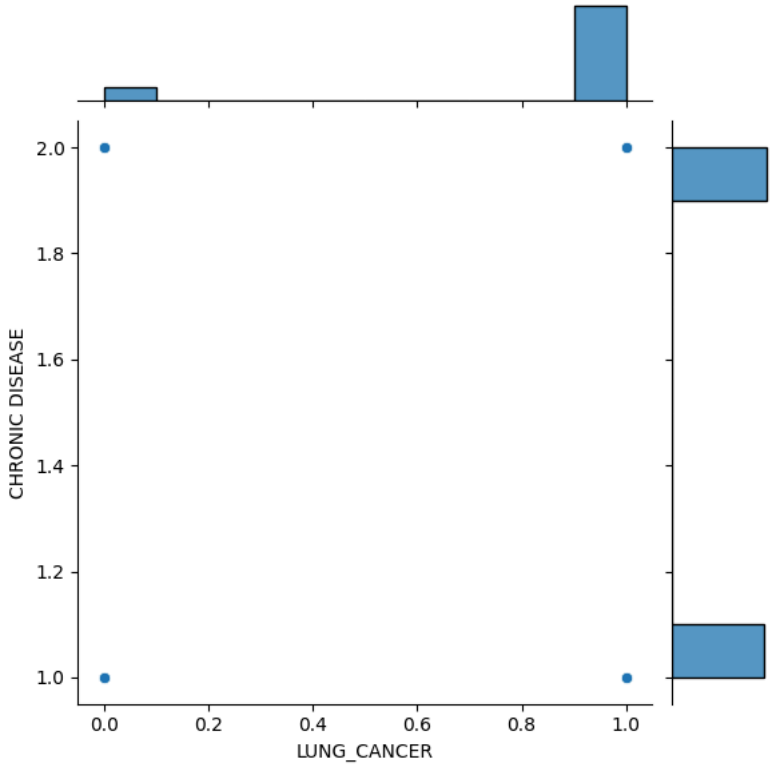

```
sns.violinplot(x='LUNG_CANCER', y='SMOKING',data=data)
```

```
<Axes: xlabel='LUNG_CANCER', ylabel='SMOKING'>
```



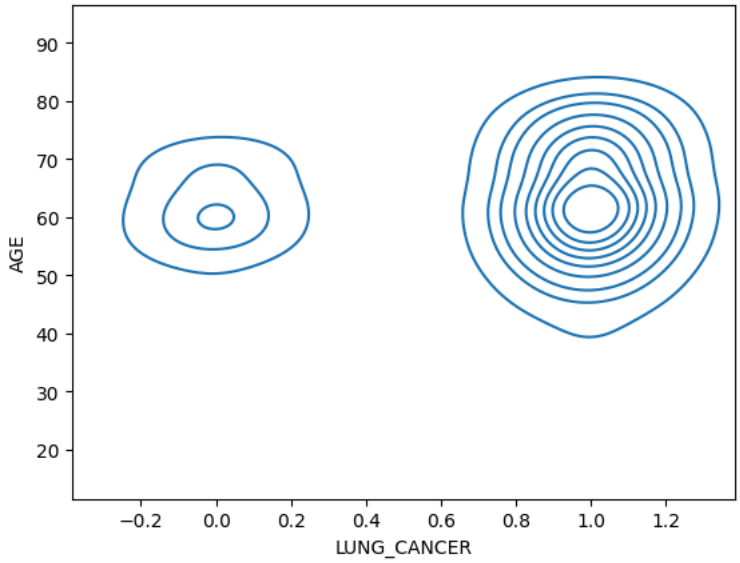

```
sns.jointplot(x='LUNG_CANCER', y='CHRONIC DISEASE',data=data)
```

```
<seaborn.axisgrid.JointGrid at 0x7822d820b980>
```




```
sns.kdeplot(x='LUNG_CANCER', y='AGE',data=data)
```

```
<Axes: xlabel='LUNG_CANCER', ylabel='AGE'>
```




```
data.columns
```

```
Index(['GENDER', 'AGE', 'SMOKING', 'YELLOW_FINGERS', 'PEER_PRESSURE',
       'CHRONIC DISEASE', 'FATIGUE ', 'ALLERGY ', 'WHEEZING',
       'ALCOHOL CONSUMING', 'COUGHING', 'SHORTNESS OF BREATH',
       'SWALLOWING DIFFICULTY', 'CHEST PAIN', 'LUNG_CANCER'],
      dtype='object')
```

```
data.head(1)
```

	GENDER	AGE	SMOKING	YELLOW_FINGERS	PEER_PRESSURE	CHRONIC DISEASE	FATIGUE	ALLERGY	WHEEZING	ALCOHOL CONSUMING	COUGHING	SHORTNESS OF BREATH	SWA DIF
0	M	69	1	2	1	1	2	1	2	2	2	2	

Next steps: Generate code with data | New interactive sheet

```
from sklearn.preprocessing import LabelEncoder
le=LabelEncoder()
data['GENDEER']=le.fit_transform(data['GENDER'])
data['LUNG_CANCER']=le.fit_transform(data['LUNG_CANCER'])
```

```
data
```

	GENDER	AGE	SMOKING	YELLOW_FINGERS	PEER_PRESSURE	CHRONIC DISEASE	FATIGUE	ALLERGY	WHEEZING	ALCOHOL CONSUMING	COUGHING	SHORTNESS OF BREATH	S D
0	M	69	1	2	1	1	2	1	2	2	2	2	
1	M	74	2	1	1	2	2	2	1	1	1	2	
2	F	59	1	1	2	1	2	1	2	1	2	2	
3	M	63	2	2	1	1	1	1	1	2	1	1	
4	F	63	1	2	1	1	1	1	2	1	2	2	
...	...	...	...	...	...	...	...	...	...	...	...	...	...
304	F	56	1	1	2	2	2	1	1	2	2	2	
305	M	70	2	1	1	1	2	2	2	2	2	2	
306	M	58	2	1	1	1	1	2	2	2	2	1	
307	M	67	2	1	1	1	2	2	1	2	2	2	
308	M	62	1	1	2	1	2	2	2	2	1	1	

309 rows × 16 columns

Next steps: Generate code with data | New interactive sheet

```
x=data.drop(['LUNG_CANCER', 'GENDER'],axis=1)
y=data['LUNG_CANCER']
```

x

	AGE	SMOKING	YELLOW_FINGERS	PEER_PRESSURE	CHRONIC DISEASE	FATIGUE	ALLERGY	WHEEZING	ALCOHOL CONSUMING	COUGHING	SHORTNESS OF BREATH	SWALLOWIN DIFFICULT
0	69	1	2	1	1	2	1	2	2	2	2	
1	74	2	1	1	2	2	2	1	1	1	2	
2	59	1	1	2	1	2	1	2	1	2	2	
3	63	2	2	1	1	1	1	1	2	1	1	
4	63	1	2	1	1	1	1	2	1	2	2	
...	...	...	...	...	...	...	...	...	...	...	...	...
304	56	1	1	2	2	2	1	1	2	2	2	
305	70	2	1	1	1	2	2	2	2	2	2	
306	58	2	1	1	1	1	2	2	2	2	1	
307	67	2	1	1	1	2	2	1	2	2	2	
308	62	1	1	2	1	2	2	2	2	1	1	

309 rows × 14 columns

Next steps: Generate code with x | New interactive sheet

```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,random_state=42)
```

```
from sklearn.preprocessing import StandardScaler
sc=StandardScaler()
x_train=sc.fit_transform(x_train)
x_test=sc.fit_transform(x_test)
```

```
from sklearn.linear_model import LogisticRegression
lr=LogisticRegression()
lr.fit(x_train,y_train)
```

▾ LogisticRegression

```
LogisticRegression()
```

```
from sklearn.svm import SVC
model=SVC()
model.fit(x_train,y_train)
```

▾ SVC

```
SVC()
```

```
model.score(x_train,y_train)*100, model.score(x_test,y_test)*100
```

```
(94.73684210526315, 93.54838709677419)
```

```
y_predict=model.predict(x_test)
```

```
#Creating the confusion matrix
from sklearn.metrics import confusion_matrix
cm=confusion_matrix(y_test,y_predict)
```

```
cm
```

```
array([[ 1,  1],
       [ 3, 57]])
```

```
from sklearn.metrics import accuracy_score, classification_report
print("Accuracy:", accuracy_score(y_test,y_predict))
print("\nClassification Report:\n",classification_report(y_test,y_predict))
```

```
Accuracy: 0.9354838709677419

Classification Report:
              precision    recall  f1-score   support

           0       0.25      0.50      0.33         2
```

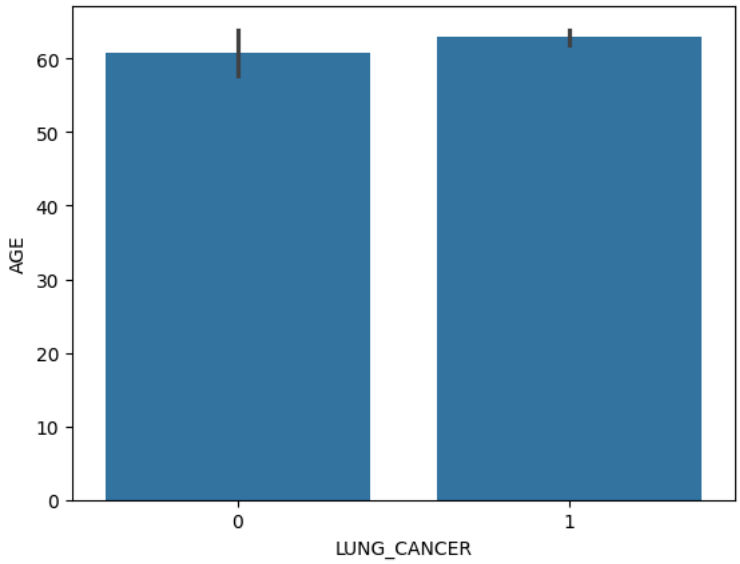
```
           1       0.98      0.95      0.97        60

    accuracy                           0.94        62
   macro avg       0.62      0.72      0.65        62
weighted avg       0.96      0.94      0.95        62
```

```
data.columns
```

```
Index(['GENDER', 'AGE', 'SMOKING', 'YELLOW_FINGERS', 'PEER_PRESSURE',
       'CHRONIC DISEASE', 'FATIGUE ', 'ALLERGY ', 'WHEEZING',
       'ALCOHOL CONSUMING', 'COUGHING', 'SHORTNESS OF BREATH',
       'SWALLOWING DIFFICULTY', 'CHEST PAIN', 'LUNG_CANCER', 'GENDEER'],
      dtype='object')
```

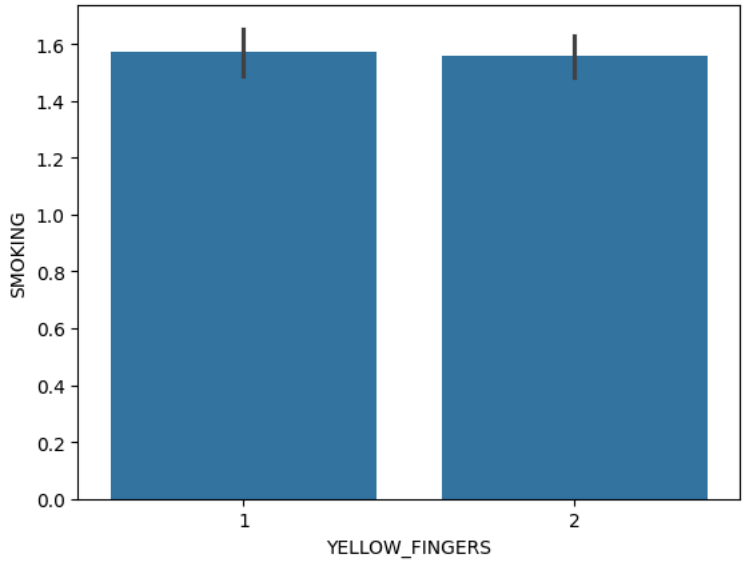
```
sns.barplot(x='LUNG_CANCER',y='AGE',data=data)
```

```
<Axes: xlabel='LUNG_CANCER', ylabel='AGE'>
```



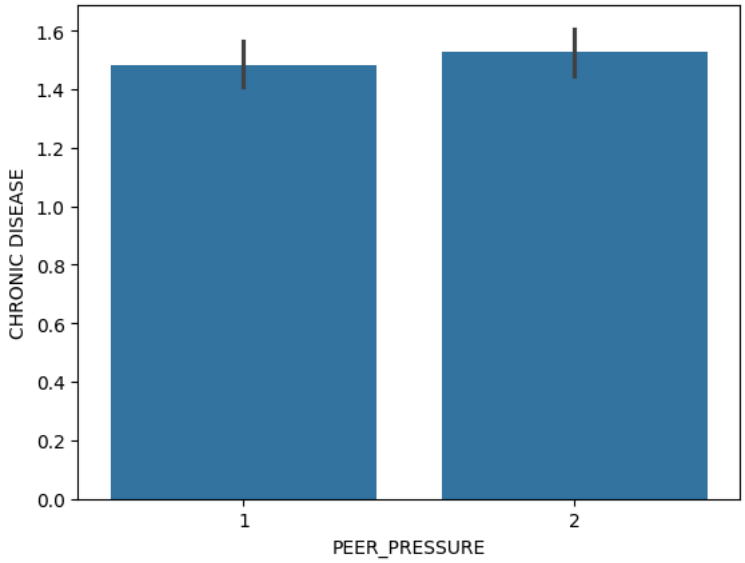

```
sns.barplot(x='YELLOW_FINGERS', y='SMOKING',data=data)
```

```
<Axes: xlabel='YELLOW_FINGERS', ylabel='SMOKING'>
```



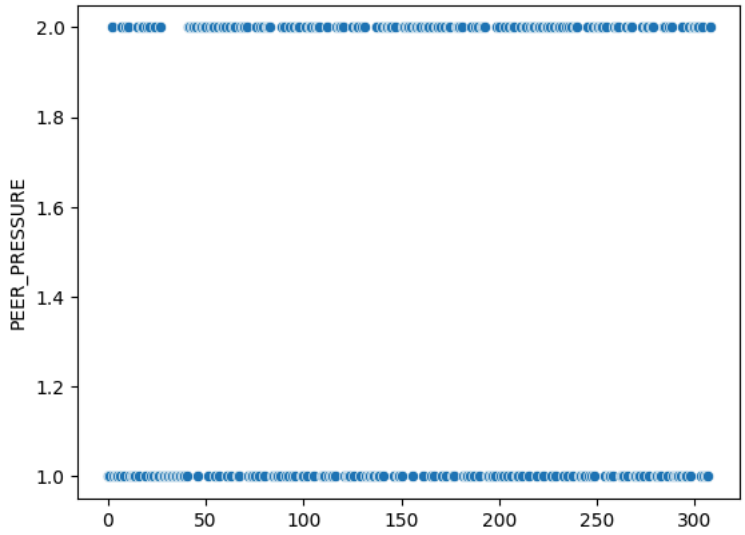

```
sns.barplot(x='PEER_PRESSURE',y='CHRONIC DISEASE',data=data)
```

```
<Axes: xlabel='PEER_PRESSURE', ylabel='CHRONIC DISEASE'>
```



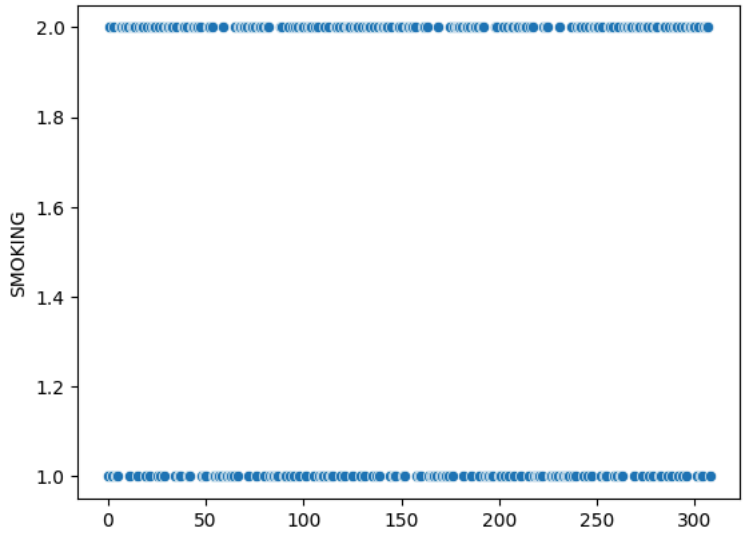

```
sns.scatterplot(data['PEER_PRESSURE'])
```

```
<Axes: ylabel='PEER_PRESSURE'>
```




```
sns.scatterplot(data['SMOKING'])
```

```
<Axes: ylabel='SMOKING'>
```




```
sns.scatterplot(data['CHRONIC DISEASE'])
```

```
<Axes: ylabel='CHRONIC DISEASE'>
```