

In [2]:

```
!pip install bioinfokit
```

Collecting bioinfokit

Downloading bioinfokit-2.0.8.tar.gz (84 kB)

Requirement already satisfied: pandas in c:\users\ashraf\anaconda3\lib\site-packages (from bioinfokit) (1.3.4)

Requirement already satisfied: numpy in c:\users\ashraf\anaconda3\lib\site-packages (from bioinfokit) (1.20.3)

Requirement already satisfied: matplotlib in c:\users\ashraf\anaconda3\lib\site-packages (from bioinfokit) (3.4.3)

Requirement already satisfied: scipy in c:\users\ashraf\anaconda3\lib\site-packages (from bioinfokit) (1.7.1)

Requirement already satisfied: scikit-learn in c:\users\ashraf\anaconda3\lib\site-packages (from bioinfokit) (0.24.2)

Requirement already satisfied: seaborn in c:\users\ashraf\anaconda3\lib\site-packages (from bioinfokit) (0.11.2)

Collecting matplotlib-venn

Downloading matplotlib-venn-0.11.6.tar.gz (29 kB)

Collecting tabulate

Downloading tabulate-0.8.9-py3-none-any.whl (25 kB)

Requirement already satisfied: statsmodels in c:\users\ashraf\anaconda3\lib\site-packages (from bioinfokit) (0.12.2)

In [3]:

```
from pandas import read_csv
import pandas as pd

from sklearn.manifold import TSNE
from bioinfokit.visuz import cluster
```

In [4]:

```
# Load data
filename = 'C:/Users/Ashraf/Documents/Datafiles/TSNE_data.csv'
dataframe = pd.read_csv(filename)
```

In [5]:

dataframe

Out[5]:

	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	cc
0	M	17.99	10.38	122.80	1001.0	0.11840	
1	M	20.57	17.77	132.90	1326.0	0.08474	
2	M	19.69	21.25	130.00	1203.0	0.10960	
3	M	11.42	20.38	77.58	386.1	0.14250	
4	M	20.29	14.34	135.10	1297.0	0.10030	
...	
564	M	21.56	22.39	142.00	1479.0	0.11100	
565	M	20.13	28.25	131.20	1261.0	0.09780	
566	M	16.60	28.08	108.30	858.1	0.08455	
567	M	20.60	29.33	140.10	1265.0	0.11780	
568	B	7.76	24.54	47.92	181.0	0.05263	

569 rows × 31 columns

In [6]:

```
# Split-out validation dataset
array = dataframe.values
# separate array into input and output components
X = array[:,1:]
Y = array[:,0]
```

In [7]:

X

Out[7]:

```
array([[17.99, 10.38, 122.8, ..., 0.2654, 0.4601, 0.1189],
       [20.57, 17.77, 132.9, ..., 0.186, 0.275, 0.08902],
       [19.69, 21.25, 130.0, ..., 0.243, 0.3613, 0.08758],
       ...,
       [16.6, 28.08, 108.3, ..., 0.1418, 0.2218, 0.0782],
       [20.6, 29.33, 140.1, ..., 0.265, 0.4087, 0.124],
       [7.76, 24.54, 47.92, ..., 0.0, 0.2871, 0.07039]], dtype=object)
```

In [8]:

Y

Out[8]:

```
array(['M', 'M', 'M', 'M', 'M', 'M', 'M', 'M', 'M', 'M', 'M', 'M', 'M',
      'M', 'M', 'M', 'M', 'M', 'M', 'B', 'B', 'B', 'M', 'M', 'M', 'M',
      'M', 'M', 'M', 'M', 'M', 'M', 'M', 'M', 'M', 'M', 'M', 'M', 'B', 'M',
      'M', 'M', 'M', 'M', 'M', 'M', 'M', 'B', 'M', 'B', 'B', 'B', 'B',
      'B', 'M', 'M', 'B', 'M', 'M', 'B', 'B', 'B', 'B', 'M', 'B', 'M',
      'M', 'B', 'B', 'B', 'B', 'B', 'M', 'B', 'M', 'M', 'B', 'M', 'B', 'M',
      'M', 'B', 'B', 'B', 'M', 'M', 'B', 'M', 'M', 'M', 'B', 'B', 'B',
      'M', 'B', 'B', 'M', 'M', 'B', 'B', 'B', 'B', 'B', 'B', 'B', 'B',
      'M', 'M', 'M', 'B', 'M', 'M', 'B', 'B', 'B', 'M', 'B', 'B', 'M', 'B',
      'B', 'B', 'B', 'M', 'B', 'B', 'B', 'B', 'B', 'B', 'B', 'B', 'B',
      'M', 'B', 'B', 'B', 'B', 'B', 'M', 'M', 'B', 'M', 'B', 'B', 'M', 'M',
      'B', 'B', 'M', 'M', 'B', 'B', 'B', 'B', 'B', 'B', 'B', 'B', 'B',
      'M', 'B', 'B', 'B', 'B', 'B', 'M', 'M', 'M', 'M', 'M', 'M', 'M', 'M',
      'M', 'M', 'M', 'M', 'M', 'M', 'B', 'B', 'B', 'B', 'B', 'B', 'B', 'M',
      'B', 'M', 'B', 'B', 'M', 'B', 'B', 'B', 'M', 'B', 'M', 'M', 'B', 'B',
      'B', 'B', 'B', 'B', 'B', 'B', 'B', 'B', 'B', 'B', 'B', 'M', 'B',
      'B', 'M', 'B', 'M', 'B', 'B', 'B', 'B', 'B', 'B', 'B', 'B', 'B', 'B',
      'B', 'B', 'B', 'B', 'B', 'B', 'M', 'B', 'B', 'B', 'B', 'B', 'M', 'B',
      'B', 'B', 'B', 'B', 'B', 'B', 'M', 'B', 'B', 'B', 'B', 'B', 'M', 'B',
      'B', 'B', 'B', 'B', 'B', 'B', 'M', 'B', 'B', 'B', 'B', 'B', 'B', 'B',
      'M', 'B', 'B', 'B', 'B', 'B', 'B', 'B', 'B', 'B', 'B', 'B', 'M', 'B',
      'B', 'B', 'B', 'B', 'B', 'B', 'M', 'B', 'M', 'B', 'M', 'B', 'B', 'B',
      'B', 'B', 'B', 'B', 'B', 'B', 'B', 'B', 'B', 'B', 'B', 'B', 'B', 'B',
      'B', 'B', 'B', 'M', 'M', 'M', 'M', 'M', 'M', 'M', 'M', 'B'], dtype=object)
```

In [9]:

```
#TSNE visualization
from bioinfokit.visuz import cluster

data_tsne = TSNE(n_components=2).fit_transform(X)
cluster.tsneplot(score=data_tsne)
```

In [10]:

```
# get a list of categories
color_class = dataframe['diagnosis'].to_numpy()
cluster.tsneplot(score=data_tsne, colorlist=color_class, legendpos='upper right', legendanch
#Plot will be stored in the default directory
```

In [11]:

```
data_tsne
```

Out[11]:

```
array([[ -3.364533 , -30.947056 ],
       [ -2.0137036, -28.417938 ],
       [ -6.0225997, -25.734827 ],
       ...,
       [-18.157185 , -11.338934 ],
       [ -3.943487 , -27.31959  ],
       [ 34.673336 , -13.596828 ]], dtype=float32)
```

In [13]:

```
import os
os.getcwd()
```

Out[13]:

```
'C:\\Users\\Ashraf'
```

In []: