KTH Computer Science and Communication

Exam in DD2421 Machine Learning
2024-03-15, kl 14.00 – 18.00

Aids allowed: *calculator*, *language dictionary*.

To take this exam you must be registered to this specific exam as well as to the course.

In order to pass this exam, your score x first needs to be 20 or more (out of 42, full point). In addition, given your points y from the Programming Challenge (out of 18, full point), the requirements on the total points, $p = x + y$, are preliminarily set for different grades as:

$$
\begin{aligned}
54 < p \leq 60 &\quad \rightarrow \quad A \\
48 < p \leq 54 &\quad \rightarrow \quad B \\
42 < p \leq 48 &\quad \rightarrow \quad C \\
36 < p \leq 42 &\quad \rightarrow \quad D \\
29 < p \leq 36 &\quad \rightarrow \quad E \quad \text{(A pass is guaranteed with the required points for 'E'.)} \\
0 \leq p \leq 29 &\quad \rightarrow \quad F
\end{aligned}
$$

This exam consists of sections **A**, **B**, and **C**.
NB: use different papers (answer sheets) for different sections.

A Graded problems

Potential inquiries to be addressed to Atsuto Maki.

A-1 Terminology (5p)

For each term (a–d) in the left list, find the explanation from the right list which *best* describes how the term is used in machine learning.

a) Dropout

b) Curse of dimensionality

c) RANSAC

d) Occams's razor

e) k-fold cross validation

1) An approach to find useful dimension for classification

2) Random strategy for area compression

3) An approach to train artificial neural networks

4) Algorithm to learn with latent variables

5) A robust method to fit a model to data with outliers

6) A technique for margin maximization

7) A technique for assessing a model by exploiting available data for training and testing

8) A principle to choose the simplest explanation

9) Issues in data sparsity in space

10) Problems in increasing computation cost

Solution: a-3, b-9, c-5, d-8, e-7

A-2 Classification (4p)

Suppose that we take a data set, divide it into training and test sets, and then try out two different classification procedures. We use *two-thirds* of the data for training, and the remaining *one-third* for testing. First we use Bagging and get an error rate of 10% on the training data. We also get the average error rate (averaged over both test and training data samples) of 15%. Next we use k-nearest neighbor (where $k = 1$) and get an average error rate (averaged over both test and training data samples) of 10%.

a) What was the error rate with 1-nearest neighbor on the training set? (1p)

b) What was the error rate with 1-nearest neighbor on the test set? (1p)

c) What was the error rate with Baggind on the test set? (1p)

d) Based on these results, indicate the method which we should prefer to use for classification of new observations, with a simple reasoning. (1p)

Solution: **a**) 0%. Training error for 1-NN is always zero. **b**) 30%. Given the answer of **a**), the testing error is 30%. **c**) 25%. **d**) Bagging, because it achieves lower error rate on the test data (25% < 30%).

A-3 Regression with regularization (4p)

For a set of N training samples $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}$, each consisting of input vector $\mathbf{x}$ and output y, suppose we estimate the regression coefficients $\mathbf{w}^\top (\in \mathbf{R}^d) = \{w_1, \ldots, w_d\}$ in a linear regression model by minimizing

$$\sum_{n=1}^{N} (y_n - \mathbf{w}^\top \mathbf{x}_n)^2 + \lambda \sum_{i=1}^{d} w_i^2$$

for a particular value of λ.

Now, let us consider different models trained with different values of λ, starting from a very large value (infinity) and *decreasing* it down to 0. Then, for parts **a**) through **c**), indicate which of i. through v. is most likely to be the case.

Briefly justify each of your answers.

a) As we decrease λ, the variance of the model will:
i. Remain constant.
ii. Steadily decrease.
iii. Steadily increase.
iv. Follow a U shape.
v. Follow an inverted U shape. (1p)

b) Repeat **a**) for the training error (residual sum of squares, RSS). (1p)

c) Repeat **a**) for test RSS. (2p)

Solution: **a**)-iii, **b**)-ii, **c**)-iv
If λ was infitity, all w_i would be zero; the model is extremely simple, predicting a constant and has no variance with the prediction being far from actual value (thus with high bias). As we decrease λ, all w_i increase from zero toward their least square estimate values while steadily decreasing the *training* error to the ordinary least square RSS (and also decreasing bias) as the model continues to better fit training data. The values of w_i then become more dependent on training data, thus increasing the variance. The *test* error initially decreases as well, but eventually start increasing due to overfitting to the training data.

A-4 Ensemble Methods (5p)

Briefly answer the following questions regarding ensemble methods of classification.

a) What are the two kinds of randomness involved in the design of Random Forests? (2p)

b) In Adaboost algorithm, each training sample is given a weight and it is updated according to some factors through an iteration of training weak classifiers. What are the two most dominant factors in updating the weights? (2p)

c) In Adaboost algorithm, how are the two factors mentioned in **b)** used? (1p)

Solution:

a) Randomness (i) in generating bootstrap replicas and (ii) in possible features to use at each node.

b) The update is according to (i) if the sample was misclassified, and (ii) the reliability of the weak classifier based on the training error; the smaller the training error, the greater the reliability.

c) The weight is increased if misclassified, and decreased if classified correctly.
The reliability is then used as the coefficient.

A-5 PCA, Subspace Methods (4p)

We consider to solve a K-class classification problem with the Subspace Method and for that we compute a subspace $\mathcal{L}^{(j)}$ ($j = 1, ..., K$) using training data for each class, respectively. That is, given a set of feature vectors (as training data) which belong to a specific class C (i.e. with an identical class label), we perform PCA on them and generated an orthonormal basis $\{\mathbf{u}_1, ..., \mathbf{u}_p\}$ which spans a p-dimensional subsapce, $\mathcal{L}$, as the outcome. Provide an answer to the following questions.

a) Given that we compute eigenvectors of the auto-correlation matrix Q based on the training data and use some of them to form the basis $\{\mathbf{u}_1, ..., \mathbf{u}_p\}$ as, do we take eigenvectors corresponding to the p smallest, or p largest eigenvalues of Q? Simply mention your answer. (1p)

b) We have a new input vector $\mathbf{x}$ whose class is unknown, and consider its projectiton length on $\mathcal{L}$. Describe how the projection length is represented, using a simple formula. (2p)

c) Given $\mathbf{x}$, we computed its projectiton length on each subspace as $S^{(j)}$ ($j = 1, ..., K$). Among those $S^{(\alpha)}$, $S^{(\beta)}$, and $S^{(\gamma)}$ were the largest, the second largest, and the smallest, respectively. Based on this observation which class should $\mathbf{x}$ belong to? Simply choose one of the three. (1p)

Solution:

a) largest

b) $\sqrt{S}$ where $S = \Sigma_{i=1}^{p} (\mathbf{x}, \mathbf{u}_i)^2$

c) α

B Graded problems

Potential inquiries to be addressed to Olov Andersson.

B-1 Probability Theory: The Monty Hall Problem (2p)

The famous Monty Hall problem is sometimes called a paradox due to how counter intuitive it may first seem. This problem is usually presented as a game show where the contender is presented with three closed doors and asked to pick one. Only one has the prize behind it (a car), while the rest has goats behind them. After the contestant has chosen a door, and *before* opening it to reveal what is behind it, the show host Monty opens a different door and asks if the contestant wants to stay with his choice or switch to another door. You can assume that Monty will not open a door that has the prize behind it (as he knows where the prize is).

Model this problem using probability theory and Bayes' rule to show which choice is best (staying with your original door, or switching to the remaining door that Monty did not open). Show your calculations.

Solution: Define D as the (binary) random variable representing the probability that the door you originally chose has the prize behind it. We need to compute $P(D = \text{Prize} \,|\, M = \text{Goat})$, where $M = \text{Goat}$ means it is conditioned on the event of Monty opening another door with a goat.

Short version using Bayes' rule:

$$P(D = \text{Prize} \,|\, M = \text{Goat}) = \frac{P(M = \text{Goat} \,|\, D = \text{Prize})P(D = \text{Prize})}{P(M = \text{Goat})} \tag{1}$$

Plug in information derived from the text:

$$P(D = \text{Prize} \,|\, M = \text{Goat}) = \frac{1 \cdot \frac{1}{3}}{1} = \frac{1}{3}, \text{as Monty will always pick a door with a goat.} \tag{2}$$

The law of total probability means that the probability of the other door holding the prize is $1 - P(D = \text{Prize} \,|\, M = \text{Goat}) = \frac{2}{3}$, which is clearly the better choice. The information we gained from Monty's actions and switching doors let us improve our chances from the prior of $P(D = \text{Prize}) = \frac{1}{3}$.

B-2 Maximum Likelihood Estimation (6p)

Consider a probabilistic regression problem for the data $\mathcal{D} = ((x, y)_i : x \in \mathbb{R}^2, y \in \mathbb{R})$ by using the model $y = \mathbf{w}^T\mathbf{x} + \epsilon$, where $\epsilon \sim \mathcal{N}(0, \sigma^2)$.

a) Write the likelihood function for ML-estimation of the parameters $\mathbf{w}^T, \sigma^2$ from data. (2p)

Hint: Use that any normally distributed variable x (not the $\mathbf{x}$ from above) is typically defined as

$$\mathcal{N}(x|\mu, \sigma^2) \sim \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}. \tag{3}$$

b) Derive the ordinary least-squares linear regression problem from the ML estimation problem above. Show your calculations. (4p)

Solution:

a) Reformulate the problem such that y is normally distributed with mean $\mathbf{w}^T\mathbf{x}$ and use i.i.d. assumption from the definition of the model.

$$\begin{aligned} Pr(y|\mathbf{x}, \mathbf{w}, \sigma^2) &= \prod_i Pr(y_i|\mathbf{x}_i, \mathbf{w}, \sigma^2) \\ &= \prod_i \mathcal{N}(y_i|\mathbf{x}_i\mathbf{w}, \sigma^2) \\ &= \prod_i \left[\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y_i - \mathbf{w}^T\mathbf{x}_i)^2}{2\sigma^2}}\right] \end{aligned}$$

b) We typically take the log of the likelihood for computational reasons,

$$\begin{aligned} \log Pr(y|\mathbf{x}, \mathbf{w}, \sigma^2) &= \log \prod_i Pr(y_i|\mathbf{x}_i, \mathbf{w}, \sigma^2) \\ &= \sum_i \log Pr(y_i|\mathbf{x}_i, \mathbf{w}, \sigma^2) \\ &= \sum_i \log \left[\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y_i - \mathbf{w}^T\mathbf{x}_i)^2}{2\sigma^2}}\right] \\ &= \sum_i \left[-\frac{1}{2}\log(2\pi\sigma^2) - \frac{(y_i - \mathbf{w}^T\mathbf{x}_i)^2}{2\sigma^2}\right] \end{aligned}$$

Maximizing the likelihood is the same as minimizing the negative log-likelihood. ML-estimating the parameters w is then equivalent to

$$\begin{aligned} \arg\max_{\mathbf{w}} Pr(y|x, \mathbf{w}, \sigma^2) &= \arg\min_{\mathbf{w}} -\sum_i \left[-\frac{1}{2}\log(2\pi\sigma^2) - \frac{(y_i - \mathbf{w}^T\mathbf{x}_i)^2}{2\sigma^2}\right] \\ &= \arg\min_{\mathbf{w}} \sum_i \left[\frac{(y_i - \mathbf{w}^T\mathbf{x}_i)^2}{2\sigma^2}\right] \\ &= \arg\min_{\mathbf{w}} \sum_i \left[(y_i - \mathbf{w}^T\mathbf{x}_i)^2\right] \end{aligned}$$

As the first term does not include $\mathbf{w}$, it can be removed from the optimization problem. The σ^2 is also a constant that can be moved out and removed when optimizing for $\mathbf{w}$. Therefore $\mathbf{w}_{\text{MLE}} = \mathbf{w}_{\text{OLS}}$ (ordinary least-squares regression).

B-3 Maximum A Posteriori and Bayesian Methods (4p)

Consider a binary (1/0) classification problem where you have a labeled data set $\mathcal{D} = ((\mathbf{x}, y)_i)$. You have assumed the data follows some probabilistic model $Pr(y|\mathbf{x}, \theta)$ with parameter vector θ, resulting in a parameter likelihood function $\prod_i Pr(y_i|\mathbf{x}_i, \theta)$. You additionally assume some (weak) prior distribution $Pr(\theta)$ on the parameters.

a) Given a new input $\mathbf{x}'$, show how you would compute the probability of the new label y' being $y' = 1$. Assume you are estimating the model *parameters* from data using using maximum a posteriori (MAP) estimation. (2p)

b) Do the same, but this time assume you are using Bayesian methods for the model parameters θ. (2p)

Solution:

a) First MAP estimate parameters,

$$\begin{aligned} \theta_{\text{MAP}} &= \arg\max_{\theta} Pr(\theta|\mathcal{D}) \\ &\propto \arg\max_{\theta} \prod_i Pr(y_i|\mathbf{x}_i, \theta) Pr(\theta) \text{ (normalization term is constant wrt. } \theta) \end{aligned}$$

Then plug this into the probabilistic model $Pr(y' = 1|\mathbf{x}', \theta_{\text{MAP}})$.

b) Directly write the joint distribution over the involved random variables, using the parameter posterior from the data, and marginalize out the parameter vector. E.g.,

$$Pr(y' = 1|\mathbf{x}', \mathcal{D}) = \int_{\theta} Pr(y' = 1|\mathbf{x}', \theta) Pr(\theta|\mathcal{D}) \tag{4}$$

C Graded problems

Potential inquiries to be addressed to Jörg Conradt.

C-1 Warm-up: Support Vector Machine (1p)

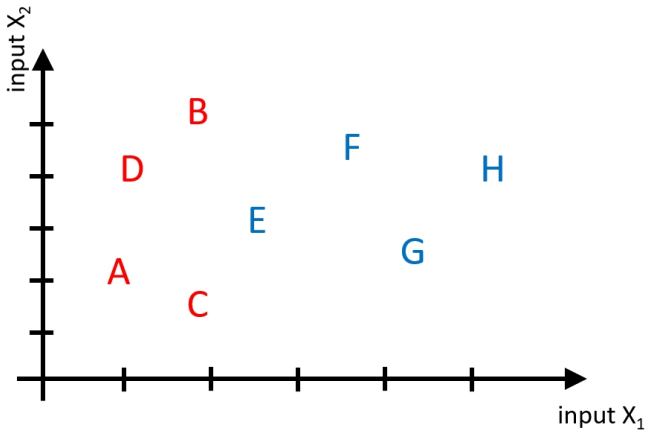
Select exactly **one** option of (1), (2), or (3), justify your answer **with keywords!**
Complete the following sentence: Out of all hyperplanes which solve a classification problem, the one with widest margin will probably ...

1. ... compute fastest.
2. ... have the smallest number of parameters.
3. ... generalize best.

Solution: 3 The wide margin maximizes the distance to known data points; thereby making future classifications most likely correct.

C-2 Support Vector Classification (3p)

The following diagram shows a small data set consisting of four RED samples (A, B, C, D) and four BLUE samples (E, F, G, H). This data set can be linearly separated.

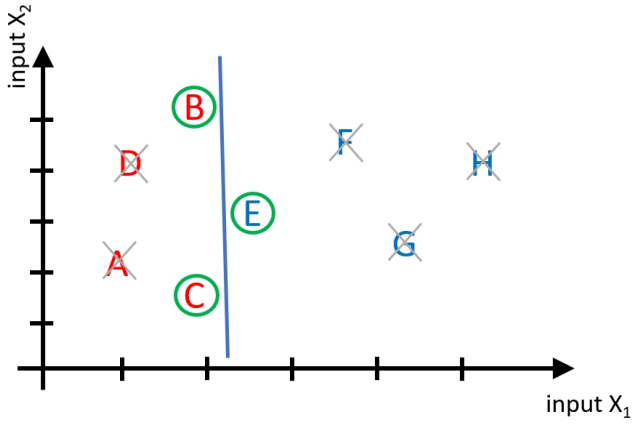


a) We use a linear support vector machine (SVM) without kernel function to correctly separate the RED (A-D) and the BLUE (E-H) class. Which of the data points (A-H) will the support vectors machine use to separate the two classes? Name the point(s) and explain your answer **IN KEYWORDS!** (2p)

b) Assume someone suggests using a non-linear kernel for the SVM classification of the above data set (A-H). Give one argument **in favor of** using non-linear SVM classification for such a data set. **USE KEYWORDS!** (1p)

Solution:

a) The blue line shows the linear decision boundary between the two classes.



The data points in green circles are support vectors for the linear decision boundary **B**, **C**, and **E**. Reason: if any of those data points change, the decision boundary changes.

b) at least one of the following (or similar) **in favor** (1p):

+) The decision boundary margin might get wider with a non-linear kernel.

+) The same learning approach is likely to work for additional (possibly more complex) data.

C-3 Warm-up: Artificial Neural Networks (1p)

Select exactly **one** option of (1), (2), or (3), justify your answer **with keywords!**
Error-Backpropagation-Training in neural networks mainly performs the following activity:

1. identifies the best neuron activation function f(x).
2. modifies all neurons' input weights and biases.
3. changes the connectivity between layers in the network.

Solution: 2 the neuron activation function (1) and the connectivity between layers (3) are predetermined. Error-backpropagation uses the gradient of the neuron's activation to update the significance of signals between neurons; hence changes the weights.

C-4 Artificial Neural Networks (3p)

Consider the training data in the table, where $+$ means a positive sample and $-$ a negative.

x_1	x_2	Class
8	8	$-$
8	-4	$-$
4	0	$+$
0	4	$+$
-6	-6	$-$
-6	8	$-$

a) What is the minimum number of *layers* needed for an artificial neural network to correctly classify all these points? Motivate your answer **IN KEYWORDS**. (2p)

b) How many input nodes and how many output nodes does your neuronal network need to address this problems? (1p)

Solution:

a) Two layers are needed and sufficient. The points are not linearly separable, so one layer is **not** sufficient. Two layers can solve any separation problem.

b) Two input nodes and one output node (given by the data table).