Written Exam
Introduction to Machine Learning (DM825)

Institute for Mathematics and Computer Science
University of Southern Denmark

Wednesday, June 22, 2010, 9:00–12:00, U49

All usual helping tools (textbooks, lecture notes, etc.) together with pocket calculators are allowed. It is not allowed to use computers, smart-phones and personal digital assistants.

The exam consists of 5 tasks and relative subtasks distributed on 10 pages.

The weight in the evaluation of each task and subtask is given in points. The total sum is 100 points. More points have been assigned to tasks that require fundamental knowledge that the course aimed to transmit. Points are not necessarily representative of the difficulty of the task.

Remember to justify all your statements. You may refer to results from the textbooks or the lecture notes in the syllabus. In particular, it is possible to justify a statement by saying that it derives trivially from a result in the textbook (if this is true!). You may use all methods or extensions that have been used in the assignment sheets, published during the course. However, it is not allowed to answer a subtask exclusively by reference to an exercise seen during the course. Reference to other books (outside the course material) is not accepted as answer to a task!

You may write your answers in Danish or in English.

Task 1 Boosting (10 points)

Figure 1 shows positive and negative examples in a two dimensional feature space. The figure also shows the normalized weights on the examples resulting from having run the AdaBoost algorithm for some number of iterations. There are also three decision boundaries drawn in the figure, $h(\mathbf{x};\theta_A), h(\mathbf{x};\theta_B)$, and $h(\mathbf{x};\theta_C)$ or A, B and C for short.

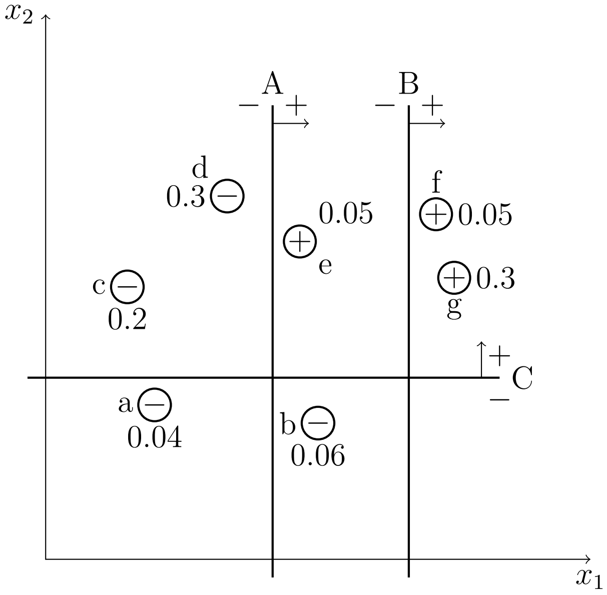


Figure 1: Points with identifiers and weights, and decision boundaries for Task 1.

Subtask 1.a (5 points)

Using the weighted misclassification error which one of the region splits would you use at the next iteration (please answer A, B, or C)? Briefly justify your answer.

Subtask 1.b (5 points)

Which training point(s) (possibly none) the ensemble $h_2(\mathbf{x}) = h(\mathbf{x};\theta_A) + h(\mathbf{x};\theta_C)$ cannot classify correctly?

Task 2 Exponential family and geometric distribution (20 points)

A way to solve constraint satisfaction problems is by complete tree search. In other courses, we saw that random restart of the solver may reduce the time for solving a specific problem instance. Let $y_j = 1, 2, 3, ...$ be the number of times we need to restart the solver in a specific run j before being able to solve the given instance. For each run of the solver we know the features of the instance to solve (eg, size, density and type of constraints, etc.) and the heuristics used in the search procedure. We use this information to construct for each run j a feature vector $\mathbf{x}_j$. On the basis of the results collected $y_1, y_2, y_3, \ldots$ and the corresponding feature vectors $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \ldots$ we could learn to predict how many times we need to restart the solver in a particular run.
The probability that the first occurrence of a success requires k number of independent trials, each with success probability ϕ, is $p(Y = y, \phi) = (1 - \phi)^{y-1}\phi, y = 1, 2, 3, \ldots$. This distribution is known as the *geometric distribution* and it seems well suited to model $y|\mathbf{x}$ in our learning task.

Subtask 2.a (10 points)

Show that the geometric distribution is in the exponential family

$$p(y|\eta) = b(y)g(\eta)\exp\{\eta^T u(y)\}$$

by giving $b(y)$, $g(\eta)$, η and $u(y)$.

Subtask 2.b (5 points)

Consider performing regression using a GLM model with a geometric response variable. What is the canonical response function for the family? You may use the fact that the mean of a geometric distribution is given by $1/\phi$.

Subtask 2.c (5 points)

For a training set $(\mathbf{x}^j, y^j); j = 1, \ldots, m$, let the log-likelihood of an example be $\log p(y^j|\mathbf{x}^j, \boldsymbol{\theta})$. By taking the derivative of the log-likelihood with respect to θ_i, derive the stochastic gradient ascent rule for learning using a GLM model with geometric responses y. Show that this rule depends on the training responses y^j and their predicted value through the canonical response function.

Task 3 SVM (20 points)

Consider a classification problem on m labelled training points in a two-dimensional input space that we want to separate with a linear classifier through origin.
The following is a condensed description of the SVM procedure as given in class and in the text book [B2]. (You can go directly to the subtasks if you remember it but please refer to the given equation numbers in the justifications to your answers.)

The discriminant function is a hyperplane $\{x : \boldsymbol{\theta}^T \mathbf{x} = 0\}$ with $\theta_0 = 0$ and the classification rule $h(\mathbf{x}, \boldsymbol{\theta}) = \text{sign}(\boldsymbol{\theta}^T \mathbf{x})$. The parameters $\boldsymbol{\theta}$ are learned by solving the following optimization problem

$$\textbf{Primal}: \quad \underset{\boldsymbol{\theta},\boldsymbol{\xi}}{\text{Minimize}} \quad \frac{1}{2}||\boldsymbol{\theta}||^2 + C\sum_{j=1}^{m} \xi_j \tag{1}$$

$$\text{subject to} \quad y^j(\boldsymbol{\theta}^T \cdot \mathbf{x}^j) \geq 1 - \xi_j, \quad j = 1, \ldots, m \tag{2}$$

$$\xi_j \geq 0, \quad j = 1, \ldots, m \tag{3}$$

To solve Primal we Lagrange relax the constraints in the objective function with multipliers $\alpha_j \geq 0$ and $\mu_j \geq 0, j = 1, \ldots, m$:

$$L_P = \frac{1}{2}||\boldsymbol{\theta}||^2 + C\sum_{j=1}^{m} \xi_j - \sum_{j=1}^{m} \alpha_j \left[y^j(\boldsymbol{\theta}^T \cdot \mathbf{x}^j) - (1 - \xi_j)\right] - \sum_{j=1}^{m} \mu_j \xi_j$$

Setting the derivatives of L_P in θ_i, $i = 1, 2$ and $\xi_j, j = 1, \ldots, m$ to zero we get

$$\theta_i = \sum_{j=1}^{m} \alpha_j y^j x_i^j, \quad i = 1, 2 \tag{4}$$

$$\alpha_j = C - \mu_j, \quad j = 1, \ldots, m \tag{5}$$

Substituting in L_P we get L_D, a lower bound to L_P that we wish to maximize. This is done in the Lagrangian dual problem

$$\textbf{Dual}: \quad \underset{\boldsymbol{\alpha}}{\text{Maximize}} \quad \sum_{j=1}^{m} \alpha_j - \frac{1}{2}\sum_{j=1}^{m}\sum_{l=1}^{m} \alpha_j \alpha_l y^j y^l \mathbf{x}^j \cdot \mathbf{x}^l \tag{6}$$

$$\text{subject to} \quad 0 \leq \alpha_j \leq C, \quad j = 1, \ldots, m \tag{7}$$

In addition, for the solution of Primal we have the KKT conditions

$$\alpha_j \left[y^j(\boldsymbol{\theta}^T \cdot \mathbf{x}^j) - (1 - \xi_j)\right] = 0, \quad j = 1, \ldots, m \tag{8}$$

$$\mu_j \xi_j = 0, \quad j = 1, \ldots, m \tag{9}$$

$$y^j(\boldsymbol{\theta}^T \cdot \mathbf{x}^j) - (1 - \xi_j) \geq 0, \quad j = 1, \ldots, m \tag{10}$$

$$\xi_j \geq 0, \quad j = 1, \ldots, m \tag{11}$$

Subtask 3.a (5 points)

If the points in the training set are not linearly separable we can use basis expansion. Let ϕ be a feature mapping but suppose that $\phi(\mathbf{x})$ is so high-dimensional that it is infeasible to compute it explicitly. Describe briefly how you would apply the "Kernel trick" to *learn* and *predict* in the high-dimensional feature space ϕ, but without ever explicitly computing $\phi(\mathbf{x})$. Indicate precisely which computations can be done efficiently (there are two).

Subtask 3.b (5 points)

Consider the kernel

$$K(\mathbf{x}, \mathbf{z}) = \mathbf{x} \cdot \mathbf{z} + 4(\mathbf{x} \cdot \mathbf{z})^2$$

where the vectors $\mathbf{x}$ and $\mathbf{z}$ are 2-dimensional. This kernel is equal to an inner product $\phi(\mathbf{x}) \cdot \phi(\mathbf{z})$ for some definition of ϕ. What is the feature mapping ϕ?

Subtask 3.c (10 points)

Figure 2 shows both decision boundaries and support vectors (circled) from different SVM-like training methods. In all cases, the boundaries correspond to $\boldsymbol{\theta}^T \cdot \mathbf{x} + \theta_0$, where $\theta_0 = 0$ unless θ_0 is included in the training method. Below there are four methods. The number of figures is instead five. Please assign each method to all the figures that they could potentially produce (there may be multiple choices and some figure may remain unassigned).

Method I:

$$\begin{aligned} \textbf{Primal:} \quad & \underset{\boldsymbol{\theta},\boldsymbol{\xi}}{\text{Minimize}} \quad \frac{1}{2}||\boldsymbol{\theta}||^2 + C\sum_{j=1}^{m}\xi_j \\ & \text{subject to} \quad y^j(\boldsymbol{\theta}^T \cdot \mathbf{x}^j + \theta_0) \geq 1 - \xi_j, \quad j = 1, \ldots, m \\ & \qquad\qquad\quad \xi_j \geq 0, \quad j = 1, \ldots, m \\ & \qquad\qquad\quad \text{where } C = \infty. \end{aligned}$$

Method II:

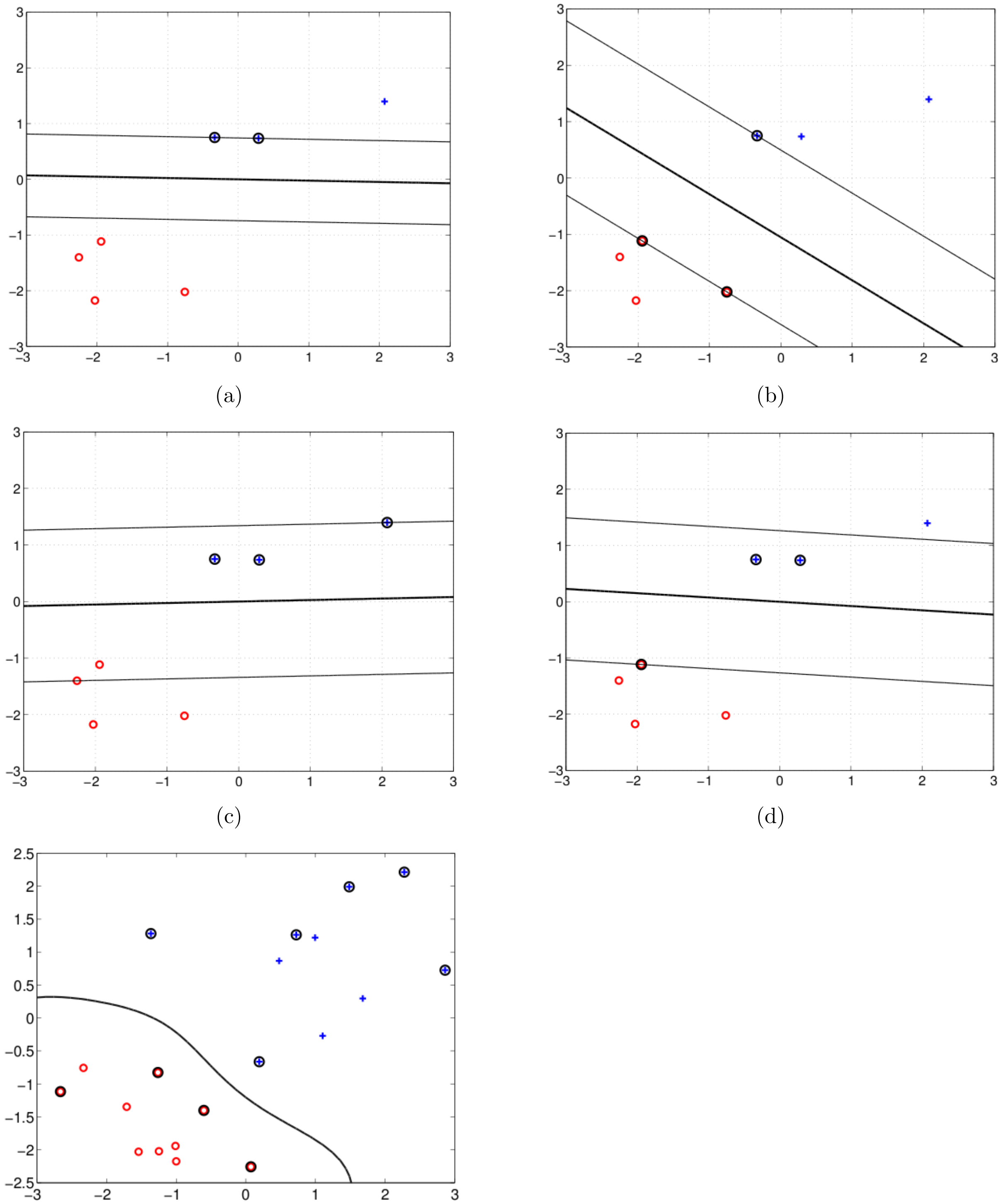
$$\begin{aligned} \textbf{Primal:} \quad & \underset{\boldsymbol{\theta},\boldsymbol{\xi}}{\text{Minimize}} \quad \frac{1}{2}||\boldsymbol{\theta}||^2 + C\sum_{j=1}^{m}\xi_j \\ & \text{subject to} \quad y^j(\boldsymbol{\theta}^T \cdot \mathbf{x}^j) \geq 1 - \xi_j, \quad j = 1, \ldots, m \\ & \qquad\qquad\quad \xi_j \geq 0, \quad j = 1, \ldots, m \\ & \qquad\qquad\quad \text{where } C = \infty. \end{aligned}$$

Method III:

$$\begin{aligned} \textbf{Primal:} \quad & \underset{\boldsymbol{\theta},\boldsymbol{\xi}}{\text{Minimize}} \quad \frac{1}{2}||\boldsymbol{\theta}||^2 + C\sum_{j=1}^{m}\xi_j \\ & \text{subject to} \quad y^j(\boldsymbol{\theta}^T \cdot \mathbf{x}^j) \geq 1 - \xi_j, \quad j = 1, \ldots, m \\ & \qquad\qquad\quad \xi_j \geq 0, \quad j = 1, \ldots, m \\ & \qquad\qquad\quad \text{where } C = 1. \end{aligned}$$

Method IV:

$$\begin{aligned} \textbf{Dual}: \quad & \underset{\boldsymbol{\alpha}}{\text{Maximize}} \quad \sum_{j=1}^{m}\alpha_j - \frac{1}{2}\sum_{j=1}^{m}\sum_{l=1}^{m}\alpha_j\alpha_l y^j y^l K(\mathbf{x}^j, \mathbf{x}^l) \\ & \text{subject to} \quad 0 \leq \alpha_j \leq C, \quad j = 1, \ldots, m \\ & \qquad\qquad\quad \text{where } K(\mathbf{x}^j, \mathbf{x}^l) = \exp(-1/2||\mathbf{x}^j - \mathbf{x}^l||^2) \end{aligned}$$



(a)

(b)

(c)

(d)

(e)

Figure 2: Plots of $\boldsymbol{\theta}^T\phi(\mathbf{x}) = 0$ for different training methods along with the support vectors. In addition we show the lines $\boldsymbol{\theta}^T\phi(\mathbf{x}) = 1$ and $\boldsymbol{\theta}^T\phi(\mathbf{x}) = -1$. Support vectors have bold circles surrounding them.

Task 4 Probabilistic graphical models (40 points)

In this task, we use probabilistic graphical models to tune heuristic algorithms for solving combinatorial optimization problems. Let's consider the case of a local search heuristic in which we have to choose two components: an initial solution and a neighborhood. The table on the left in Figure 3 shows the details of the components. We will identify the construction heuristic by the variable X_1, the local search by X_2, and their corresponding choices by x_k and x_l, respectively. The probabilistic graphical model is depicted in Figure 3, left.

Factor	Type	Levels
construction heuristics	categorical	{`nn`, `nearest_insertion`, `farthest_insertion`, `cheapest_insertion`, `arbitrary_insertion`}
local search	categorical	{`none`, `2-opt`, `linkern`}

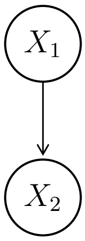


Figure 3: On the left the variables and the values they can take. On the right the probabilistic graphical model.

We use the model to predict the configuration that will perform best. The tuning algorithm works as follows. At each iteration N configurations are sampled from the joint probability distribution $p(X_1, X_2)$. These sampled configurations are then run on a number of instances and the best $\rho \cdot N$, $0 < \rho < 1$ are selected. The model is updated on the basis of the selected best configurations to learn the parameters that will increase the probability of sampling these best configurations.

Subtask 4.a (3 points)

Define the parameterized probability distributions for X_1 and $X_2|X_1$.

Subtask 4.b (6 points)

After the first iteration the $\rho \cdot N = 5$ best configurations out of the $N = 100$ sampled are

```
                     nn-linkern
                     nn-linkern
arbitrary_insertion-linkern
arbitrary_insertion-linkern
arbitrary_insertion-linkern
```

These are to be considered joint realizations of the variables X_1 and X_2 that explain good performance and that therefore we want to learn. For example, `nn-linkern` indicates that the event $(X_1 = \texttt{nn}, X_2 = \texttt{linkern})$ is likely to occur when the configuration is good and we want our model to increase the chances of sampling it.

Let $\mathcal{D}$ denote this data base of five observations. Calculate the maximum likelihood estimate of $P(X_2 = \texttt{linkern}|X_1 = \texttt{nn}, \mathcal{D})$.

Subtask 4.c (3 points)

Following the frequentist approach of the previous point what would be the probability of the configuration `nearest_insertion-linkern` on the basis of the five observations? Indicate which problem this solution exhibits and sketch a repair procedure.

Subtask 4.d (6 points)

In a full Bayesian approach the uncertainty of the parameters of the distributions of X_1 and $X_2|X_1$ are captured by prior probability distributions. Define these probabilities in some hyperparameters for the parameters that you introduced in the first subtask of this task.

Subtask 4.e (12 points)

Show how to derive the value of

$$P(X_2 = \texttt{linkern}|X_1 = \texttt{nn}, \mathcal{D})$$

where $\mathcal{D}$ are the same five observations of point Subtask 4.b. In the numerical computations, assume that all initial local prior distributions are uniform distributions, that is, all values for $p(X_1 = x_{1k})$ and $p(X_2 = x_{2k}|X_1 = x_{1k})$ are equally likely.

Subtask 4.f (6 points)

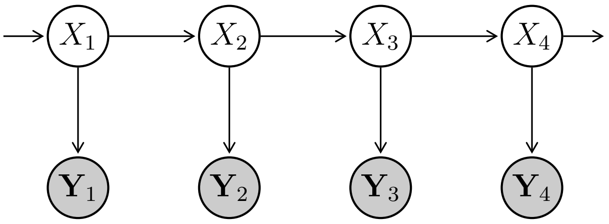
What is the configuration most likely to perform best after the learning phase with the data base $\mathcal{D}$ and what is its probability?

Subtask 4.g (4 points)

Without carrying out the numerical computations, derive $p(X_2 = \texttt{linkern}|\mathcal{D})$.

Task 5 Hidden Markov Models (10 points)

Consider a homogeneous HMM with four underlying states and a two dimensional space as illustrated in Figure 4. Also shown in the figure are the initial state distribution $\boldsymbol{\pi}_0$, permitted state transitions $\mathbf{A}$, and the Gaussian emission distributions. The emission distributions $p(\mathbf{y}|x, \boldsymbol{\mu}_x, \sigma^2 \cdot I) = N(\mathbf{y}; \boldsymbol{\mu}_x, \sigma^2 \cdot I)$ for $x = \{1, 2, 3, 4\}$ share the same overall variance parameter σ^2.
If we were to sample a sequence of the two dimensional space emissions $\mathbf{y}$ from this HMM model, we would get $\mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3, \ldots$ (first time point is $t = 1$).



$$\boldsymbol{\pi}_0 = [0.5, 10^{-9}, 10^{-9}, 0.5] \quad \mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

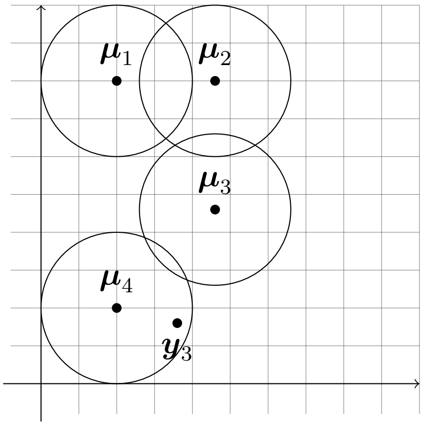


Figure 4: The Hidden Markov Model of Task 4.

Subtask 5.a (3 points)

Draw the state transition graph for the model of Figure 4.

Subtask 5.b (5 points)

Suppose we only observe $\mathbf{y}_3$ in the figure (at time $t = 3$). What is the most likely hidden state sequence given y_3? Briefly justify your answer.

Subtask 5.c (2 points)

Would the most likely initial state in Subtask 5.b change if we were to decrease σ^2? Briefly justify your answer.