In [5]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

In [8]:
```
df = pd.read_csv(r'C:/Users/aziz/Desktop/Downloads/KNN/data.csv')
```

In [10]:
```
df.head()
```

Out[10]:

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mea
0	842302	M	17.99	10.38	122.80	1001.0	0.1184
1	842517	M	20.57	17.77	132.90	1326.0	0.0847
2	84300903	M	19.69	21.25	130.00	1203.0	0.1096
3	84348301	M	11.42	20.38	77.58	386.1	0.1425
4	84358402	M	20.29	14.34	135.10	1297.0	0.1003

5 rows × 33 columns

In [11]:
```
df.columns
```

Out[11]:
```
Index(['id', 'diagnosis', 'radius_mean', 'texture_mean', 'perimeter_mean',
       'area_mean', 'smoothness_mean', 'compactness_mean', 'concavity_mean',
       'concave points_mean', 'symmetry_mean', 'fractal_dimension_mean',
       'radius_se', 'texture_se', 'perimeter_se', 'area_se', 'smoothness_se',
       'compactness_se', 'concavity_se', 'concave points_se', 'symmetry_se',
       'fractal_dimension_se', 'radius_worst', 'texture_worst',
       'perimeter_worst', 'area_worst', 'smoothness_worst',
       'compactness_worst', 'concavity_worst', 'concave points_worst',
       'symmetry_worst', 'fractal_dimension_worst', 'Unnamed: 32'],
      dtype='object')
```

In [12]:
```
x = df.drop(['id', 'Unnamed: 32', 'diagnosis'], axis=1)
```

In [13]:
```
pd.set_option('display.max_columns', None)
```

In [17]:
```
y = df['diagnosis']
```

In [18]:
```
from sklearn.preprocessing import LabelEncoder
le = LabelEncoder()
le.fit(y)
le.classes_
s = le.transform(y)
y = s
```

In [20]:
```
from sklearn.model_selection import train_test_split
```

In [21]:
```
X_train, X_test, y_train, y_test = train_test_split(x, y, test_size = 0.2, rando
```

In [22]:
```
from sklearn import svm
```

In [23]:
```
from sklearn.neighbors import KNeighborsClassifier
```

In [24]:
```
knn = KNeighborsClassifier(n_neighbors = 3)
```

In [25]:
```
knn.fit(X_train, y_train)
```

Out[25]:
```
KNeighborsClassifier(algorithm='auto', leaf_size=30, metric='minkowski',
           metric_params=None, n_jobs=None, n_neighbors=3, p=2,
           weights='uniform')
```

In [26]:
```
knn.predict(X_test) [0:5]
```

Out[26]:
```
array([1, 1, 0, 1, 0])
```

In [27]:
```
print("KNN: ")
knn.score(X_test, y_test)
```

```
KNN:
```

Out[27]:
```
0.9298245614035088
```

In [29]:
```
model = svm.SVC()
model.fit(X_train, y_train)
```

```
C:\Users\aziz\Anaconda3\lib\site-packages\sklearn\svm\base.py:196: FutureWarnin
g: The default value of gamma will change from 'auto' to 'scale' in version 0.2
2 to account better for unscaled features. Set gamma explicitly to 'auto' or 's
cale' to avoid this warning.
  "avoid this warning.", FutureWarning)
```

Out[29]:
```
SVC(C=1.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovr', degree=3, gamma='auto_deprecated',
  kernel='rbf', max_iter=-1, probability=False, random_state=None,
  shrinking=True, tol=0.001, verbose=False)
```

In [30]:
```
model.score(X_test, y_test)
```

Out[30]:
```
0.6140350877192983
```