EXAM IN STATISTICAL MACHINE LEARNING
STATISTISK MASKININLÄRNING

DATE: August 24, 2023

RESPONSIBLE TEACHER: Dave Zachariah

NUMBER OF PROBLEMS: 5

AIDING MATERIAL: Calculator, mathematical handbook

PRELIMINARY GRADES:

grade 3	23 points
grade 4	33 points
grade 5	43 points

Some general instructions and information:

- Your solutions should be given in *English.*
- Only write on *one* page of the paper.
- Write your exam code and a 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, kept in order).

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!

Formula sheet for Statistical Machine Learning

Warning: This is not a complete list of formulas used in the course, some exam problems may require expressions not listed here. Furthermore, the formulas below are not self-explanatory, you need to be familiar with the expressions to interpret them.

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 $\mathrm{Var}\left(\frac{1}{n}\sum_{i=1}^n z_i\right) = \frac{1-\rho}{n}\sigma^2 + \rho\sigma^2$.

Linear regression and regularization:

- The least-squares estimate of $\boldsymbol{\theta}$ in the linear regression model

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

  is given by the solution $\widehat{\boldsymbol{\theta}}_{\mathrm{LS}}$ to the normal equations $\mathbf{X}^{\mathsf{T}}\mathbf{X}\widehat{\boldsymbol{\theta}}_{\mathrm{LS}} = \mathbf{X}^{\mathsf{T}}\mathbf{y}$, 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 $\lambda\|\boldsymbol{\theta}\|_2^2 = \lambda\sum_{j=0}^p \theta_j^2$.
  The ridge regression estimate is $\widehat{\boldsymbol{\theta}}_{\mathrm{RR}} = (\mathbf{X}^{\mathsf{T}}\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^{\mathsf{T}}\mathbf{y}$.
- 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

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

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

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

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

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

where

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

For quadratic discriminant analysis (QDA), the model is

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

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

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

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 $\widehat{y}(\mathbf{x}) = \text{sign}\{C(\mathbf{x})\}$, 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 and each incorrect answer scores -1 point (capped at 0 for the whole problem). Answers left blank score 0 points.

   Hint: It is often better to only answer statements where you are confident. You do not need to classify all statements.

   i. Logistic regression is a method in which outputs y and inputs x must be categorical (qualitative).

   ii. The standard k-nearest neighbours compares the Euclidean distance between test input x and each training point x_i.

   iii. Learning a model $f(x)$ from a complex family of models is more prone to overfitting to noise than when using a simpler family.

   iv. The estimated new error E_{train} using a learned model typically overestimates the actual new error E_{new}.

   v. A neural network with one hidden layer and a linear activation function is linear in the input x.

   vi. Learning from a family of linear models using the least squares method always yields a unique model.

   vii. Ridge regression penalizes learning models that use many inputs.

   viii. Convolutional neural networks are well suited for classification problems where the input is an image.

   ix. A hospital wants to predict the number of incoming patients to the emergency department. Since the number of patients is an integer, this is best viewed as a classification problem.

   x. By minimizing $J(\boldsymbol{\theta})$ via gradient descent we always learn the globally optimal model parameters.

   (10p)

2. Consider a dataset $\{(x_i, y_i)\}_{i=1}^n$, where x_i represents the input feature, y_i represents a corresponding real-valued output, and n is the total number of data points.

   a) Suppose our choice of linear regression model is $y = \theta_0 + \theta_1 x + \varepsilon$ with model parameters $\boldsymbol{\theta} = \begin{bmatrix}\theta_0 & \theta_1\end{bmatrix}^\top$. The goal is to derive the ordinary least-squares (LS) estimator step-by-step. See the solution to the normal equations in the provided formula sheet. *Hint: some useful vector and matrix expressions are:* $\frac{\partial}{\partial \mathbf{z}} \mathbf{z}^\top A \mathbf{z} = 2A\mathbf{z}$, $\frac{\partial}{\partial \mathbf{z}} \mathbf{a}^\top \mathbf{z} = \mathbf{a}$ *and* $||\mathbf{a}||_2^2 = \mathbf{a}^\top \mathbf{a}$

      i. Write the model as a matrix-valued equation where all n data points are stacked into different rows. Explain the sizes and contents of each variable. (1p)

      ii. Define the predicted output $f(x)$ in terms of model parameters and features. Further, define the objective function $J(\boldsymbol{\theta})$ as mean squared error over all training data (1p)

      iii. Minimize the objective with respect to $\boldsymbol{\theta}$ to obtain the LS estimator (i.e. best values for θ_0 and θ_1).
      Hint: write the objective $J(\boldsymbol{\theta})$ from ii. as matrix-valued equation as a 2-norm. (3p)

   b) When applied to real data, answer the following questions in one sentence each (2p)

      i. How can you interpret the model parameters θ_0 and θ_1?

      ii. How does the data have to be such that the chosen model captures the information in the data?

   c) Let's consider the general case with $\mathbf{x} \in \mathbb{R}^p$ implying $\boldsymbol{\theta} \in \mathbb{R}^p$. We add a penalty term to the objective function such that $J_{new}(\boldsymbol{\theta}) = J(\boldsymbol{\theta}) + \lambda R(\boldsymbol{\theta})$, with $R(\boldsymbol{\theta})$ as either Ridge or Lasso regularization. Which regularization type and strength λ should we use, if we want to select only some features of $\mathbf{x}$? Explain why. (1p)

   d) Can you use a neural network to perform linear regression? If so, sketch the neural network layer(s) with a description of inputs and outputs. If not, then explain why this is not possible. (2p)

3. To complete a marathon (42.2 km) in less than 3 hours is a major milestone for many runners. Given a runner's split time at the halfway mark (21.1 km), we want to be able to predict whether they will complete the race in less than 3 hours ($y = 1$) or not ($y = -1$). We will use real-world data from the Stockholm Marathon 2021 to build our models.

   a) In this part, we consider 10 data points with runner id $= \{1, \ldots, 10\}$ as training data, as shown in Table 1. The halfway split times of these runners range from 82 minutes to 97 minutes.
   Fit a QDA model to the 10 training data points given in Table 1. I.e., compute $\hat{\pi}_m$, $\hat{\mu}_m$, $\hat{\Sigma}_m$ for both classes ($m = 1$ and $m = -1$). (2p)

Runner id	x	y
1	82	1
2	84	1
3	85	1
4	88	1
5	83	-1
6	85	-1
7	87	-1
8	89	-1
9	93	-1
10	97	-1

Table 1: Available data points for QDA.

   b) According to the QDA model, what is the probability that a runner with a split time of 90 minutes finishes the race in less than 3 hours? (3p)

   c) Using a larger training dataset of 100 runners, we fit the following logistic regression model

$$p(y = 1|x; \hat{\theta}) = \frac{e^{\hat{\theta}_0 + \hat{\theta}_1 x}}{1 + e^{\hat{\theta}_0 + \hat{\theta}_1 x}}, \tag{1}$$

   with estimated parameters $\hat{\theta}_0 = 48.9$, $\hat{\theta}_1 = -0.56$.
   According to the logistic regression model, what is the probability that a runner with a split time of 90 minutes finishes the race in less than 3 hours? How about a runner with a split time of 85 minutes? (2p)

d) According to the logistic regression model, how fast of a split time (in minutes) does a runner need in order to have a 50% chance to finish the race in less than 3 hours? (3p)

4. Consider the following training data

i	1	2	3	4	5	6	7	8
x_1	1.0	6.0	7.0	9.0	1.0	4.0	4.0	9.0
x_2	8.0	4.0	9.0	8.0	2.0	1.0	6.0	2.0
y	1	1	1	1	-1	-1	-1	-1

where x_1 and x_2 are the input variables, y is the output and i is an index for the data points.

(a) Illustrate the training data points in a graph with x_1 and x_2 on the two axes. Represent the points belonging to class $y = -1$ with a circle and those belonging to class $y = 1$ with a cross. Further, annotate the data points with their data point indices.

(2p)

(b) Based on the training data we want to construct bagging classification trees with three ensemble members. For this we draw three new datasets by bootstrapping the training data (sampling with replacement). The following data points indices have been drawn for each of the three bootstrapped datasets

	Data point indices i							
Dataset 1	1	2	3	4	5	5	8	8
Dataset 2	1	4	5	5	6	7	7	8
Dataset 3	2	2	3	5	5	6	7	7

Construct three classification trees, one for each of the three bootstrapped datasets. Each tree shall consist of one single binary split that minimizes the misclassification error.

(6p)

(c) The final classifier predicts according to the majority vote of the three classification trees. Sketch the decision boundary of the final classifier.

(2p)

5. (a) Suppose the output is expressed as:

$$y = \underbrace{\theta_0^* + \theta_1^* x}_{=f_0(x)} + \varepsilon, \qquad \varepsilon \sim \mathcal{N}(0, 1^2). \tag{2}$$

Assume that we have a training data set consisting of two data points $\mathcal{T} = \{(x_1, y_1), (x_2, y_2)\} = \{(-2, 1), (1, -2)\}$. What is the least-squares estimate $\widehat{\boldsymbol{\theta}}_{\text{LS}}$ of $\boldsymbol{\theta}$? (2p)

(b) Assuming that the model assumption in (2) is correct, consider the model for prediction

$$f(x; \widehat{\boldsymbol{\theta}}) = \widehat{\theta}_0 + \widehat{\theta}_1 x.$$

Compute the bias and variance in $f(x_\star; \widehat{\boldsymbol{\theta}})$ for a test point $x_\star = -1$, when estimating $\widehat{\boldsymbol{\theta}}$ using least-squares. (3p)

Hint: The model bias is $\mathbb{E}[f(x_\star; \widehat{\boldsymbol{\theta}}) - f_0(x_\star)]$ *and the model variance is* $\mathbb{E}[(f(x_\star; \widehat{\boldsymbol{\theta}}) - \mathbb{E}[f(x_\star)])^2]$.

(c) Assuming that the model assumption in (2) is correct, compute the covariance matrix of the estimate $\widehat{\boldsymbol{\theta}}$. (3p)

Hint 1: Note that we need to take the into account the fact that the training outputs y_1 *and* y_2 *(and thus the estimate* $\widehat{\beta}$*) are random variables whereas* x_1 *and* x_2 *are fixed.*
Hint 2: You may use $Cov[A\mathbf{z}] = A\Sigma A^\mathsf{T}$*, where* A *is fix and* $\mathbf{z}$ *is a random variable with* $\Sigma = Cov[\mathbf{z}]$.

(d) What is the ridge regression estimate $\widehat{\boldsymbol{\theta}}_{\text{RR}}$ of $\boldsymbol{\theta}$? Express your solution in terms of the regularization parameter λ and simplify as much as possible. (2p)