AIM: IMPLEMENTATION OF CLASSIFYING DATA USING NON-LINEAR KERNELS SUPPORT VECTOR MACHINES (SVMS).

THEORY:

"Support Vector Machine" (SVM) is a supervised machine learning algorithm which can be used for both classification or regression challenges. However, it is mostly used in classification problems. In the SVM algorithm, we plot each data item as a point in n-dimensional space (where n is number of features you have) with the value of each feature being the value of a particular coordinate. Then, we perform classification by finding the hyper-plane that differentiates the two classes very well . Support Vectors are simply the co-ordinates of individual observation. The SVM classifier is a frontier which best segregates the two classes (hyper plane/ line) In the SVM classifier, it is easy to have a linear hyper-plane between these two classes. But, another burning question which arises is, should we need to add this feature manually to have a hyper-plane. No, the SVM algorithm has a technique called the kernel trick. The SVM kernel is a function that takes low dimensional input space and transforms it to a higher dimensional space i.e. it converts not separable problem to separable problem. It is mostly useful in non linear separation problem. Simply put, it does some extremely complex data transformations, then finds out the process to separate the data based on the labels or outputs you've defined. In Python, scikit-learn is a widely used library for implementing machine learning algorithms. SVM is also available in the scikit-learn library and we follow the same structure for using it(Import library, object creation, fitting model and prediction)

Non-Linear SVM is used for non-linearly separated data, which means if a dataset cannot be classified by using a straight line, then such data is termed as non-linear data and classifier used is called as Non-linear SVM classifier.

SOURCE CODE:

1) **IMPORTING LIBRARIES:**

```
[1] import pandas as pd
    import numpy as np
    import matplotlib.pyplot as plt
```

2) **READING DATASET:**

```
data=pd.read_csv('apples_and_oranges.csv')
data.head(5)
```

	Weight	Size	Class
0	69	4.39	orange
1	69	4.21	orange
2	65	4.09	orange
3	72	5.85	apple
4	67	4.70	orange

3) SPLITTING DATASET INTO TRAIN & TEST DATA:

```
[3]  from sklearn.model_selection import train_test_split

     X = data.drop(['Class'], axis='columns')
     Y = data.drop(['Weight','Size'], axis='columns')

     X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.3)
```

4) BUILDING MODEL:

```
from sklearn.svm import SVC

model = SVC(kernel='rbf')
model.fit(X_train,Y_train)
```

```
/usr/local/lib/python3.7/dist-packa
  y = column_or_1d(y, warn=True)
SVC()
```

5) PREDICTION:

```
[6]  model.predict([[62,3.0]])
```

```
/usr/local/lib/python3.7/dist-pa
  "X does not have valid feature
array(['orange'], dtype=object)
```

```
[7]  model.predict([[74, 5.3]])
```

```
/usr/local/lib/python3.7/dist-p
  "X does not have valid featur
array(['apple'], dtype=object)
```

CONCLUSION:

From this practical, I have learned and implemented Non-Linear Support Vector Machine(SVM) in python.