Examination

Linköping University, Department of Computer and Information Science, Statistics

Course code and name	TDDE01 Machine Learning
Date and time	2023-01-13, 14.00-19.00
Assisting teacher	Oleg Sysoev
Allowed aids	PDF of the course book + your help file (if submitted to LISAM in due time)
Grades:	
	5=18-20 points
	4=14-17 points
	3=10-13 points
	U=0-9 points

Provide a detailed report that includes plots, conclusions and interpretations. Give motivated answers to the questions. If an answer is not motivated, the points are reduced. Provide all necessary codes in the appendix.

Note: seed 12345 should be used in all codes that assumes randomness unless stated otherwise!

To start work in RStudio, type this in the Terminal application:

```
module add courses/TDDE01
rstudio
```

To submit your report:

1. Create one file (DOC, DOCX, ODT, PDF)
2. Use Exam Client to submit, and choose Assignment 1 in the drop box
3. Attach your report
4. Submit.
5. “Request Received” status implies that your report is successfully submitted.

Assignment 1 (10p)

File **tecator.csv** contains results of spectrographic analysis of various samples, including their characteristics such as Fat, Protein and Moisture content.

1. Split data into training and test sets (50/50) appropriately. Compute training and test MSE for K-nearest neighbor models having k=1,2,…,30 and Fat as target and all Channels as features. Plot dependences of the training and test errors as a function of k and interpret this plot in terms of bias-variance tradeoff. Report the training and test errors for the optimal model, and which k corresponds to the optimal model. **(4p)**
    a. **Note:** make sure to choose the right "kernel" parameter in kknn, that corresponds to the conventional K-nearest neighbor algorithm.
2. Perform PCA on all Channel columns of the original dataset (without scaling), obtain the PCA scores (coordinates of the data in the PC coordinate system), and then split the scores and Fat column into training and test data by using same observation indices as in step 1. Compute a K-nearest neighbor model using Fat as target and first 10 PC columns as features and optimal k from step 1 and report training and test MSEs. Compare them to the MSEs obtained for the optimal model in step 1 and comment why there such a difference, especially for K-nearest neighbor models? (**2p)**
3. Scale the training and test sets from step 1 appropriately. Consider a third-degree polynomial model where Protein is target and Fat is feature and implement the code that uses the Conjugate Gradient (CG) optimizer to find optimal parameters of this model, and plots dependence of training and test log(MSE) on the optimization iteration number. Is early stopping needed? Make a plot of (Fat, Protein) training data overlayed by the (Fat, Protein_predicted) of the model corresponding to the number of iterations equal to 20, and same plot for the number of iterations equal to 200, compare these plots and make necessary conclusions. **(4p)**
    a. **Note:** you are supposed to use function optim() to conduct the optimization.
    b. **Note2:** as a vector of initial parameter values of the optimizer, use zero vector.

Assignment 2 (10p)

Support Vector Machines - 5 Points

You are asked to use the function **ksvm** from the R package **kernlab** to learn a support vector machine (SVM) for classifying the **spam** dataset that is included with the package. Consider the radial basis function kernel (also known as Gaussian) with a width of 0.05. For the C parameter, consider values 0.5, 1 and 5. This implies that you have to consider three models.

(2 p) Perform model selection, i.e. select the most promising of the three models (use any method of your choice).

(1 p) Estimate the generalization error of the SVM selected above (use any method of your choice).

(1 p) Produce the SVM that will be returned to the user, i.e. show the code.

(1 p) What is the purpose of the parameter C ?

Kernel Methods - 5 Points

One of the labs in the course consisted in implementing a kernel method to predict the hourly temperatures for a date and place in Sweden. Modify your lab solution to classify the **spam** dataset used in the previous exercise. Use the Gaussian kernel and show results for different kernel widths, e.g. use 2/3 of the data for learning and 1/3 for testing. Use only the first 48 attributes in the dataset, in addition to the last one which is the class label. Assume that the class label is a continuous random variable (so that you actually solve a regression problem, like you did in the lab).