**Activity 2**

**Classification with SVM, BP and MLR**

* **Git Repository**

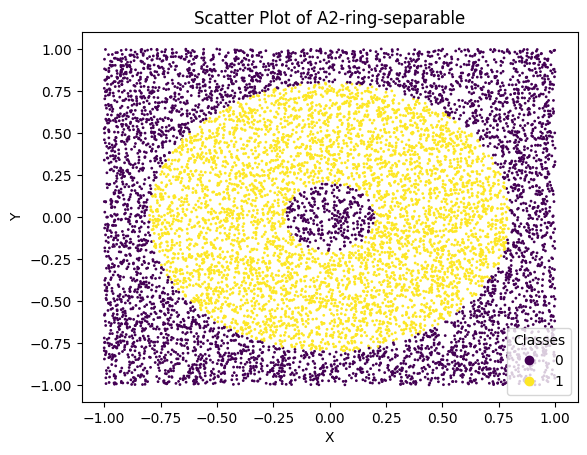
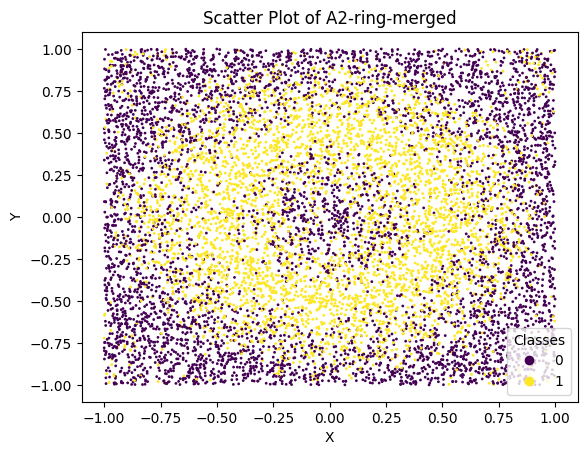
[https://github.com/YoussefEzz/Classification-SVM-BP-and-LR](https://github.com/YoussefEzz/Prediction-BP-and-LR)

* **Part 1 : Selecting and analyzing the datasets**

Since we do not want to give a priori more importance to some of the input variables w.r.t. the others, we should scale all of them to the same range of variation.

1. **Ring datasets A2-ring-merged.txt and A2-ring-separable.txt** analyzed in **Ring\_Datasets.ipynb**

since the two input variables(call them x and y) lie in the same range [-1.0, 1.0], No pre-processing is needed. Both Training sets have the same two input feature values but with different output values(class labels) such that the plot of A2-ring-merged shows that the class of points are emerged but the plot of A2-ring-separable shows that the class of points are separable

****

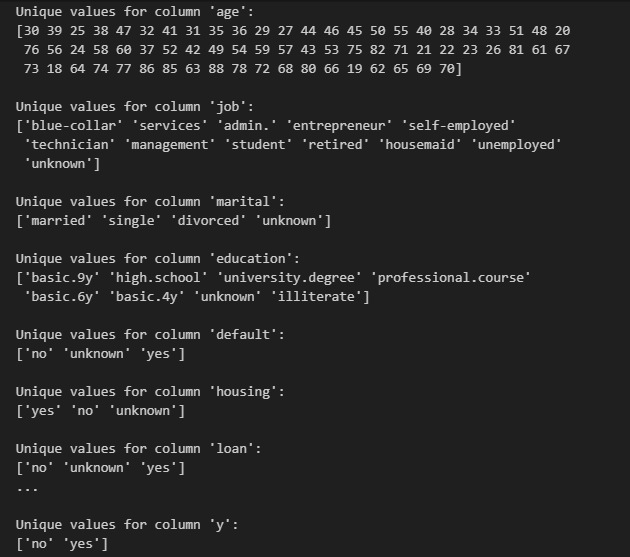
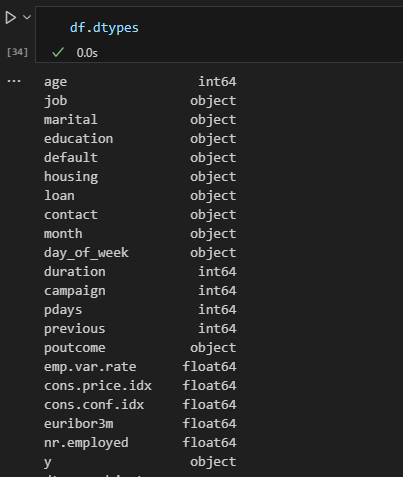
1. **Bank Dataset bank-additional.csv** analyzed in **Bank\_Datasets.ipynb**

the dataset contains 20 input variables and one output variable(y) ,so total 21 columns.

* + 10 numerical columns
    - 5 integer columns e.g. (age, duration)
    - 5 float columns e.g. (emp.var.rate, cons.price.idx)
  + 11 categorical values
    - 9 nominal columns(No particular order) e.g. (marital, education)
    - 2 ordinal columns(some ordered) e.g. (month, day\_of\_week)

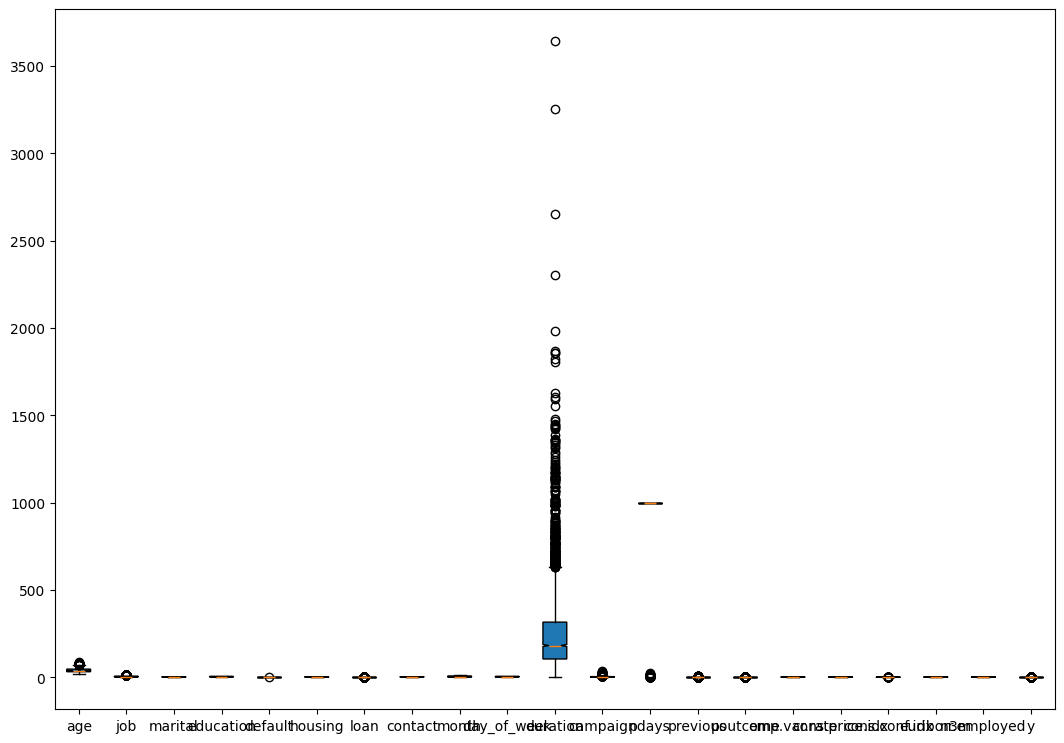
As per <https://www.kaggle.com/code/pythonafroz/categorical-to-numerical-encoding-methods>

Unique values for categorical columns datatype for each column



Steps followed to preprocess the data

1. Drop rows with missing information tagged as “**unknown**” in any column
2. **categorical columns** : encode as ordinal using category\_encoders
3. **numerical float columns** : encode by scaling from -1 to 1
4. **output class label column** : encode by replacing yes with 1 and no with 0
5. **integer columns** like age, duration are not preprocessed



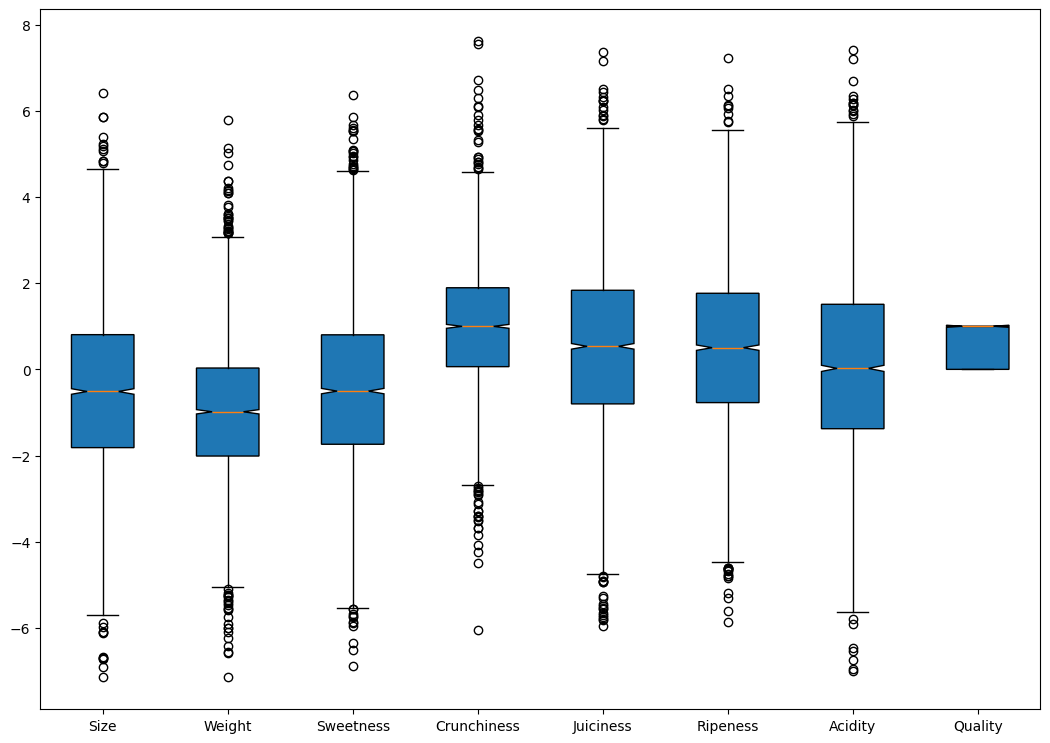
1. **dataset from the Internet** : **apple\_quality.csv** analyzed in **apple\_quality\_dataset.ipynb**

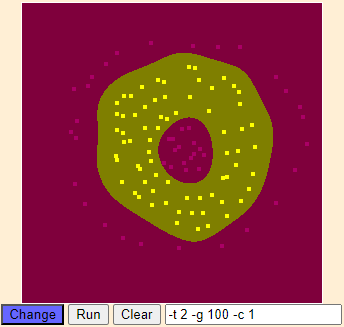
**URL :** <https://www.kaggle.com/datasets/nelgiriyewithana/apple-quality>

Note that size and weight are already negative values in the original dataset might be due to incorrect data entry as discussed in link <https://www.kaggle.com/datasets/nelgiriyewithana/apple-quality/discussion>

Steps followed to preprocess the data

1. Drop unnecessary column **A\_id**
2. Drop **nan** values
3. change the type of **Acidity** column from object to float
4. replace **good** by 1 and **bad** by 0

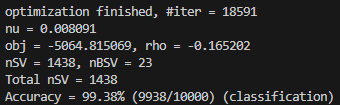


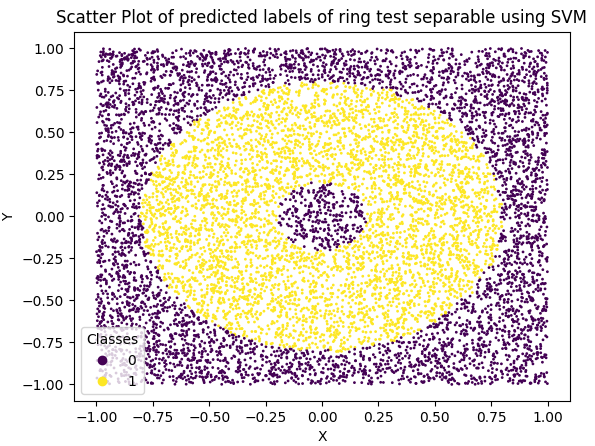
**Part 2 : Classification problem**

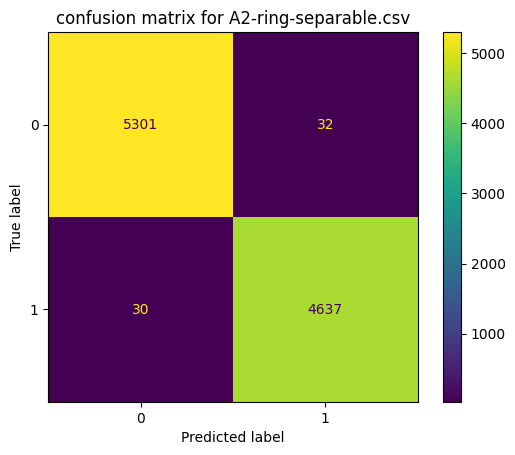
1. **SVM classification model for Dataset in**
2. **SVM\_ring\_separable.ipynb**

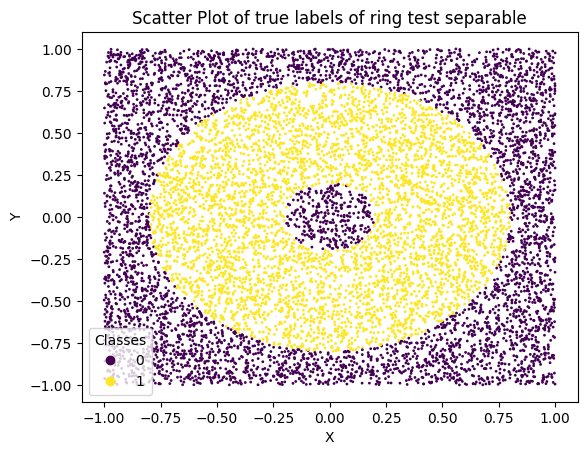
**Libsvm** library is used to train the model using the training set of **A2-ring-separable.txt** with the default parameter **-t** for kernel type 2 -- radial basis function: **exp(-gamma\*|u-v|^2)**, but gamma variable  **-g**  had to be tuned to be 50 – 200 (default 1/num\_features) to maintain accuracy above **99%**.

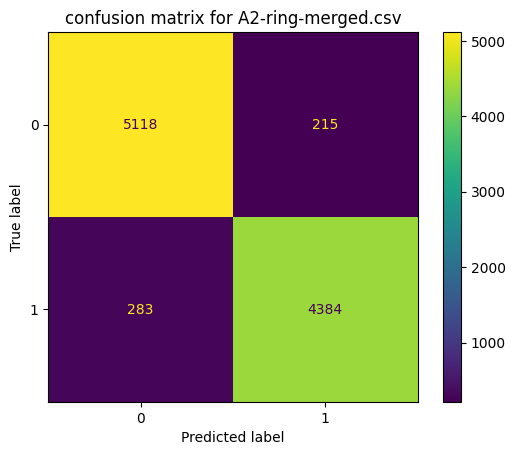
Graphic Interface applet on LiBsvm website [https://www.csie.ntu.edu.tw/~cjlin/libsvm/](https://www.csie.ntu.edu.tw/~cjlin/libsvm/%20%20) helped select the value of gamma to obtain below results.

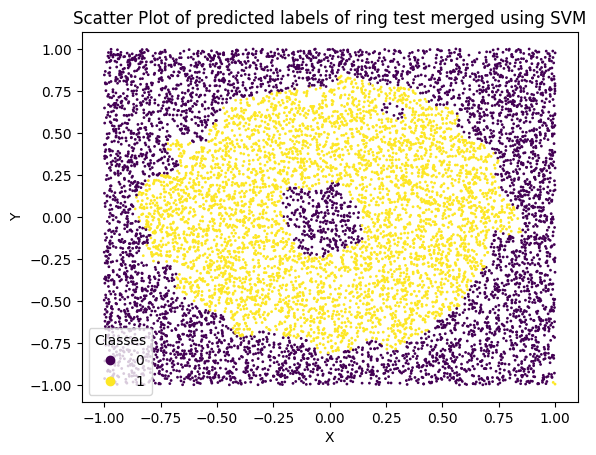


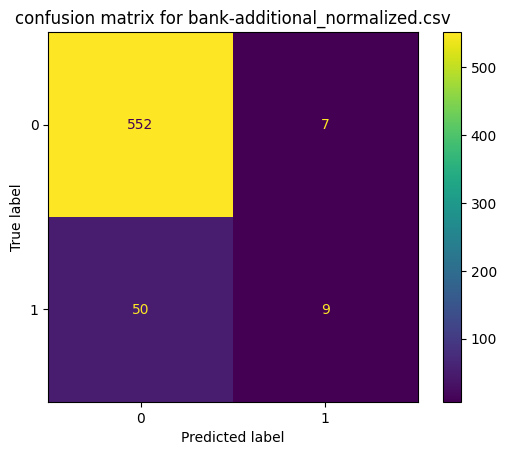
Comparison between predicted labels of test set obtained in **SVM\_ring\_separable.py** and true labels obtained in **Ring\_Datasets.ipynb**

****



**b) SVM\_ring\_merged.ipynb**





 **c) SVM\_bank-F.ipynb**

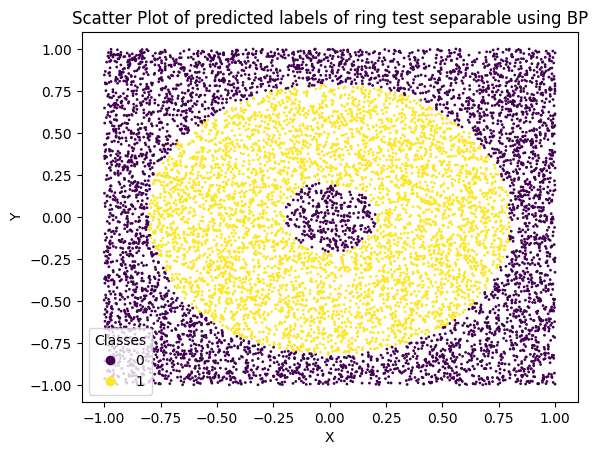
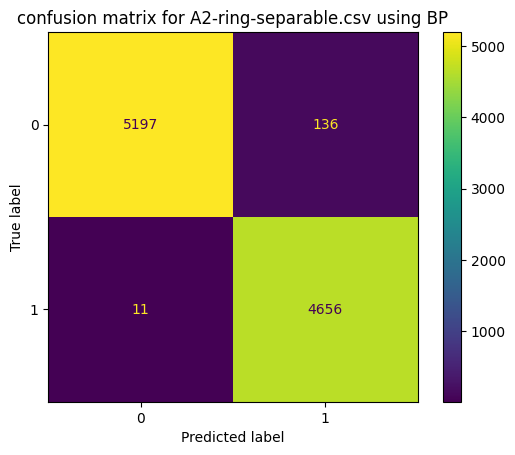
kernel\_type used is linear

accuracy obtained = 90.7767%

1. **BP classification model for Dataset in**
2. **BP\_ring\_separable.ipynb**

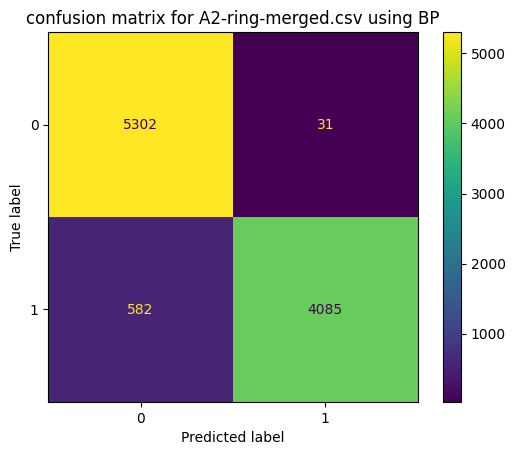
**TensorFlow** library is used to train the model using the training set of **A2-ring-separable.txt** with parameters :

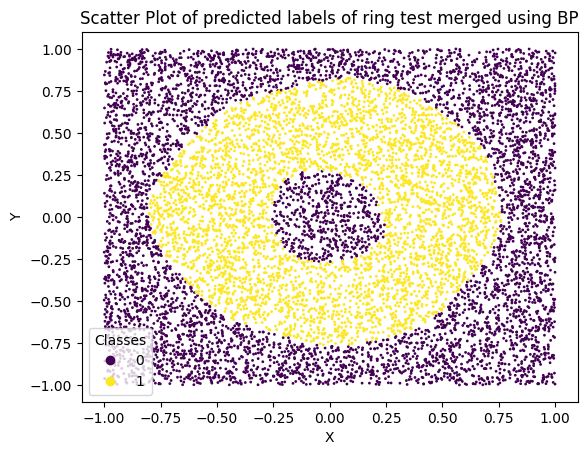
* + 1 input layer 2 \* 10000
  + 3 hidden layers with sizes 100, 50 and 25
  + 1 output layer with size 1
  + Learning rate 0.01
  + Momentum 0.9
  + Number of epochs 100



**b) BP\_ring\_merged.ipynb**

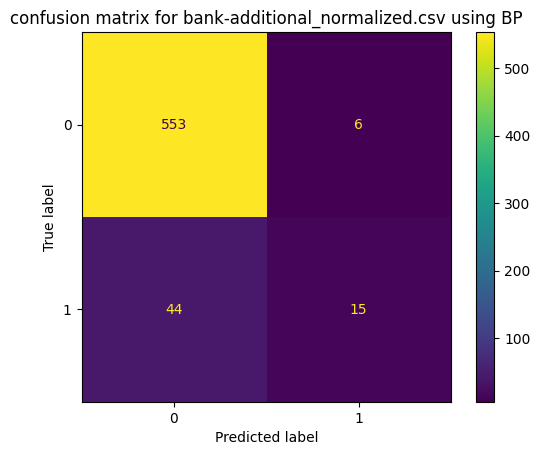
**TensorFlow** library is used to train the model using the training set of **A2-ring-merged.txt** with parameters :

* + 1 input layer 2 \* 10000
  + 3 hidden layers with sizes 100, 50 and 25
  + 1 output layer with size 1
  + Learning rate 0.01
  + Momentum 0.9
  + Number of epochs 100



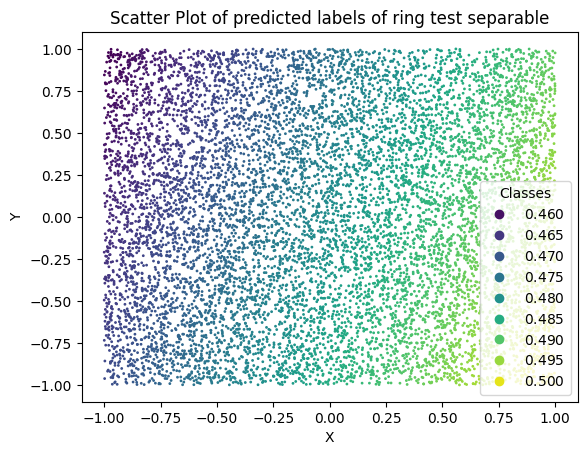
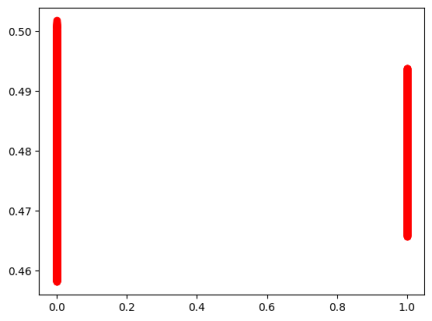
**c) BP\_bank.ipynb**

**TensorFlow** library is used to train the model using the training set of **A2-ring-merged.txt** with parameters :

* + 1 input layer 2 \* 10000
  + 3 hidden layers with sizes 100, 50 and 25
  + 1 output layer with size 1
  + Learning rate 0.01
  + Momentum 0.9
  + Number of epochs 100

1. **MLR classification model for Ring Dataset in**
2. **MLR\_ring\_separable.ipynb**

Results are incorrect since MLR cannot be used to predict nonlinear data which is expected to be the same for MLR\_ring\_separable.ipynb



1. **MLR\_Bank.ipynb**

