Examination

Linköping University, Department of Computer and Information Science, Statistics

Course code and name	732A95 Introduction to Machine Learning
Date and time	2018-01-11, 08.00-13.00
Assisting teacher	Oleg Sysoev
Allowed aids	"Pattern recognition and Machine Learning" by Bishop and "The Elements of Statistical learning" by Hastie
Grades:	A=19-20 points B=16-18 points C=11-15 points D=9-10 points E=7-8 points F=0-6 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. Use seed 12345 when randomness is present unless specified otherwise.

Assignment 1 (10p)

The data file **video.csv** contains characteristics of a sample of Youtube videos. Import data to R and divide it randomly (50/50) into training and test sets.

1. Perform principal component analysis using the numeric variables in the training data except of "utime" variable. Do this analysis with and without scaling of the features. How many components are necessary to explain more than 95% variation of the data in both cases? Explain why so few components are needed when scaling is not done. **(2p)**
2. Use only first 100 rows from the entire data and all numeric variables and create interaction variables up to the third order with the following code:

   ```
   data=t(apply(as.matrix(your_data_with_100rows_and_numeric_columns), 1,
   combn, 3, prod))
   ```

Use the obtained matrix to perform a Nearest Shrunken Centroid (NSC) analysis in which "codec" is target and all other interaction variables are features. Obtain the cross-validation plot and interpret it in terms of bias-variance tradeoff. How does model complexity change when the threshold Δ increases? **(2p)**

3. Use the cross-validated NSC model to extract the threshold giving the optimal value of the log-likelihood (log-likelihood values are also available in the cross-validated model). Why is the multinomial log-likelihood applied for this model? **(1p)**
4. Use original data to create variable "class" that shows "mpeg" if variable "codec" is equal to "mpeg4", and "other" for all other values of "codec". Create a plot of "duration" versus "frames" where cases are colored by "class". Do you think that the classes are easily separable by a linear decision boundary? **(1p)**
5. Fit a Linear Discriminant Analysis model with "class" as target and "frames" and "duration" as features to the entire dataset (scale features first). Produce the plot showing the classified data and report the training error. Explain why LDA was unable to achieve perfect (or nearly perfect) classification in this case. **(2p)**
6. Fit a decision tree model with "class" as target and "frames" and "duration" as features to the entire dataset, choose an appropriate tree size by cross-validation. Report the training error. How many leaves are there in the final tree? Explain why such a complicated tree is needed to describe such a simple decision boundary. **(2p)**

Assignment 2 (10p)

ONLINE LEARNING

(3p) Implement the budget online support vector machine (SVM). Check the course slides for the pseudo-code. Feel free to use the template below. Note that you are not using all the attributes and points in the file. Run your code on the spambase.csv file for the (M, β) values (500,0) and (500,-0.05). Plot the error rate as a function of the number of training point.

(2p) Analyze the results obtained. In particular,

- explain why (500,0) gives better results than (500,-0.05), and
- explain why the setting (50,0) is the slowest (you do not need to run your code until completion for this setting).

```
set.seed(1234567890)
spam <- read.csv2("spambase.csv")
ind <- sample(1:nrow(spam))
spam <- spam[ind,c(1:48,58)]
h <- 1
beta <- # Your value here
M <- # Your value here
N <- 500 # number of training points
gaussian_k <- function(x, h) { # Gaussian kernel
# Your code here
}
SVM <- function(sv,i) { # SVM on point i with support vectors sv
```

```
# Your code here
# Note that the labels in spambase.csv are 0/1 and SVMs need -1/+1
# Then, use 2*label-1 to convert from 0/1 to -1/+1
# Do not include the labels when computing the Euclidean distance between
# the point i and each of the support vectors. This is the distance to use
# in the kernel function. You can use dist() to compute the Euclidean
distance
}
errors <- 1
errorrate <- vector(length = N)
errorrate[1] <- 1
sv <- c(1)
for(i in 2:N) {
# Your code here

}

plot(errorrate[seq(from=1, to=N, by=10)], ylim=c(0.2,0.4), type="o")
length(sv)

errorrate[N]
```

Neural Networks

(3p) Train a neural network (NN) to learn the trigonometric sine function. To do so, sample 50 points uniformly at random in the interval [0, 10]. Apply the sine function to each point. The resulting pairs are the data available to you. Use 25 of the 50 points for training and the rest for validation. The validation set is used for early stop of the gradient descent. Consider threshold values $i/1000$ with $i = 1, \ldots, 10$. Initialize the weights of the neural network to random values in the interval $[-1, 1]$. Consider two NN architectures: A single hidden layer of 10 units, and two hidden layers with 3 units each. Choose the most appropriate NN architecture and threshold value. Motivate your choice. Feel free to reuse the code of the corresponding lab.

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

(1p) In the light of the results above, would you say that the more layers the better ? Motivate your answer.