Examination

Linköping University, Department of Computer and Information Science, Statistics

Course code and name	TDDE01 Machine Learning
Date and time	2025-03-20, 14.00-19.00
Assisting teacher	Oleg Sysoev
Allowed aids	Contents of “help_materials” folder + 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 (allowed formats: 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 **wdbc.csv** contains measurements of tumor images as well as type of the tumor given by the Diagnosis variable.

1. Scale the data and apply cross-validation compute LASSO regression models which predict Area from the other numerical variables (i.e exclude Diagnosis). Provide a plot showing dependence of the cross-validation error and its uncertainty on the value of the penalty parameter. How many features are selected by the optimal model? Report an equation that shows how the target variable can be predicted from the features when $\log(\lambda) = -2$. Is the model for $\log(\lambda) = -2$ statistically significantly different from the optimal model? **(3p)**
2. Divide the original (unscaled) data into training and validation (50/50) and compute a logistic regression model in which Diagnosis is predicted from the remaining variables. Compute the training and test misclassification errors and comment if the model seems to be overfitted. Assuming "M" to be the positive class, compute precision values for the test data with the following loss matrices and comment why precision changed in that direction with L_2 **(3p)**:

$$L_1 = \begin{matrix} \\ True\ B \\ True\ M \end{matrix} \begin{matrix} \begin{matrix} Predict\ B & Predict\ M \end{matrix} \\ \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \end{matrix}$$

$$L_2 = \begin{matrix} \\ True\ B \\ True\ M \end{matrix} \begin{matrix} \begin{matrix} Predict\ B & Predict\ M \end{matrix} \\ \begin{pmatrix} 0 & 20 \\ 1 & 0 \end{pmatrix} \end{matrix}$$

3. Assume now the following classification model: $\hat{y}(x) = sign\big(f(x)\big)$ where $f(x) = w^T x$, where $\hat{y}$ is the predicted diagnosis ("-1" corresponds to "B" and "1" corresponds to "M"), x are all other variables in the data and w is the set of parameters. Implement a cost function in R depending on argument w that uses this model, training data from step 2 and the Hinge loss function. Optimize the cost function from the starting point (0,…, 0) by the BFGS method and estimate training and test misclassification errors for the optimal model. Report the estimated predictive equation. Compare the quality of this model with the quality of the logistic model from step 2. (**4p**)

Assignment 2 (10p)

KERNEL MODELS – 8 Points

In the course, you have learned about kernel models for classification and regression. Kernel models can also be used for density estimation, i.e., to model a probability distribution or density function p(x_*). In particular,

$$p(x_*) = \frac{1}{n}\sum_{i=1}^{n} k\left(\frac{x_* - x_i}{h}\right)$$

where the kernel function k() must integrate to 1. To ensure this, you will hereinafter consider k() to be the density function of a Gaussian distribution with mean equal to 0 and standard deviation equal to 1. You can get it by using the command `dnorm` in R.

The kernel model presented above can be used to estimate the class conditional density functions p(x_*|class=1) and p(x_*|class=2). These can in their turn be used to produce posterior class probabilities p(class|x_*) via Bayes theorem. Specifically,

p(class=1|x_*) = p(x_*|class=1) p(class=1) / [p(x_*|class=1) p(class=1) + p(x_*|class=2) p(class=2)]

You are asked to **come up with a dataset of your own** to illustrate the effect of h in the generalization error. Specifically, you should show that overfitting occurs for certain value of h, and underfitting occurs for some other value of h.

NEURAL NETWORKS - 2 Points

Run the code below to train a neural network for substracting two numbers from the interval [-1,1]. Look at the plot of the learned neural network and explain why the weights learned make sense.

```
library(neuralnet)
set.seed(1234567890)

x1 <- runif(1000, -1, 1)
x2 <- runif(1000, -1, 1)
tr <- data.frame(x1,x2, y=x1 - x2)

winit <- runif(9, -1, 1)
nn<-neuralnet(formula = y ~ x1 + x2, data = tr, hidden = c(1), act.fct = "tanh")
plot(nn)
```