Control Questions

1 Introduction to Machine Learning

Question 1.1
Explain the terms *Machine Learning* and *Artificial Intelligence* in your own words.

Question 1.2
Explain the differences between *regression* and *classification*.

Question 1.3
Explain the difference between *offline* and *online* learning.

Question 1.4
Explain the difference between *unsupervised*, *supervised* and *semi-supervised* learning.

Question 1.5
Give important examples of *semi-supervised* learning methods.

Question 1.6
Explain the difference between *Transfer Learning* and *Co-Training*?

Question 1.7
Explain the difference between *Active Learning* and *Self-Learning*.

Question 1.8
Explain the main idea of *Multiple Instance Learning*? When is this approach useful in practice?

2 Unsupervised Learning

Question 2.1
Explain the main idea if k-NN classification.

Question 2.2
What is a Voronoi Diagram and how does it relate to nearest neighbor classification?

Question 2.3
Explain the terms *clustering* and *cluster analysis* in your own words.

Question 2.4

Explain the similarities and differences between the following lines of code:

```
KM = kmeans(train, k)
kmc=KM$centers[1,]
```

and

```
PAM = pam(train,k)
kmc=PAM$medoids[1,]
```

Question 2.5

Describe the term *exemplar-based clustering* in your own words. What are the benefits of such methods compared to other clustering approaches?

Question 2.6

What is Affinity Propagation and how does it differ from other clustering methods?

Question 2.7

Describe the basic idea of *graph-based clustering* approaches in your own words.

Question 2.8

Describe *hierarchical clustering* and its relation to *minimal spanning tree* in your own words?

Question 2.9

Which clustering method would you use in practice? Why?

Question 2.10

Assuming that `P` was computed via

```
P = prcomp(data)
```

Explain the difference or similarities of the following lines of code:

```
p_data = predict(P, data)
p_data2 = p_data[,1:2]

P$rotation = P$rotation[,1:2]
p_data2= predict(P, data)
```

Question 2.11

Dimension reduction methods and *feature selection* both allow a reduction of the amount of data. However, feature selection provides a specific advantage. Which one?

Question 2.12

Name three possible approaches for feature selection.

Question 2.13
Explain the following lines of code:

```
weight=random.forest.importance(Species∼., iris, 2)
selfeat = cutoff.k(weight, 2)
```

3 Supervised Learning

Question 3.1
Explain the difference between *generative* and *discriminative* classification.

Question 3.2
Describe the basic idea of *linear classification*?

Question 3.3
What is the *Fisher criterion* and how is it related to LDA?

Question 3.4
How are the optimal solution of *LDA* and the eigenvalue problem related?

Question 3.5
Describe the *Small Sample Size Problem* and possible solutions to resolve it.

Question 3.6
Which theoretical problems arise from the standard solution of LDA and how can these be resolved.

Question 3.7
Explain the similarities and differences between *LDA* and *QDA*.

Question 3.8
Explain the basic idea of *Canonical Correlation Analysis*.

Question 3.9
Explain the basic principle of Mahalanobis metric learning.

Question 3.10
Explain the similarities and differences between the following lines of code:

```
L = lda(train, labels)
pl = predict(L, test)
```

```
S = svm(train, labels)
ps = predict(S, test, type="vector")
```

Question 3.11

Support Vector Machines build on two main principles. Which ones?

Question 3.12

Support vector machines are designed for linear problems. How can nonetheless also non-linear problems be solved?

Question 3.13

What are *Support Vectors*? Why are they theoretically/practically important?

Question 3.14

Describe the relation between the pseudo-inverse and the solution of a linear two-class problem?

Question 3.15

Explain the following problem in your own words:

$$\min \tfrac{1}{2}\|w\|^2$$
$$\text{s.t.} \quad y_i(\langle w, x_i \rangle + b) - 1 \geq 0$$

Question 3.16

What is meant by *Support Vector Expansion*? How is this concept related to the dual solution of the SVM?

Question 3.17

Explain the difference between *Hard Margin SVM* and *Soft Margin SVM*. Which variant would you use in practice?

Question 3.18

What is meant by the so-called *kernel trick*?

Question 3.19

Support vector machines are defined for two-class problems. How can also multi-class problems handled in practice?

Question 3.20

Explain the concept of a *decision tree* in your own words?

Question 3.21

Explain the meaning/function of nodes in a *decision tree*.

Question 3.22
Which strategies exist for training a decision tree?

Question 3.23
Explain the terms *entropy*, *information gain* and *Gini coefficient* in the context of decision trees.

Question 3.24
Name three different variants of decision trees and discuss their differences.

Question 3.25
Explain the basic idea of CARTs.

Question 3.26
Describe the main idea of a *Binary Decision Tree* in your own words?

Question 3.27
Explain the two basic ideas on which *Random Forests* build on.

Question 3.28
Explain the relationship between *Random Forests* and ensemble methods?

Question 3.29
Name three possible applications for random forests?

Question 3.30
What are the advantages of *Random Forests* compared to other classification methods?

Question 3.31
The following lines of code calculate and evaluate a *Random Forest* using default parameters. Explain the parameters and the influence on the classification results.

```
R = randomForest(train, label, ntree=10, mtry=100, maxnodes=1000)
P = predict(R, test)
```

4 Neural Networks

Question 4.1
Explain the key statement of the *Universal Approximation Theorem*.

Question 4.2

Why are GPUs, that originally have been developed to process and display graphics, are well suited for neural network calculations?

Question 4.3

Explain the basic concept of a neuron/perceptron and its relation to a biological neuron.

Question 4.4

How is the Nobel-Prize-winning work by Hubel and Wiesel related to neural networks?

Question 4.5

Name the five most important publications neural networks from your personal points of view and explain your decision.

Question 4.6

What is an *Activation Function* in the context of a neural network? Name at least three different examples?

Question 4.7

Explain the basic concept of neural networks from a graph-theoretical point of view: *multi-layer network*.

Question 4.8

What is a *hidden layer* in the context of neural networks?

Question 4.9

How can a neural network be trained? What is the most successful approach?

Question 4.10

Explain the basic idea of the backpropagation algorithm.

Question 4.11

In the context of training neural networks, what is meant by a *loss function*?

Question 4.12

Explain the basic idea of the *softmax classifier*.

Question 4.13

Why is regularization important for optimization problems? Name three examples in the context of neural networks.

Question 4.14

Explain the essential difference between *gradient decent* and *stochastic gradient descent*.

Question 4.15

What is meant by *batch normalization*? Why is this advantageous in practice?

Question 4.16

Explain the basic idea of the *Momentum* method.

Question 4.17

Explain the basic idea of *Dropout*.

Question 4.18

Explain the meaning of the parameters in the following call to the training function of the *deepnet* package:

```
NN = nn.train(data, label,
    hidden=c(15.5),
    activationfun="sigm",
    learning rate = 0.1,
    momentum=0.5,
    output="softmax",
    batchsize=32,
    numberepochs=50,
    hidden_dropout=0.2)
```

Question 4.19

A CNN consists of several different layers. Which one is defined in the following lines of code (*Keras*). What is the meaning of the listed parameters?

```
layer_conv_2d(filter=16,
        kernel_size=c(5,5),
        padding="same",
        input_shape=c(64, 64, 3)) %>%

layer_activation("relu") %>%
```

Question 4.20

A CNN consists of several different layers. Which guy will go with defined in the following *Keras* lines of code. What meaning do they have listed parameters?

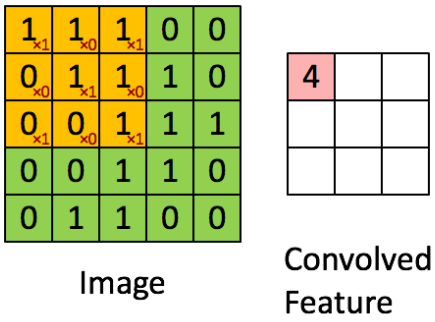
```
layer_flatten() %>%
layer_dense(num_of_units) %>%
layer_activation("relu") %>%
layer_dropout(0.5) %>%
```

Question 4.21

What is a (Deep) Convolutional Neural Network?

Question 4.22

Given a binary image (green square) and a 3×3 convolution filter (orange square):



Calculate the result of the filter $\begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ on the area highlighted in orange!

Question 4.23

Given an image of size 9×9 and a filter of size 3×3. Calculate the size of the filter output for a stride size of 2.

Question 4.24

Given a matrix of size 4×4:

$$\begin{bmatrix} 2 & 5 & 3 & 2 \\ 9 & 4 & 4 & 3 \\ 9 & 2 & 2 & 3 \\ 1 & 4 & 1 & 2 \end{bmatrix}$$

Calculate for each of these a 2×2 *max pooling* or *average pooling*.

Question 4.25

AlexNet was not the first CNN architecture to be used in the ImageNet challenge. Why was this one so successful?

Question 4.26

Explain the meaning of *Transfer Learning* in the context of Deep Learning. When can this technique not be used?

Question 4.27

What is *Data Augmentation* and why is it important in the context of Deep Learning?