INGI2262 – Machine Learning

Assignment 3

Linear Discriminants and Support Vector Machines

1 Written assignment

Q 1 We consider a binary classification problem in $\mathbb{R}^2$. The training set is made of 5 examples described in the table below. Each training example is represented by one row in this table. The two first columns give the coordinates of each example while the last column is the class label. We are interested in a SVM classifier built from this training set. We consider in particular the maximal margin hyperplane solution in the original input space.

x_1	x_2	Class label
0	0	+
-2	0	+
2	2	-
4	2	-
0	4	-

Q 1.1 Represent **graphically**[1] the 5 training examples in $\mathbb{R}^2$ and compute the maximal margin hyperplane. You are **not** supposed to compute this decision boundary by solving the full optimization problem on paper. You are supposed to **reason geometrically** in order to determine the solution. Report the **equation** of the maximal margin hyperplane (this can also be computed easily from the geometry of the problem).

Q 1.2 For the classifier defined in question 1.1, what is the predicted class of the test example $x_1 = 2, x_2 = -1.25$? **Explain your reasoning.**

Q 1.3 Which are the support vectors for the model estimated in question 1.1? **Why?**

Q 1.4 Represent **graphically** a hyperplane that would perfectly separate the training data without being the maximal margin solution. Report the equation of this hyperplane.

Q 1.5 Define an additional training example such that the two classes are no longer separable by a hyperplane. What would be the solution returned by a hard margin SVM solver in such a case?

[1] You are supposed to include a 2-D representation of the input space in your report. The content of such representation matters more than its form. You are thus free to use a graphical software to produce such a representation but a scanned image of a clear drawing by hand could also do the job.

Q 2 We consider an input space made of real vectors in $\mathbb{R}^2$ (two-dimensional real vectors) and a polynomial kernel $k : \mathbb{R}^2 \times \mathbb{R}^2 \rightarrow \mathbb{R}$ defined as follows:

$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = (\langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle + 1)^3 \text{ with } \boldsymbol{x}_i, \boldsymbol{x}_j \in \mathbb{R}^2$$

Q 2.1 How many dimensions would have a feature space defined by this kernel? Define mathematically a projection ϕ that would map an input vector $\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ to this new feature space.

Q 2.2 Is it possible to have a set of training examples that would **not** be linearly separable in the input space but that could be linearly separable once projected to the new feature space defined in question 2.1? If so, report a small and illustrative training set in $\mathbb{R}^2$. If not so, **why?**

Q 2.3 What would be the feature space for the following kernel? Stress the similarities and differences from your answer to question 2.1

$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = (\langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle)^3 \text{ with } \boldsymbol{x}_i, \boldsymbol{x}_j \in \mathbb{R}^2$$

Q 3 We consider the following training set of points represented in a one dimensional space $\mathbb{R}$.

x	Class label
1	+
2	+
4	-
5	-
6	+

Q 3.1 What is the form of a linear discriminant lying is such one dimensional space $\mathbb{R}$? What could be the equation of such a linear model discriminating at best these training points? Are there any mistake when this model is used to classify the training examples? Why?

Q 3.2 Suppose one would consider the following kernel: $k(\boldsymbol{x}_i, \boldsymbol{x}_j) = (\langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle + 1)^2$ and one would look for a maximal margin hyperplane in the feature space induced by this kernel. With a proper choice of the regularization constant ($C = 100$), a solver for the dual problem on this training set returns the following α values, according to the order of the training points in the above table.

$$\alpha_1 = 0, \alpha_2 = 2.5, \alpha_3 = 0, \alpha_4 = 7.333, \alpha_5 = 4.833$$

Report the equation of the discriminant function $g(x)$ which is used by this model to classify any new point $x \in \mathbb{R}$.

Q 3.3 How is such discriminant function represented in the original input space $\mathbb{R}$? Are there any mistake when this model is used to classify the training examples? Why?

2 Mini-project

2.1 Task description

Speech recognition is the task of automatically transforming a spoken utterance captured by a microphone into a textual form representing what has been said. A simplified version of this task is the problem of isolated letter recognition, for which individual letters are uttered and must be recognized. You are in charge of implementing a software to perform this task.

The 617 input features are made of spectral coefficients (representing the energy of the speech in various frequency bands), contour features, sonorant, pre- or post-sonorant features. The LETTERS dataset is available as ***Letters.zip*** on Moodle. This archive contains 3 `.csv` files for being used with the R Software, ***LettersTrain.csv, LettersValid.csv*** and ***LettersTest.csv***, to be used as training, validation and test sets respectively.

Those ***.csv*** files should be read and loaded in R data frames. Each line is actually starting by a sample ID which should not be considered as an input feature but can be conveniently used as a row name once loaded into a R data frame. The subsequent columns correspond to the numerical input features. The last column, to be considered as a R factor, represents the class label.

2.2 Questions

Q 4 Estimate a Support Vector Machine on the ***training set*** to predict the class label. Use in particular a classifier learned with **`svm`** from the **`e1071`** package. Some pre-processing of the data could be considered (check the scaling of feature values, possibly discard some features with low variance). You have to decide whether and how to do it. Analyze the classification performance on the ***validation set*** as a function of the choice of the meta-parameters you estimate important. Consider at least the kernel choice, the kernel meta-parameter(s) if any, and the C regularization constant[2].
Report learning curves by measuring classification performance on the validation set with an increasing number of training examples. Choose an increasing proportion of

[2] This constant is referred as such in the lecture slides but may be denoted differently in a concrete package. Check the documentation to figure this out.

training examples uniformly at random from the global training set and report median performances on the validation set (boxplots may be useful to illustrate the variability across different selections of training examples).
Are the default meta-parameter definitions of the `svm` method appropriate for your experiments on this dataset? Use the validation set to select the best choices of meta-parameters. Can you conclude that the results obtained on the ***validation set***, after selecting the optimal meta-parameters, predict accurately ***test set*** classification results? Turn in one or several plot(s) to sustain your claims.

Q 5 Is the optimal performance of a SVM on the LETTERS dataset correlated with the minimal number of support vectors obtained when varying the meta-parameters? Estimate a SVM on the whole training set according to the best choices of meta-parameters (as determined previously on the validation set) and report performances on the whole test set. Choose a significantly different meta-parameters setting (based on the validation results computed in question 4), report the corresponding number of support vectors, training and test set classification results. Conclude about the observed relationships between those 3 quantities and report a plot to support your conclusion.

2.3 Extra credit (5 points)

Q 6 We consider now a binary classification problem from a simplified version of the LETTERS dataset restricted to 10 specific input features and only the examples either labeled **`A`** or **`E`**.
Such a data set is available from Moodle in the **`RestrictedLetters.RData`** file. You can load it in your R environment through the R command

```
> load("RestrictedLetters.RData")
```

and you should get 3 dataframes **`train, valid, test`**, respectively as training, validation and test sets.
Implement in R the ***perceptron with margin*** algorithm as described in the lecture slides. Run this algorithm on the training set till observing convergence or when a maximal number of iterations is reached. How is the margin chosen influencing the number of iterations performed before convergence? To draw sound conclusions, you should probably repeat your experiments for various initializations of the model. Choose an optimal margin value so that the classification accuracy on the validation set is optimized. Build a final model on the training set with the optimal meta-parameters (max. number of iterations, learning rate, margin, ...) and report its classification accuracy on the test set. How do your results compare to those obtained with a SVM and a linear kernel, respectively learned and tested on the same data as the perceptron with margin?

Submit a **PDF** report with answers to the questions of this assignment. Add any comment you consider relevant about the use of the R software for this task.