Machine Learning I: Foundations

Prof. Dr. Marius Kloft

August 3, 2020

Disclaimer

This exam is reproduced entirely from memory. To the best of my knowledge, it contains all questions from the exam, except where mentioned. I have tried to phrase the questions as closely to how they appeared on the exam as possible. I am not completely sure about distribution of marks (given in square brackets next to section name) for each question.

Instructions

1) The exam consists of 100 marks.

2) Total time of the exam is 150 minutes.

3) Write in non-erasable ink which is not green or red.

4) You will be given 15 minutes to read the assignments before the start of the exam. Any questions you may have will be collected by one of the proctors, and at the end of 15 minutes, we will answer all your questions together.

5) If you have a question during the exam, you will have to be led outside the exam hall where your question will be dealt with. You will lose exam time this way, so try to ask all your questions before the start of the exam.

6) No helping material including calculators, scripts, dictionaries are allowed. You may only keep your writing material, food and drinks with you.

7) Turn off and put all your electronic devices, including mobile phones and smart-watches in your bags, or submit them for safekeeping.

8) You cannot leave the exam hall during the last 30 minutes of the exam in order to not disturb other students.

1 TRUE / FALSE Questions [2+2+2+2+2=10]

For each of the questions below, exactly one of the three statements is correct. Choose the correct statement. For each correct answer, you get +2 score, and -2 score for a wrong answer. Leaving a question blank will give you 0 scores. Minimum scores obtained in this question cannot be below 0.

(a) Gradient descent can be used to

1. ... calculate the global minimum of any function.
2. ... ?
3. ... calculate the global minimum of a function with some assumptions on the function.

(b) Backpropagation is an algorithm for ...

1. ... optimization of parameters while training a neural network.
2. ... efficient calculation of gradients during gradient descent.
3. ... ?

(c) Linear learning machines can

1. ... be converted to non-linear machines by using upstream mapping.
2. ... be converted to non-linear machines by using kernel method.
3. ... not be converted to non-linear learning machines.

(d) ... ?

1. ... ?
2. ... ?
3. ... ?

(e) ... ?

1. ... ?
2. ... ?
3. ... ?

Level of Expectation: Give your answers as a) 1, b) 2, etc. Make sure to write the answers on the answer sheet or they will not be graded. No explanations are required.

2 Convolution [20?]

Output size in a CNN can be given by the following formula

$$o = \frac{i - k + 2p}{s} + 1$$

where i is the input size, k is the kernel size, s is the stride and p is the padding.

(a) Explain the variables i, o, k, p, s in context of a convolutional neural network.

(b) There are two other variables, c, the number of channels, and f, the number of filters, which are also important in a convolutional neural network, but they don't appear in this formula. Why is that and why are they important? What changes if either c or f is changed?

(c) Give an example of where this formula can be used, or explain why is it important.

(d) Sometimes, we may want to use two different values of kernel size k in each direction instead of just one, because we expect our data to be different in both directions. Why is this problematic and how can we fix this?

Level of Expectation:

3 Regression [2+6+2=10]

The following formula gives a variation of the ridge regression we studied in class.

$$\mathbf{w}_{crr} = \arg\min_{\mathbf{w} \in \mathbb{R}^d} \frac{1}{2}||\mathbf{w}||^2 + C||y - X^T\mathbf{w}||^2 + (\mathbf{s}^T\mathbf{w}) * 2$$

(a) State the difference in this formula from ridge regression, with its geometric interpretation.

(b) Derive the closed form of $\mathbf{w}_{crr}$.

(c) Prove that this formula is actually the same as ridge regression for $\mathbf{s} \in \mathbb{R}$. ***Hint:*** *The final answer of part (b) may be useful here.*

Level of Expectation: In (b), a complete derivation is required to obtain full marks.

4 Gradient Descent [10?]

1: $\boldsymbol{\theta}_{t+1} := \boldsymbol{\theta}_t + \lambda_t \nabla^2 f(\boldsymbol{\theta}_t)$
2: end for
3: initialize $\boldsymbol{\theta}_0$
4: $\boldsymbol{\theta}_{t+1} := \boldsymbol{\theta}_t - \lambda_t \nabla f(\boldsymbol{\theta}_t)$
5: $\boldsymbol{\theta}_{t+1} := \boldsymbol{\theta}_t + \lambda_t \nabla f(\boldsymbol{\theta}_t)$
6: for $t = 1 : T$ do

(a) Use the lines given above to write code of gradient descent algorithm. A line may be used more than once. Some lines are traps.

(b) The gradient descent we used in class was different from your code in (a). We used a concept of batches and data points in class. What was that concept?

(c) We need to standardize our dataset with a normalizing constant when using the concept of batches. What is this normalization constant?

Level of Expectation: On the answer sheet, for (a), just write the sequence of lines of code, e.g. 3, 5, 1, 2, 6. In (b), a formula is not required but you can write one if it helps you explain better.

5 K-Means Clustering [10?]

1: end for
2: initialize cluster centers $\mathbf{c}_1, \ldots, \mathbf{c}_k$
3: function KMEANS(k, $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^d$)
4: for $i = 1 : n$ do
5: repeat
6: for $i, j = 1 : n$ do
7: $y_i := \arg\min_{j=1,\ldots,k} ||\mathbf{x}_i - \mathbf{c}_j||^2$
8: return cluster centers $\mathbf{c}_1, \ldots, \mathbf{c}_k$
9: for $j = 1 : k$ do
10: until convergence criterion is met
11: $\mathbf{c}_j := mean(\{\mathbf{x}_i : y_i = j\})$

(a) Use the lines given above to write code of K-Means algorithm. A line may be used more than once. Some lines are traps.

(b) The code in part (a) does not support kernels. How can this be modified for kernelization?

Level of Expectation: On the answer sheet, for (a), just write the sequence of lines of code, e.g. 3, 5, 1, 2, 6. In part (b), a complete derivation of the formula of kernel k-means is expected.

6 Kernel Methods [6+10+4=20]

Given that A is a symmetric, positive definite matrix and $\mathbf{s} \in \mathbb{R}$.

(a) Prove that $\mathbf{x}_i^T A \mathbf{x}_j$ is a kernel.

(b) Prove $(s^T \mathbf{x}_i^T \mathbf{x}_j s)(\mathbf{x}_i^T A \mathbf{x}_j + \mathbf{x}_i^T \mathbf{x}_j + s^T s)$ is a kernel.

(c) $k(\mathbf{x}_i, \mathbf{x}_j) := \alpha k_1(\mathbf{x}_1, \mathbf{x}_1) + \beta k_2(\mathbf{x}_2, \mathbf{x}_2)$ is a kernel if k_1 and k_2 are kernels. Prove that this statement is **false** for all $\alpha, \beta \in \mathbb{R}$.

Level of Expectation: For part (b), you may refer to any of the theorems we studied in class, but you must first explicitly state those theorems before using them.

7 Support Vector Machines [4+10+4+2=20]

We are given the following data set D.

$$D := \{(\begin{bmatrix} 0 \\ -1 \end{bmatrix}, 1), (\begin{bmatrix} 0 \\ 1 \end{bmatrix}, 1), (\begin{bmatrix} 2 \\ 0 \end{bmatrix}, -1), (\begin{bmatrix} 1 \\ 0 \end{bmatrix}, 1)\}$$

(a) What is the objective function of soft-margin linear SVM?

(b) Optimize the SVM by running gradient descent on it with the following parameters

$$C = \frac{1}{2}, \lambda = 1, \mathbf{w}_0 = \begin{bmatrix} 0 & 0 \end{bmatrix}^T, b_0 = 0$$

(c) Find two points lying on the separating hyperplane you obtained after optimization in part (b).

(d) Find the label predicted by the optimized SVM for the following data point

$$\mathbf{x} = \begin{bmatrix} \frac{1}{2} \\ 0 \end{bmatrix}$$

Level of Expectation: In (a) only a formula is expected. For part (b), you have to show step-by-step working of gradient descent with enough detail to follow along. If you cannot solve (b), use $\mathbf{w}^* = \begin{bmatrix} \frac{3}{2} & 0 \end{bmatrix}^T, b^* = 2$ to solve parts (c) and (d).