Examination

Linköping University, Department of Computer and Information Science, Statistics

Course code and name	TDDE01 Machine Learning
Date and time	2020-01-16, 14.00-19.00
Assisting teacher	Oleg Sysoev
Allowed aids	"Pattern recognition and Machine Learning" by Bishop and "The Elements of Statistical learning" by Hastie
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!

Assignment 1 (7p)

The data file **glass.csv** contains information about the chemical components of two different glass types. The Type of glass is represented by variable Class (0/1), data also contains identification number ID. Import data to R and divide it into training, validation and test sets (40/30/30) by using this kind of code:

```
n=dim(data)[1]
set.seed(12345)
id=sample(1:n, floor(n*percent1))
train=data[id,]
id1=setdiff(1:n, id)
set.seed(12345)
id2=sample(id1, floor(n*percent2))
valid=data[id2,]
id3=setdiff(id1,id2)
test=data[id3,]
```

1. Use combined training and validation data to train a logistic regression model in which all chemical components are features and Class variable is the target. Report confusion matrix for the test data and comment on the quality of prediction. Report the fitted probabilistic model and the equation of the decision boundary. **(3p)**
2. Use training and validation data in order to train a decision tree with the same features and target as in step 1 and choose the optimal number of leaves in the decision tree. Provide a plot showing the training and validation error as the function of the number of leaves and report the confusion matrix for the test data. Explain how the optimal number of leaves should be selected according to this plot. Finally, combine predictions of logistic regression and decision tree together and compute confusion matrix for the test data by using the combined classifier. Compare the quality of prediction of the combined classifier with prediction quality of individual classifiers (Logistic, Tree) and make conclusions. **(4p)**

Assignment 2 (3p)

The subjects, students in grades 4-6 in selected schools in Michigan, were asked the following question: What would you most like to do at school?

A. Make good grades
B. Be good at sports.
C. Be popular.
Demographic information was also collected for each student. The collected information is available in **popularkids.csv**

- By using a Naïve Bayes model, find out what is predicted to be most important for a boy in grade 6 from Elm school: to make good grades, to be good in sports or to be popular? Do not use existing R packages for Naïve Bayes here: compute all probabilities needed by writing your own code.

Assignment 3 (10p)

SUPPORT VECTOR MACHINES - 4 POINTS

You are asked to use the function `ksvm` from the R package `kernlab` to estimate the generalization error of a support vector machine (SVM) for classification of the `spam` dataset, which is included with the package. Use the radial basis function kernel (also known as Gaussian kernel) with a width of 0.05. The C parameter can take value 0.1 or 1. Since the value of the C parameter is not fixed, you need to use nested cross-validation to solve this task.

NEURAL NETWORKS - 3 POINTS

In the lab on neural networks (NNs), you learned a NN to mimic the sine function. However, you did not estimate the generalization error of the learned NN. You are now asked to do it. Feel free to choose how to do it but note that you already used

all the data provided in the lab for learning the NN.

NEURAL NETWORKS - 3 POINTS

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

- Note that the activation function is `tanh`.
- The two intercepts are so small that you can disregard them.
- Note that the weights in the first layer are the inverse of the weight in the second layer, i.e. $0.13 \approx 1/7.75$.

```
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)
```