

```
In [97]: import pandas as pd
import seaborn as sns

In [98]: df=pd.read_csv("titanic.csv")

In [99]: df.head()

Out[99]: PassengerId      Name  Pclass   Sex  Age  SibSp  Parch      Ticket    Fare  Cabin Embarked  Survived
0         1  Braund, Mr. Owen Harris     3   male  22.0    1    0      A/5 21171   7.2500   NaN        S         0
1         2  Cumings, Mrs. John Bradley (Florence Briggs Th...     1  female  38.0    1    0      PC 17599  71.2833   C85        C         1
2         3    Heikkinen, Miss. Laina     3  female  26.0    0    0  STON/O2. 3101282   7.9250   NaN        S         1
3         4  Futrelle, Mrs. Jacques Heath (Lily May Peel)     1  female  35.0    1    0      113803  53.1000  C123        S         1
4         5    Allen, Mr. William Henry     3   male  35.0    0    0      373450   8.0500   NaN        S         0

In [100]: df.drop(['PassengerId','Name','Pclass','Ticket','Parch','Cabin','Embarked','SibSp'],axis="columns",inplace=True)

In [101]: df.head() # various column are useless so we drop it

Out[101]:      Sex  Age  Fare  Survived
0   male  22.0   7.2500           0
1  female  38.0  71.2833           1
2  female  26.0   7.9250           1
3  female  35.0  53.1000           1
4   male  35.0   8.0500           0

In [102]: target=df.Survived
inputs=df.drop('Survived',axis=1) # we separte the target variable and dependent variable

In [103]: inputs.head()

Out[103]:      Sex  Age  Fare
0   male  22.0   7.2500
1  female  38.0  71.2833
2  female  26.0   7.9250
3  female  35.0  53.1000
4   male  35.0   8.0500

In [104]: target.head()

Out[104]: 0      0
1      1
2      1
3      1
4      0
Name: Survived, dtype: int64

In [105]: dummy=pd.get_dummies(inputs.Sex) # the sex contain the string value so we need numeric value so we create the dummy variable

In [106]: dummy

Out[106]:      female  male
0         0      1
1         1      0
2         1      0
3         1      0
4         0      1
...      ...      ...
886        0      1
887         1      0
888         1      0
889         0      1
890         0      1
891 rows x 2 columns

In [107]: inputs=pd.concat([inputs,dummy],axis=1)

In [108]: inputs

Out[108]:      Sex  Age  Fare  female  male
0   male  22.0   7.2500         0      1
1  female  38.0  71.2833         1      0
2  female  26.0   7.9250         1      0
3  female  35.0  53.1000         1      0
4   male  35.0   8.0500         0      1
...      ...      ...      ...      ...
886   male  27.0  13.0000         0      1
887  female  19.0  30.0000         1      0
888  female  NaN  23.4500         1      0
889   male  26.0  30.0000         0      1
890   male  32.0   7.7500         0      1
891 rows x 5 columns

In [109]: inputs.drop('Sex',axis="columns")

Out[109]:      Age  Fare  female  male
0   22.0   7.2500         0      1
1   38.0  71.2833         1      0
2   26.0   7.9250         1      0
3   35.0  53.1000         1      0
4   35.0   8.0500         0      1
...      ...      ...      ...
886  27.0  13.0000         0      1
887  19.0  30.0000         1      0
888  NaN  23.4500         1      0
889  26.0  30.0000         0      1
890  32.0   7.7500         0      1
891 rows x 4 columns

In [110]: na_value =inputs.isna().any()

In [111]: na_value

Out[111]: Sex      False
Age      True
Fare     False
female   False
male     False
dtype: bool

In [112]: # since the age column has any we need to fill the value with mean

In [113]: inputs.Age=inputs.Age.fillna(inputs.Age.mean())

In [114]: inputs

Out[114]:      Sex      Age  Fare  female  male
0   male  22.000000   7.2500         0      1
1  female  38.000000  71.2833         1      0
2  female  26.000000   7.9250         1      0
3  female  35.000000  53.1000         1      0
4   male  35.000000   8.0500         0      1
...      ...      ...      ...      ...
886   male  27.000000  13.0000         0      1
887  female  19.000000  30.0000         1      0
888  female  29.699118  23.4500         1      0
889   male  26.000000  30.0000         0      1
890   male  32.000000   7.7500         0      1
891 rows x 5 columns

In [115]: na_value =inputs.isna().any()

In [116]: na_value

Out[116]: Sex      False
Age      False
Fare     False
female   False
male     False
dtype: bool

In [117]: from sklearn.naive_bayes import GaussianNB

In [118]: model=GaussianNB()

In [119]: from sklearn.model_selection import train_test_split

In [120]: inputs.drop('Sex',axis=1,inplace=True)
x_train,x_test,y_train,y_test=train_test_split(inputs,target,test_size=0.2)

In [ ]:

In [ ]:

In [121]: model.fit(x_train,y_train)

Out[121]: GaussianNB
GaussianNB()

In [122]: model.score(x_test,y_test)

Out[122]: 0.7932960893854749

In [123]: model.predict(x_test[0:10])

Out[123]: array([0, 0, 1, 1, 0, 0, 0, 0, 1], dtype=int64)

In [125]: model.predict_proba(x_test[0:10])

Out[125]: array([[0.98071176, 0.01928824],
 [0.97724245, 0.02275755],
 [0.03803181, 0.96196819],
 [0.04266538, 0.95733462],
 [0.9485649 , 0.0514351 ],
 [0.9802145 , 0.0197855 ],
 [0.96372987, 0.03627013],
 [0.98027734, 0.01972266],
```

```
[0.94838101, 0.05161899],  
[0.83943908, 0.96056092]])
```

```
In [ ]: # it shows that probabilty of being dead is 0.98 and probability of being alive is 0.019
```

```
0.98
```

```
In [ ]: # it shows that probabilty of being dead is 0.98 and probability of being alive is 0.019
```