DATE AND TIME: March 10, 2017, 8.00–13.00
RESPONSIBLE TEACHER: Fredrik Lindsten
NUMBER OF PROBLEMS: 5
AIDING MATERIAL: Calculator, mathematical handbooks
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.
• 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).
• When asked to pair e.g. plots with corresponding formulas, the order of the plots/formulas is always randomly generated using the function sample(n) in R, where n is the number of options. Thus, it is not possible to infer the correct answer from the way in which the problem is presented.

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 useful 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!* Consequently, some of the problems may require knowledge about certain expressions not listed here. Furthermore, the formulas listed below are not all self-explanatory, meaning that you need to be familiar with the expressions to be able to interpret them. Thus, the list should be viewed as a support for solving the problems, rather than as a comprehensive collection of formulas.

**Marginalization and conditioning of probability densities:** For a partitioned random vector \( Z = (Z_1^T \ Z_2^T)^T \) with joint probability density function \( p(z) = p(z_1, z_2) \), the marginal probability density function of \( Z_1 \) is

\[
p(z_1) = \int_{Z_2} p(z_1, z_2) \, dz_2
\]

and the conditional probability density function for \( Z_1 \) given \( Z_2 = z_2 \) is

\[
p(z_1 \mid z_2) = \frac{p(z_1, z_2)}{p(z_2)} = \frac{p(z_2 \mid z_1)p(z_1)}{p(z_2)}.
\]

**The Gaussian distribution:** The probability density function of the \( p \)-dimensional Gaussian distribution is

\[
\mathcal{N}(x \mid \mu, \Sigma) = \frac{1}{(2\pi)^{p/2}\sqrt{\det \Sigma}} \exp \left( -\frac{1}{2} (x - \mu)^T \Sigma^{-1} (x - \mu) \right), \quad x \in \mathbb{R}^p.
\]

For a Gaussian random vector \( X \sim \mathcal{N}(\mu, \Sigma) \) partitioned according to,

\[
X = \begin{pmatrix} X_a \\ X_b \end{pmatrix}, \quad \mu = \begin{pmatrix} \mu_a \\ \mu_b \end{pmatrix}, \quad \Sigma = \begin{pmatrix} \Sigma_a & \Sigma_{ab} \\ \Sigma_{ab}^T & \Sigma_b \end{pmatrix}
\]

it holds that the marginal probability density of \( X_a \) is \( p(x_a) = \mathcal{N}(x_a \mid \mu_a, \Sigma_a) \) and the conditional density of \( X_a \) given \( X_b = x_b \) is \( p(x_a \mid x_b) = \mathcal{N}(x_a \mid \mu_{a|b}, \Sigma_{a|b}) \), where

\[
\begin{align*}
\mu_{a|b} &= \mu_a + \Sigma_{ab}\Sigma_b^{-1}(x_b - \mu_b), \\
\Sigma_{a|b} &= \Sigma_a - \Sigma_{ab}\Sigma_b^{-1}\Sigma_{ab}^T.
\end{align*}
\]
If we have that \(X_a\), as well as \(X_b\) conditioned on \(X_a = x_a\), are Gaussian distributed according to
\[
p(x_a) = \mathcal{N}(x_a | \mu_a, \Sigma_a),
\]
\[
p(x_b | x_a) = \mathcal{N}(x_b \mid Mx_a + b, \Sigma_{b|a}),
\]
where \(M\) is a matrix (of appropriate dimension) and \(b\) is a constant vector, then the joint distribution of \(X_a\) and \(X_b\) is given by
\[
p(x_a, x_b) = \mathcal{N}
\begin{pmatrix}
x_a \\
x_b
\end{pmatrix}
\mid
\begin{pmatrix}
\mu_a \\
M\mu_a + b
\end{pmatrix},
\begin{pmatrix}
\Sigma_a & \Sigma_a M^T \\
M\Sigma_a & \Sigma_a + MMM^T
\end{pmatrix}.
\]

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

Linear regression and regularization:

- The least-squares estimate of \(\beta\) in the linear regression model \(Y = \beta^T X + \varepsilon\) is given by \(\hat{\beta}_{LS} = (X^T X)^{-1} X^T y\), where
\[
X = \begin{pmatrix}
x_1^T \\
\vdots \\
x_N^T
\end{pmatrix}, \quad \text{and} \quad y = \begin{pmatrix}
y_1 \\
\vdots \\
y_N
\end{pmatrix}.
\]

- Ridge regression uses the regularization term \(\lambda \|\beta\|^2_2 = \lambda \sum_{j=0}^{p} \beta_j^2\). The ridge regression estimate is \(\hat{\beta}_{RR} = (X^T X + \lambda I)^{-1} X^T y\).

- LASSO uses the regularization term \(\lambda \|\beta\|_1 = \lambda \sum_{j=0}^{p} |\beta_j|\). (The LASSO estimate does not admit a simple closed form expression.)

- For a probabilistic linear regression model with \(\varepsilon \sim \mathcal{N}(0, \sigma^2)\) and prior distribution \(p(\beta) = \mathcal{N}(\beta | \mu_0, \Sigma_0)\) the posterior distribution is \(p(\beta | y) = \mathcal{N}(\beta | \mu_N, \Sigma_N)\) with
\[
\mu_N = \Sigma_N \left(\Sigma_0^{-1} \mu_0 + \sigma^{-2} X^T y\right), \quad \Sigma_N = \left(\Sigma_0^{-1} + \sigma^{-2} X^T X\right)^{-1}.
\]

Maximum likelihood: The maximum likelihood estimate is given by \(\hat{\beta}_{ML} = \arg\max_{\beta} \ell(\beta)\) where \(\ell(\beta) = \log p(y | \beta) = \sum_{i=1}^{N} \log p(y_i | \beta)\) is the log-likelihood.
function (the last equality holds when the \( N \) training data points are independent).

**Logistic regression:** The logistic regression model uses a linear regression for the the log-odds. In the binary classification context we thus have

\[
\log \left( \frac{\Pr(Y = 1 | X)}{\Pr(Y = 0 | X)} \right) = \beta^T X.
\]

**Discriminant Analysis:** The linear discriminant analysis (LDA) classifier assigns a test input \( X = x \) to class \( k \) for which,

\[
\hat{\delta}_k(x) = x^T \hat{\Sigma}_k^{-1} \hat{\mu}_k - \frac{1}{2} \hat{\mu}_k^T \hat{\Sigma}_k^{-1} \hat{\mu}_k + \log \hat{\pi}_k
\]

is largest, where \( \hat{\pi}_k = \frac{N_k}{N} \) and \( \hat{\mu}_k = \frac{1}{N_k} \sum_{i:y_i = k} x_i \) for \( k = 1, \ldots, K \), and \( \hat{\Sigma} = \frac{1}{N-K} \sum_{k=1}^K \sum_{i:y_i = k} (x_i - \hat{\mu}_k)(x_i - \hat{\mu}_k)^T \).

For quadratic discriminant analysis (QDA) we instead use the discriminant functions

\[
\hat{\delta}_k(x) = -\frac{1}{2} \log |\hat{\Sigma}_k| - \frac{1}{2} (x - \hat{\mu}_k)^T \hat{\Sigma}_k^{-1} (x - \hat{\mu}_k) + \log \hat{\pi}_k,
\]

where \( \hat{\Sigma}_k = \frac{1}{N_k-1} \sum_{i:y_i = k} (x_i - \hat{\mu}_k)(x_i - \hat{\mu}_k)^T \) for \( k = 1, \ldots, K \).

**Loss functions for classification:**

- Misclassification loss: \( I(y \neq G(x)) \).
- Exponential loss: \( \exp(-yC(x)) \) where \( G(x) = \text{sign}(C(x)) \).

**Gaussian processes:** If the function \( f \) is distributed according to a Gaussian process, \( f \sim \mathcal{GP}(m, k) \), it holds that for any arbitrary selection of inputs \( \{x^{(1)}, x^{(2)}, \ldots, x^{(n)}\} \) the output values \( f(x^{(1)}), f(x^{(2)}), \ldots, f(x^{(n)}) \) are jointly Gaussian,

\[
\begin{pmatrix}
    f(x^{(1)}) \\
    \vdots \\
    f(x^{(n)})
\end{pmatrix} \sim \mathcal{N}
\begin{pmatrix}
    m(x^{(1)}) \\
    \vdots \\
    m(x^{(n)})
\end{pmatrix},
\begin{pmatrix}
    k(x^{(1)}, x^{(1)}) & \cdots & k(x^{(1)}, x^{(n)}) \\
    \vdots & \ddots & \vdots \\
    k(x^{(n)}, x^{(1)}) & \cdots & k(x^{(n)}, x^{(n)})
\end{pmatrix}.
\]

For a Gaussian process regression model \( Y = f(X) + \varepsilon \) with \( f \sim \mathcal{GP}(m, k) \) and \( \varepsilon \sim \mathcal{N}(0, \sigma^2) \), the prediction model is given by

\[
f(x_*) | y \sim \mathcal{N}
\left(m(x_*) + s^T (y - m(X)), k(x_*, x_*) - s^T k(X, x_*) \right),
\]

where \( s^T = k(x_*, X)(k(X, X) + \sigma^2 I_N)^{-1} \).
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).

i. Regression problems have only quantitative inputs.

ii. The following (so called “probit”) classifier is linear:

\[ \hat{G}(x) = I(\Phi(\beta^T x) > 0.2) \]

where \( \Phi(x) = \int_{-\infty}^{x} N(z | 0, 1)dz \) is the cumulative distribution function of the standard Gaussian distribution, \( I(\cdot) \) is the indicator function, and the class labels are 0 and 1.

iii. Ensemble methods can be used to reduce both bias and variance (compared to the base model used).

iv. The input partitioning shown in Figure 1 could not have been generated by recursive binary splitting.

v. The model \( Y = \beta_0 + \beta_1 X_1 + \beta_0\beta_1 X_2 + \varepsilon \) (where \( \beta_0 \) and \( \beta_1 \) are model parameters) is a linear regression model.

vi. Over-fitting for a k-NN classifier occurs when there is too much data, so that the k-neighborhoods become too localized in the input space.

Figure 1: Input partitioning for Problem 1.iv.
vii. Maximum likelihood estimation is another word for least-squares estimation.

viii. The mean-squared-error (in \( \hat{f}(X; \mathcal{T}) \) w.r.t. \( f(X) \)) can be decomposed into the sum of the squared model bias and the model variance, i.e.

\[
E[(\hat{f}(X; \mathcal{T}) - f(X))^2] = \left( E[\hat{f}(X; \mathcal{T}) - f(X)] \right)^2 + E\left[\left( \hat{f}(X; \mathcal{T}) - E[\hat{f}(X; \mathcal{T})] \right)^2 \right].
\]

ix. AdaBoost can use logistic regression as a base classifier.

x. The \( k \)-NN classifier is sensitive to the scaling of the input variables.
2. A large Swedish retailer of alcoholic beverages has asked you to build a model for predicting the price of a wine based on various types of information, such as the wine’s chemical composition, region, etc. They have collected a database with their data, containing the following columns:

- **id** – a unique identification number for each row of the database, specified as an integer value
- **grape** – one out of 6 different grapes (e.g. Syrah, Zinfandel, Merlot, etc.)
- **alcohol** – percentage of alcohol in the wine, specified as a real number between 0 and 1 (e.g. 0.135 for 13.5%)
- **year** – production year of the wine, specified as an integer in the range 2010–2016
- **region** – one out of 78 different wine regions (e.g. Bordeaux, Burgundy, Rhône, etc.)
- **proline** – concentration of the amino acid proline in mg/l specified as an integer (typically in the range 200–2000)
- **timestamp** – a time stamp specifying the time when the row was entered into the database, on the format ‘YYYY-MM-DD HH:MM:SS’
- **price** – the current price of the wine in SEK, specified as an integer (typically in the range 80–400)

*Don’t forget to clearly motivate all your answers!*

(a) The customer want’s to try a simple model first, like a linear regression or a logistic regression. Which one of these two methods do you suggest? (1p)

(b) For each column of the customer’s database as listed above, specify whether you would consider that variable as an input of the model, an output of the model, or neither. (4p)

(c) For each of the inputs and outputs of your model (from the previous question), specify whether that variable is best viewed as quantitative or qualitative. (3p)

(d) The customer has previously tried to use a CART\(^1\) model for this problem, but ran into problems when trying to learn this model. In fact, they were not able to train the CART model at all! The

\(^1\text{CART}=\text{Classification And Regression Trees}\)
issue was traced to the variable \textit{region}. Considering specifically this variable, the splitting criteria they considered was to compute

$$\hat{I} = \arg \min_{I \subset R} \sum_{i : x_i \in I} L(y_i, \hat{c}_1) + \sum_{i : x_i \not\in I} L(y_i, \hat{c}_2)$$

where $L$ is a given loss function and $\hat{c}_1$ and $\hat{c}_2$ are constant predictions for the two regions obtained in the split. Furthermore, $R$ is the set of all possible values for \textit{region}, i.e.

$$R = \{\text{Bordeaux, Burgundy, Rhône, \ldots}\}$$

and the optimization is with respect to all proper subsets $I \subset R$.

The CART implementation they used was based on a brute-force solution of the optimization problem above, i.e. all possible values of $I$ are enumerated, the loss is computed, and $\hat{I}$ is taken as the value of $I$ which attains the smallest loss. Why did they not succeed in this approach? (2p)
3. (a) Consider the scatter plots in Figure 2 which depict three different training data sets for three binary classification problems. In which dataset(s) could the classes be well separated by...

- ...an LDA classifier?
- ...a QDA classifier?

(In both cases we assume that $X_1$ and $X_2$ are the only inputs to the classifiers.) (2p)

![Dataset i](image1)
![Dataset ii](image2)
![Dataset iii](image3)

Figure 2: Scatter plots of training data for Problems 3a and 3b.

(b) Consider again the scatter plots in Figure 2. The LDA and QDA classifiers are based on different assumptions about the properties of the data. Which dataset(s) in Figure 2 appear to correspond well to the assumptions made by LDA and QDA, respectively? (4p)

(c) Consider a Gaussian process regression model $Y = f(X) + \varepsilon$ with $\varepsilon \sim \mathcal{N}(0, \sigma^2)$ and $f \sim \mathcal{GP}(0, k(x, x'))$. Five different GP models are fit to a given training data set $T = \{(x_i, y_i)\}_{i=1}^{3}$.

**M1.** squared-exponential kernel, length scale $\ell = 0.1$; noise variance $\sigma^2 = 0.3^2$

**M2.** squared-exponential kernel, length scale $\ell = 1$; noise variance $\sigma^2 = 3^2$

**M3.** periodic kernel, length scale $\ell = 1$ and period $p = 3$; noise variance $\sigma^2 = 0.3^2$

**M4.** Matérn-3 kernel, length scale $\ell = 5$; noise variance $\sigma^2 = 0.3^2$

**M5.** Matérn-3 kernel, length scale $\ell = 1$; noise variance $\sigma^2 = 0$

The kernels for the models M1-M5 are plotted in Figure 3. The four panels of Figure 4 show sample paths from the posterior distributions of $f(x) \mid T$ for four out of these five models. Specify which model that belongs to which panel in the figure. (4p)
Figure 3: Kernels $k(r) = k(|x - x'|)$ for the models in Problem 3c.

Figure 4: Sample paths from the posterior Gaussian process for four of the models in Problem 3c. The circles mark the three training data points.
4. (a) Consider the simple linear regression model with one input variable $X$,

$$Y = f(X) + \varepsilon = \beta_0 + \beta_1 X + \varepsilon, \quad \varepsilon \sim \mathcal{N}(0, \sigma^2).$$

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

(b) What is the ridge regression estimate $\hat{\beta}_{RR}$ of $\beta$? Express your solution in terms of the regularization parameter $\lambda$. (2p)

(c) What is the LASSO estimate $\hat{\beta}_{LASSO}$ of $\beta$? Express your solution in terms regularization parameter $\lambda$. (6p)

*Hint: In two dimensions, the LASSO estimate of $\beta_j$ is either 0, or has the same sign as the least-squares estimate of $\beta_j$. (This does not hold in higher dimensions.) Furthermore, the LASSO estimate varies continuously with $\lambda$ in a piecewise linear way.*
5. (a) Explain briefly the difference between parametric models and non-parametric models. (2p)

(b) Consider a logistic regression model for a binary classification problem, \( Y \in \{0,1\} \), with log-odds for class 1 given by \( \tilde{\beta}^T X \) where

\[
\tilde{\beta}^T = \begin{pmatrix} \beta_0 & \beta_1 & \beta_2 & \beta_3 \end{pmatrix} = \begin{pmatrix} 3 & -1 & 3 & 2 \end{pmatrix},
\]

where the parameter \( \beta_0 \) corresponds to an offset (“intercept”) term. According to this model, what is the probability that the test input \( x_* = \begin{pmatrix} 2 & 1 & -1 \end{pmatrix} \) belongs to class 0? (2p)

(c) Consider a logistic regression model for a binary classification problem \( Y \in \{0,1\} \). We have observed a training data set \( \mathcal{T} = \{(x_i, y_i)\}_{i=1}^N \) where the \( N \) training data points are mutually independent. Write down an expression for the likelihood, i.e. the probability of the observed training data

\[
\Pr(Y_1 = y_1, \ldots, Y_N = y_N \mid X_1 = x_1, \ldots, X_N = x_N, \beta)
\]

expressed in terms of the class-1 probabilities

\[
p(x, \beta) := \Pr(Y = 1 \mid X = x, \beta).
\]

(4p)

(d) For the logistic regression model above, write down an expression for the log-likelihood function

\[
\ell(\beta) = \log \Pr(Y_1 = y_1, \ldots, Y_N = y_N \mid X_1 = x_1, \ldots, X_N = x_N, \beta)
\]

on the form,

\[
\ell(\beta) = \sum_{i=1}^N h(x_i, y_i, \beta).
\]

The function \( h \) should be expressed only in terms of elementary functions (like additions, multiplications, logarithms, exponentials, \ldots) and its dependence on the variables \( x_i, y_i \) and \( \beta \) should be explicit. (2p)