**MACHINE LEARNING FROM DATA**

**Report: Lab Session 6– Support Vector Machines**

Questions

**Q1**: Complete a table with the training and test errors for the linear and Gaussian SVMs. Which are the values of C in each case and the value of h for the Gaussian SVM?

**LINEAR**

**(C=0.1)**

train error: 0.071739

train confusion matrix:

[[2127 103]

[ 161 1289]]

test error: 0.061889

test confusion matrix:

[[536 22]

[ 35 328]]

**NON-LINEAR GAUSSIAN KERNEL**

**(C=0.1 | GAMMA=H=1)**

train error: 0.376359

train confusion matrix:

[[2230 0]

[1385 65]]

test error: 0.385451

test confusion matrix:

[[558 0]

[355 8]]

**Q2**: Plot the training and the validation scores, find the optimal value of C

A graph with a line and a line

Description automatically generated

Best Hyperparameters: {'clf\_\_C': 100}

**Q3**: For the **best classifier** found in the previous step, compute classification error on the test set, compare with the error obtained for the non-optimized linear classifier (Q1).

train error: 0.066033

train confusion matrix:

[[2133 97]

[ 146 1304]]

test error: 0.062975

test confusion matrix:

[[536 22]

[ 36 327]]

The error is almost identical for both train and test. The error in the best classifier is slightly higher when compared to the original one, this is caused by the different split of data used to train (fit) and test.

**Q4**: Plot the **training and validation scores** (two 3D plots). Find the optimal values of C and h.

Best Hyperparameters: {'clf\_\_C': 100, 'clf\_\_gamma': 0.001}  
A graph of a set of graphs

Description automatically generated with medium confidence

**Q5**: For the **best classifier** found in the previous step, compute classification error on the test set, compare with the error obtained for the non-optimized Gaussian classifier (Q1) and for the results of the optimized linear SVM (Q3).

A screenshot of a graph

Description automatically generated

TODO

**Q6**: Compute the confusion matrix for the test set. Compute the six metrics (*error*, *accuracy*, *precision*, *recall*, *specificity* and *f-score*).

A screenshot of a computer screen

Description automatically generated

train error: 0.066033

train confusion matrix:

[[2133 97]

[ 146 1304]]

test error: 0.062975

test confusion matrix:

[[536 22]

[ 36 327]]

Specificity = 1-recall

**Q7**: Explain why *precision*, *recall*, *specificity* and *f-score* are more appropriate than *accuracy* and *error* for evaluating the classifier performance.

Precision, recall, specificity, and F1-score provide more information about the classifier performance. This difference is especially in imbalanced datasets:

* Accuracy ignores the distribution of the classes, and it is not good if there is imbalance in the classes.
* Precision focuses on the correct positive predictions among all the positive prediction. This would mean that the false positives would be minimized.
* Recall focuses on how many actual positives are correctly identified in relation to all the real positive elements. This would increase the number of true positives and reduce the number of false negatives.
* Specificity focuses on the true negative element in relation to all the real negative elements, minimizing the appearance of false positives.
* F1-score focuses on both precision and recall, so we would get the benefits of both.