Exam in Statistical Machine Learning
Statistisk Maskininlärning (1RT700)

Date and time: August 22, 2020, 09.00–14.00

Responsible teacher: Johan Wågberg

Number of problems: 5

Aiding material: Calculator, mathematical handbooks, the course book.

Preliminary grades:	grade 3	23 points
	grade 4	33 points
	grade 5	43 points

Some general instructions and information:

- Your solutions can be given in Swedish or in English.
- Write only on one side of the paper.
- Write your exam code and page number on all pages.
- Do not use a red pen.
- Use separate sheets of paper for the different problems
  (i.e. the numbered problems, 1–5).
- For subproblems (a), (b), (c), . . . , it is usually possible to answer later subproblems independently of the earlier subproblems (for example, you can answer (b) without answering (a)).

With the exception of Problem 1, ***all your answers must be clearly motivated!***
A correct answer without a proper motivation will score zero points!

Good luck!

Some relevant formulas

Pages 1–3 contain some expressions that may or may not be useful for solving the exam problems. *This is not a complete list of formulas used in the course*, but some of the problems may require knowledge about certain expressions not listed here. Furthermore, the formulas listed below *are not self-explanatory*, meaning that you need to be familiar with the expressions to be able to interpret them. They are possibly a support for solving the problems, but *not* a comprehensive summary of the course.

The Gaussian distribution: The probability density function of the p-dimensional Gaussian distribution with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$ is

$$\mathcal{N}(\mathbf{x} \,|\, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{p/2}\sqrt{\det \boldsymbol{\Sigma}}} \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^{\mathsf{T}}\boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})\right), \qquad \mathbf{x} \in \mathbb{R}^p.$$

Sum of identically distributed variables: For identically distributed random variables $\{z_i\}_{i=1}^n$ with mean μ, variance σ^2 and average correlation between distinct variables ρ, it holds that $\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^n z_i\right] = \mu$ and Var $\left(\frac{1}{n}\sum_{i=1}^n z_i\right) = \frac{1-\rho}{n}\sigma^2 + \rho\sigma^2$.

Linear regression and regularization:

- The least-squares estimate of θ in the linear regression model

$$y = \theta_0 + \sum_{j=1}^{p} \theta_j x_j + \epsilon$$

  is given by the solution $\widehat{\theta}_{\text{LS}}$ to the normal equations $\mathbf{X}^{\mathsf{T}}\mathbf{X}\widehat{\theta}_{\text{LS}} = \mathbf{X}^{\mathsf{T}}\mathbf{y}$, where

$$\mathbf{X} = \begin{bmatrix} 1 & -\mathbf{x}_1^{\mathsf{T}}- \\ 1 & -\mathbf{x}_2^{\mathsf{T}}- \\ \vdots & \vdots \\ 1 & -\mathbf{x}_n^{\mathsf{T}}- \end{bmatrix} \text{ and } \mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \text{ from the training data } \mathcal{T} = \{\mathbf{x}_i, y_i\}_{i=1}^n$$

- Ridge regression uses the regularization term $\lambda\|\theta\|_2^2 = \lambda\sum_{j=0}^p \theta_j^2$.
  The ridge regression estimate is $\widehat{\theta}_{\text{RR}} = (\mathbf{X}^{\mathsf{T}}\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^{\mathsf{T}}\mathbf{y}$.

- LASSO uses the regularization term $\lambda\|\boldsymbol{\theta}\|_1 = \lambda\sum_{j=0}^p |\theta_j|$.

Maximum likelihood: The maximum likelihood estimate is given by

$$\widehat{\boldsymbol{\theta}}_{\text{ML}} = \arg\max_{\boldsymbol{\theta}} \ln \ell(\boldsymbol{\theta})$$

where $\ln \ell(\boldsymbol{\theta}) = \sum_{i=1}^{n} \ln p(y_i \,|\, \mathbf{x}_i; \boldsymbol{\theta})$ is the log-likelihood function (the last equality holds when the n training data points are modeled to be independent).

Logistic regression: The logistic regression combines linear regression with the logistic function to model the class probability

$$p(y = 1 \,|\, \mathbf{x}) = \frac{e^{\boldsymbol{\theta}^\mathsf{T}\mathbf{x}}}{1 + e^{\boldsymbol{\theta}^\mathsf{T}\mathbf{x}}}.$$

For multi-class logistic regression we use the *softmax* function and model

$$p(y = m \,|\, \mathbf{x}_i) = \frac{e^{\boldsymbol{\theta}_m^\mathsf{T}\mathbf{x}_i}}{\sum_{j=1}^{M} e^{\boldsymbol{\theta}_j^\mathsf{T}\mathbf{x}_i}}.$$

Discriminant Analysis: The linear discriminant analysis (LDA) classifier models $p(y \,|\, \mathbf{x})$ using Bayes' theorem and the following assumptions

$$p(y = m \,|\, \mathbf{x}) = \frac{p(\mathbf{x} \,|\, m)p(y = m)}{\sum_{j=1}^{M} p(\mathbf{x} \,|\, j)p(y = j)} \approx \frac{\mathcal{N}\left(\mathbf{x} \,|\, \widehat{\boldsymbol{\mu}}_m, \widehat{\boldsymbol{\Sigma}}\right)\widehat{\pi}_m}{\sum_{j=1}^{M} \mathcal{N}\left(\mathbf{x} \,|\, \widehat{\boldsymbol{\mu}}_j, \widehat{\boldsymbol{\Sigma}}\right)\widehat{\pi}_j},$$

where

$$\begin{aligned}
\widehat{\pi}_m &= n_m/n \text{ for } m = 1, \dots, M \\
\widehat{\boldsymbol{\mu}}_m &= \frac{1}{n_m} \sum_{i:y_i=m} \mathbf{x}_i \text{ for } m = 1, \dots, M \\
\widehat{\boldsymbol{\Sigma}} &= \frac{1}{n - M} \sum_{m=1}^{M} \sum_{i:y_i=m} (\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_m)(\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_m)^\mathsf{T}.
\end{aligned}$$

For quadratic discriminant analysis (QDA), the model is

$$p(y = m \,|\, \mathbf{x}) \approx \frac{\mathcal{N}\left(\mathbf{x} \,|\, \widehat{\boldsymbol{\mu}}_m, \widehat{\boldsymbol{\Sigma}}_m\right)\widehat{\pi}_m}{\sum_{j=1}^{M} \mathcal{N}\left(\mathbf{x} \,|\, \widehat{\boldsymbol{\mu}}_j, \widehat{\boldsymbol{\Sigma}}_j\right)\widehat{\pi}_j},$$

where $\widehat{\boldsymbol{\mu}}_m$ and $\widehat{\pi}_m$ are as for LDA, and

$$\widehat{\boldsymbol{\Sigma}}_m = \frac{1}{n - 1} \sum_{i:y_i=m} (\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_m)(\mathbf{x}_i - \widehat{\boldsymbol{\mu}}_m)^\mathsf{T}.$$

Classification trees: The cost function for tree splitting is $\sum_{\ell=1}^{|T|} n_\ell Q_\ell$ where T is the tree, $|T|$ the number of terminal nodes, n_ℓ the number of training data points falling in node ℓ, and Q_ℓ the impurity of node ℓ. Three common impurity measures for splitting classification trees are:

$$\begin{aligned}
\text{Misclassification error:} \qquad & Q_\ell = 1 - \max_m \widehat{\pi}_{\ell m} \\
\text{Gini index:} \qquad & Q_\ell = \sum_{m=1}^{M} \widehat{\pi}_{\ell m}(1 - \widehat{\pi}_{\ell m}) \\
\text{Entropy/deviance:} \qquad & Q_\ell = -\sum_{m=1}^{M} \widehat{\pi}_{\ell m} \log \widehat{\pi}_{\ell m}
\end{aligned}$$

where $\widehat{\pi}_{\ell m} = \frac{1}{n_\ell} \sum_{i:\, \mathbf{x}_i \in R_\ell} \mathbb{I}(y_i = m)$

Loss functions for classification: For a binary classifier expressed as $\widehat{y}(\mathbf{x}) = \operatorname{sign}\{C(\mathbf{x})\}$, for some real-valued function $C(\mathbf{x})$, the margin is defined as $y \cdot C(\mathbf{x})$ (note the convention $y \in \{-1, 1\}$ here). A few common loss functions expressed in terms of the margin, $L(y, C(\mathbf{x}))$ are,

$$\begin{aligned}
\text{Exponential loss:} \qquad & L(y, c) = \exp(-yc). \\
\text{Hinge loss:} \qquad & L(y, c) = \begin{cases} 1 - yc & \text{for } yc < 1, \\ 0 & \text{otherwise.} \end{cases} \\
\text{Binomial deviance:} \qquad & L(y, c) = \log(1 + \exp(-yc)). \\
\text{Huber-like loss:} \qquad & L(y, c) = \begin{cases} -yc & \text{for } yc < -1, \\ \frac{1}{4}(1 - yc)^2 & \text{for } -1 \leq yc \leq 0, \\ 0 & \text{otherwise.} \end{cases} \\
\text{Misclassification loss:} \qquad & L(y, c) = \begin{cases} 1 & \text{for } yc < 0, \\ 0 & \text{otherwise.} \end{cases}
\end{aligned}$$

1. This problem is composed of 10 true-or-false statements. You only have to classify these as either `true` or `false`. For this problem *(only!)* no motivation is required. Each correct answer scores 1 point, each incorrect answer scores -1 point and each missing answer scores 0 points. The total score will never be less than 0.

   i. A classification tree with a single binary split is a linear classifier.

   ii. Regularization allows us to restrict the flexibility of a model.

   iii. LDA is a special case of QDA.

   iv. The absolute error loss function is more robust to outliers than the squared error loss function.

   v. Random forest is an extension of Adaboost.

   vi. If gradient decent converges, the solution is guaranteed to be a local minimum.

   vii. Boosting primarily increases performance by reducing the bias of the base model.

   viii. The optimal weights in deep learning always have a closed form solution, but it is to expensive to compute when the number of data points is large.

   ix. Using bootstrap aggregating, we never have to worry about how much data we have collected. We can always sample more datasets to improve our models.

   x. A linear classifier has a linear decision boundary.

(10p)

2. Consider the dataset in Table 1. Using this data, we will study three classifiers.

x_1	x_2	y
-6.0	-9.0	$\times$
-1.0	-7.0	$\times$
1.0	1.0	$\times$
-3.0	7.0	$\circ$
7.0	7.0	$\circ$
2.0	1.0	$\circ$

Table 1: Data for problem 2, where $\mathbf{x} = \begin{bmatrix} x_1 & x_2 \end{bmatrix}^\mathsf{T}$ is the input and y the class label.

Note: *Problems (b), (c) and (d) can be solved independently.*

(a) Illustrate the dataset in a graph with x_1 and x_2 on the two axes.

(1p)

(b) Learning a logistic regression classifier using the input $\widetilde{\mathbf{x}} = \begin{bmatrix} 1 & x_1 & x_2 \end{bmatrix}^\mathsf{T}$ gives the parameter vector[1] $\boldsymbol{\theta} = \begin{bmatrix} 50 & -20 & -20 \end{bmatrix}^\mathsf{T}$. Compute the decision boundary and draw it in the same figure as (a). State also the misclassification error on training data.

(2p)

(c) Compute the parameters of a LDA (linear discriminative analysis) classifier and draw the decision boundary in the same figure as (a). What is the achieved misclassification error on training data?

Hint: *The decision boundary is orthogonal to the vector* $\widehat{\boldsymbol{\Sigma}}^{-1}\left(\widehat{\boldsymbol{\mu}}_\times - \widehat{\boldsymbol{\mu}}_\circ\right)$ *and passes through the point* $\frac{1}{2}\left(\widehat{\boldsymbol{\mu}}_\times + \widehat{\boldsymbol{\mu}}_\circ\right)$.

(5p)

(d) Learn a decision tree using a single binary split in the x_1 coordinate that minimizes the misclassification error on training data[2]. Draw the corresponding decision boundary in the same figure as (a). What misclassification error is achieved?

(2p)

[1] Not really, you get $\boldsymbol{\theta} = \begin{bmatrix} 48.55544197 & -20.89866765 & -18.14626533 \end{bmatrix}^\mathsf{T}$ but the given $\boldsymbol{\theta}$ gives the same result qualitatively.

[2] You should only consider x_1 when doing the split and *not* x_2.

3. (a) Using a dataset $\mathcal{D} = \{\mathbf{x}_i, y_i\}_{i=1}^n$, the parameters of a linear model $\widehat{y} = \boldsymbol{\theta}^\mathsf{T}\mathbf{x}$, where $\boldsymbol{\theta} = \begin{bmatrix}\theta_1 & \theta_2 & \theta_3\end{bmatrix}^\mathsf{T}$, have been learned by minimizing three different cost functions

$$\begin{aligned}
\text{(i):}\quad J(\boldsymbol{\theta}) &= \sum_{i=1}^{n}(y_i - \boldsymbol{\theta}^\mathsf{T}\mathbf{x}_i)^2 + \lambda\sum_{i=1}^{3}\theta_i^2,\\
\text{(ii):}\quad J(\boldsymbol{\theta}) &= \sum_{i=1}^{n}(y_i - \boldsymbol{\theta}^\mathsf{T}\mathbf{x}_i)^2 + \lambda\sum_{i=2}^{3}\theta_i^2,\\
\text{(iii):}\quad J(\boldsymbol{\theta}) &= \sum_{i=1}^{n}(y_i - \boldsymbol{\theta}^\mathsf{T}\mathbf{x}_i)^2 + \lambda\sum_{i=1}^{3}|\theta_i|.
\end{aligned}$$

In Figure 1, you can find three graphs A-C with the learned parameters for different λ. Pair the cost functions (i)-(iii) with the correct graph A-C.

(3p)

(b) In the above cost functions, only the regularization term differs. Design a new regularization term such that the parameter $\boldsymbol{\theta}$ is shrunk towards $\widetilde{\boldsymbol{\theta}} = \begin{bmatrix}1 & 1 & 1\end{bmatrix}^\mathsf{T}$.

(2p)

(c) Explain how the choice of λ typically affects the new data error in terms of the bias variance tradeoff in the cost functions in (a). Which type of error dominates for different values of λ? Explain also briefly a method for selecting a good value for λ when you have access to much more data than the number of parameters you need to learn.

(5p)

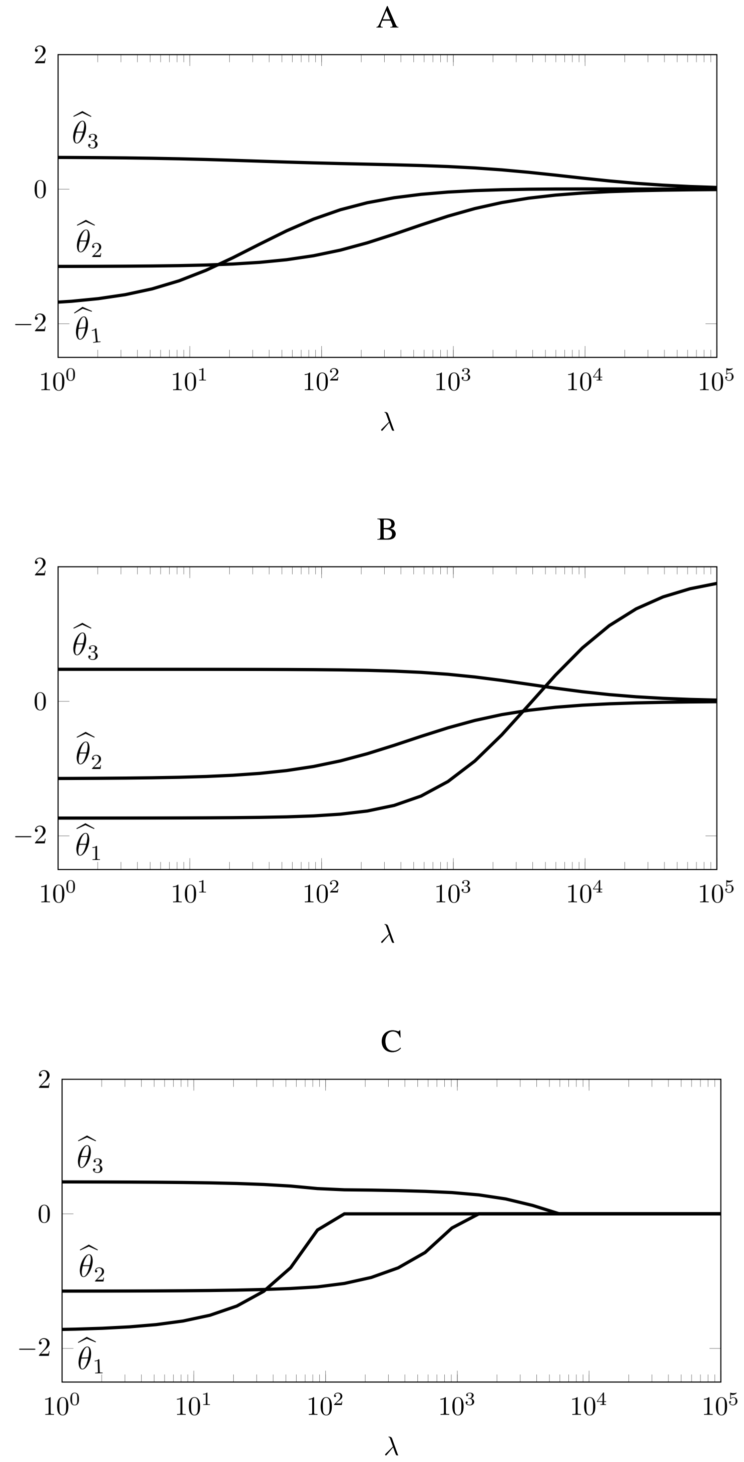

Figure 1: Learned parameter values for different values of the regularization parameter λ

4. Consider the dataset in Table 2.

index	x	y
0	0.0	2.0
1	0.5	1.0
2	1.0	-2.0
3	1.5	-1.0
4	2.0	3.0
5	2.5	1.0
6	3.0	1.0
7	3.5	-1.0
8	4.0	-1.0
9	4.5	1.0

Table 2: Dataset for problem 4

(a) Learn the two models below

i. $\widehat{y} = \theta_1 \sin(\pi x)$

ii. $\widehat{y} = \theta_1 \sin(\pi x) + \theta_2 \cos(\pi x)$

using least squares. Give the estimated parameters.

(3p)

(b) Model ii will never perform worse than model i on training data in terms of squared error loss.. Why?

(2p)

(c) Evaluate the two models using 2-fold cross-validation. Use index $\{0, 1, 2, 3, 4\}$ and $\{5, 6, 7, 8, 9\}$ as your splits. Which of the models do you recommend for in-production use?

Note: Estimate the new data error for both models and use the values to motivate your answer.

(5p)

5. Consider the dataset in Figure 2. We will study binary classification trees with different impurity measures.

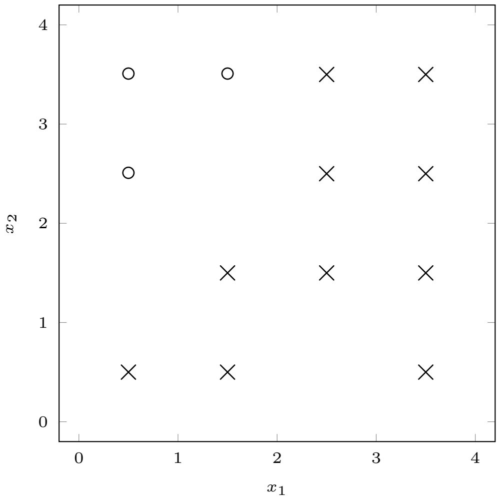

Figure 2: Dataset for problem 5.

(a) Give an optimal binary classification tree with two splits for minimizing the training misclassification loss for the dataset in Figure 2. An intuitive motivation for the optimality is sufficient.

(2p)

(b) Figure 3 shows decision boundaries of binary classification trees constructed using recursive binary splitting with the data in Figure 2 as training data. One has used entropy as impurity measure and one has used misclassification error. Which impurity measure has been used in (a) and which has been used in (b)?

(2p)

(c) Construct a binary classification tree using recursive binary splitting with Gini index as impurity measure for the data in Figure 2. Stop after 3 splits or when all training data points are correctly classified.

(6p)

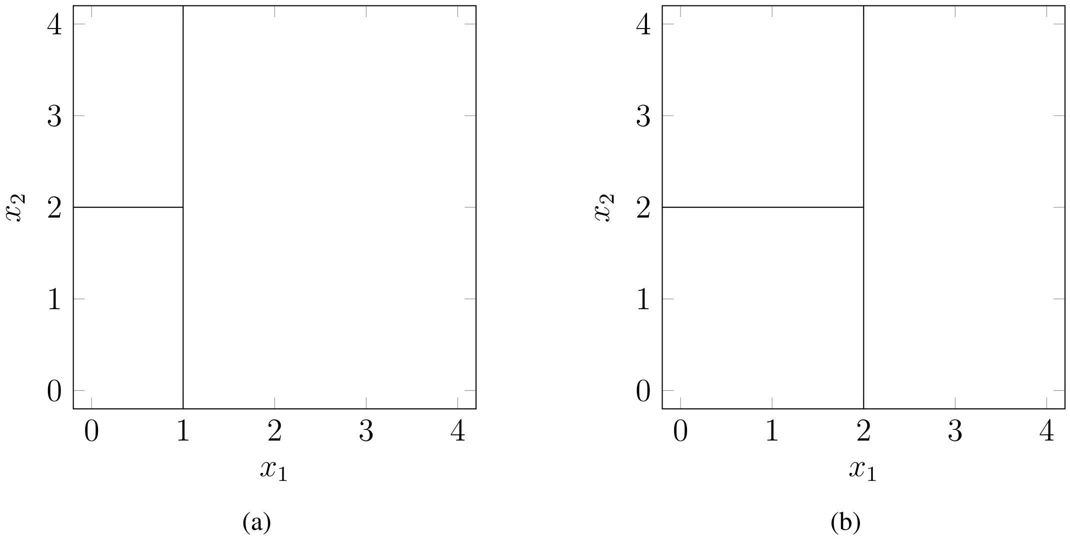

Figure 3: Decision boundaries for problem 5 (b).