+0/1/60+

Introduction to Machine Learning Exam
Questions Pack

August 21, 2021

Time limit: 120 minutes

Instructions. This pack contains all questions for the final exam. It contains the questions only. Please use the accompanying answer sheet to provide your answers by blackening out the corresponding squares. As the exam will be graded by a computer, please **make sure to do blacken out the whole square and do not use ticks or crosses**. *During* the exam you can use a **pencil** to fill out the squares as well as an **eraser** to edit your answers. *After* the exam is over, we will collect the questions pack and provide you with additional time to blacken out the squares on the answer sheet with a **black pen**. *Nothing* written on pages of the question pack will be collected or marked. **Only the separate answer sheet with the filled squares will be marked**.

Please make sure that your answer sheet is clean and all answers are clearly marked by filling the squares out completely. We reserve the right to classify answers as wrong without further consideration if the sheet is filled out ambiguously.

Collaboration on the exam is strictly forbidden. You are allowed a summary of *two* A4 pages and a simple, non-programmable calculator. The use of any other helping material or collaboration will lead to being excluded from the exam and subjected to disciplinary measures by the ETH Zurich disciplinary committee.

Question Types In this exam, you will encounter the following question types.

- **Multiple Choice questions with a single answer.**

  Multiple Choice questions have **exactly one** correct choice. Depending on the difficulty of the question **2, 3, or 4** points are awarded if answered correctly, and **zero points** are awarded if answered wrong or not attempted.

- **True Or False questions.**

  Each True Or False questions has a value of **1 point** if answered correctly, **0 points** if answered wrong or not attempted.

The total points sum up to 100 points. Not all questions need to be answered correctly to achieve the best grade. There are **no negative grades** so you are incentivized to attempt all questions.

1 Regression, Classification and Other Losses

We are given a dataset consisting of n labeled training points $\mathcal{D} = \{(x_1, y_1), ..., (x_n, y_n)\}$, where $x_i \in \mathbb{R}^d$ are the feature vectors and $y_i \in \mathbb{R}$ are the labels. The *design matrix* $X \in \mathbb{R}^{n \times d}$ contains as rows the feature vectors $x_i \in \mathbb{R}^d$. The label vector is denoted by $y = (y_1, \ldots, y_n)^\top \in \mathbb{R}^n$. We assume that X is full rank i.e., $rank(X) = \min(n, d)$. The *empirical risk* with the squared loss is defined as follows:

$$\hat{R}_{\mathcal{D}}(w) = \frac{1}{n} \sum_{i=1}^{n} (w^\top x_i - y_i)^2 = \frac{1}{n} \|y - Xw\|_2^2. \tag{1}$$

The goal is to to find $w \in \mathbb{R}^d$ that minimizes the empirical risk.

1.1 Ordinary Least Squares

Question 1 *(1 point)* $\hat{R}_{\mathcal{D}}(w)$ is a convex function in w.

A True B False

Question 2 *(1 point)* When $n \leq d$ there always exists w such that $\|y - Xw\|_2 = 0$.

A True B False

Question 3 *(3 points)* We would like to minimize the empirical risk using stochastic gradient descent (with replacement). At time step t, what is the update formula?

A $w_{t+1} = w_t + \eta_t(X^\top X w_t - 2X^\top y)$.
B $w_{t+1} = w_t - \eta_t(2X^\top X w_t - 2X^\top y)$.
C $w_{t+1} = w_t + \eta_t(2X w_t - 2XX^\top y)$.
D $w_{t+1} = w_t - \eta_t(X w_t - 2XX^\top y)$.
E $w_{t+1} = w_t + \eta_t(2y_i - 2w_t^\top x_i)x_i$, for some randomly chosen $i \in \{1, 2, \ldots, n\}$.
F $w_{t+1} = w_t - \eta_t(2y_i - 2w_t^\top x_i)x_i$, for some randomly chosen $i \in \{1, 2, \ldots, n\}$.
G $w_{t+1} = w_t + \eta_t(y_i - 2w_t^\top x_i)x_i$, for some randomly chosen $i \in \{1, 2, \ldots, n\}$.
H $w_{t+1} = w_t - \eta_t(2y_i - w_t^\top x_i)x_i$, for some randomly chosen $i \in \{1, 2, \ldots, n\}$.

Are the following statements True or False?

Question 4 *(1 point)* At each iteration t of the **gradient descent** algorithm, there exists a learning rate $\eta_t > 0$ such that the objective decreases (either strictly decreases or stays the same).

A True B False

Question 5 *(1 point)* At each iteration t of the **stochastic gradient descent** algorithm, there exists a learning rate $\eta_t > 0$ such that the objective decreases (either strictly decreases or stays the same).

A True B False

1.2 Model Evaluation

Assume that you have access to a dataset $\mathcal{D} = \{(x_1, y_1), ..., (x_n, y_n)\}$ of $n = 10000$ data samples (x_i, y_i) that are drawn i.i.d. (indpendently and identically distributed) from some (unknown) distribution $p(x, y)$. You now need to decide how to split this dataset into a training set $\mathcal{D}_{\text{train}}$ and a validation set $\mathcal{D}_{\text{val}}$ so that you can run the following standard procedure to learn and evaluate a regression model:

Step 1: Training the regression model on $\mathcal{D}_{\text{train}}$ by minimizing the empirical risk

$$\widehat{f}_{\mathcal{D}_{\text{train}}} = \arg\min_{f} \left(\widehat{R}_{\mathcal{D}_{\text{train}}}(f) \triangleq \frac{1}{|\mathcal{D}_{\text{train}}|} \sum_{(x_i, y_i) \in \mathcal{D}_{\text{train}}} (y_i - f(x_i))^2 \right). \tag{2}$$

Step 2: Estimating the true (population) risk of the learned model $R(\widehat{f}_{\mathcal{D}_{\text{train}}})$ by computing the empirical risk on $\mathcal{D}_{\text{val}}$ defined as

$$\widehat{R}_{\mathcal{D}_{\text{val}}}\left(\widehat{f}_{\mathcal{D}_{\text{train}}}\right) = \frac{1}{|\mathcal{D}_{\text{val}}|} \sum_{(x_i, y_i) \in \mathcal{D}_{\text{val}}} (y_i - \widehat{f}_{\mathcal{D}_{\text{train}}}(x_i))^2. \tag{3}$$

Remember that for a fixed estimator $f(x)$, the true (population) risk is defined as:

$$R(f) = \mathbb{E}_{(x,y)\sim p}[(y - f(x))^2].$$

Are the following statements True or False?

Question 6 *(1 point)* $\widehat{R}_{\mathcal{D}_{\text{val}}}\left(\widehat{f}_{\mathcal{D}_{\text{train}}}\right)$ is more likely to provide better a estimate of the true (population) risk $R\left(\widehat{f}_{\mathcal{D}_{\text{train}}}\right)$, when using a validation set of size 500 as opposed to a validation set of size 1000.

A True B False

Question 7 *(1 point)* Choosing a training set of size 1000 is more likely to provide a model $\widehat{f}_{\mathcal{D}_{\text{train}}}$ that has a lower true (population) risk $R\left(\widehat{f}_{\mathcal{D}_{\text{train}}}\right)$ compared to training set of size 2000.

A True B False

Question 8 *(1 point)* The training risk (error) is always less than or equal to the validation risk (error), i.e.,

$$\widehat{R}_{\mathcal{D}_{\text{train}}}(\widehat{f}_{\mathcal{D}_{\text{train}}}) \leq \widehat{R}_{\mathcal{D}_{\text{val}}}\left(\widehat{f}_{\mathcal{D}_{\text{train}}}\right).$$

A True B False

Question 9 *(1 point)* The validation risk in Equation (3) is an unbiased estimator of the true (population) risk i.e.,

$$\mathbb{E}_{\mathcal{D}}\left[\widehat{R}_{\mathcal{D}_{\text{val}}}\left(\widehat{f}_{\mathcal{D}_{\text{train}}}\right)\right] = \mathbb{E}_{\mathcal{D}}\left[R(\widehat{f}_{\mathcal{D}_{\text{train}}})\right].$$

A True B False

1.3 Classification

Question 10 *(3 points)* The table below shows the confusion matrix for a classifier f on the Iris dataset, a dataset consisting of three different types of flowers as labels.

Predicted Class \ Actual Class	Setosa	Versicolour	Virginica
Setosa	20	10	20
Versicolour	0	50	0
Virginica	5	5	40

From the table, we derive a new binary classifier f_{binary} that predicts "Virginica" (label $y = +1$) when f does and "not Virginica" (label $y = -1$) when f predicts "Versicolour" or "Setosa". What is the false discovery rate (FDR) of f_{binary} ?
Reminder: FDR $= \frac{\#\text{FP}}{\#\text{FP}+\#\text{TP}}$ where #FP is the number of false positives and #TP is the number of true positives.

A $\frac{1}{3}$ B $\frac{1}{5}$ C $\frac{1}{4}$ D $\frac{1}{2}$ E $\frac{1}{9}$ F $\frac{3}{5}$ G 0 H $\frac{2}{5}$

Question 11 *(3 points)* A classifier $f : \mathcal{X} \to \{-1, +1\}$ is called a *random classifier*, if it assigns any $x \in \mathcal{X}$ independently to class $f(x) = \widehat{y} = +1$ with probability ρ and to class $f(x) = \widehat{y} = -1$ with probability $1 - \rho$ for some $0 \leq \rho \leq 1$. Taking the perspective of hypothesis testing, we call samples with predicted class $\widehat{y} = +1$ positives and class $\widehat{y} = -1$ negatives. Assume that the distribution of the data satisfies $P(x, y)$. We can compute the false positive rate FPR $= P(\widehat{y} = 1|y = -1)$ and false negative rate FNR $= P(\widehat{y} = -1|y = +1)$ of random classifiers for different ρ. What is the smallest FNR that you can obtain with a random classifier with FPR ≤ 0.25 by tuning ρ?

A 0 B $\frac{1}{8}$ C $\frac{1}{4}$ D $\frac{3}{8}$ E $\frac{1}{2}$ F $\frac{5}{8}$ G $\frac{3}{4}$ H 1

Question 12 *(4 points)* Assume that we are training a binary classifier $y = f(x)$ that is allowed to *abstain*, i.e., refrain from making a prediction. Therefore, we include an abstention label r as part of its action (label) space, that is $f : \mathcal{X} \to \{+1, -1, r\}$. In order to ensure that the classifier does not always abstain, we introduce a cost $c > 0$ for every abstention that the classifier makes. Given a labeled data sample (x, y), the 0-1 loss with abstention is then given by

$$\ell(f(x), y) = \mathbb{1}_{f(x)\neq y}\mathbb{1}_{f(x)\neq r} + c\mathbb{1}_{f(x)=r}.$$

For a given data distribution $P(x, y)$ and a fixed classifier f, the conditional expectation of the 0-1 loss given x, i.e., $\mathbb{E}[\ell(f(x), y) \mid x]$, can then be written as:

A $P(y = +1 \mid x)\left(\mathbb{1}_{f(x)=-1} + \mathbb{1}_{f(x)=+1}\right) + c\mathbb{1}_{f(x)=r}$.

B $P(y = +1 \mid x)\mathbb{1}_{f(x)=+1} + (1 - P(y = +1 \mid x))\mathbb{1}_{f(x)=-1} + c\mathbb{1}_{f(x)=r}$.

C $P(y = +1 \mid x)\mathbb{1}_{f(x)=-1} + (1 - P(y = +1 \mid x))\mathbb{1}_{f(x)=+1} + c\mathbb{1}_{f(x)=r}$.

D $P(y = +1 \mid x)\mathbb{1}_{f(x)=-1} + (1 - P(y = +1 \mid x))\mathbb{1}_{f(x)=+1} + (1 - c)\mathbb{1}_{f(x)=r}$.

E $P(y = +1 \mid x)\mathbb{1}_{f(x)=+1} + (1 - P(y = +1 \mid x))\mathbb{1}_{f(x)=-1} + (1 - c)\mathbb{1}_{f(x)=r}$.

F $P(y = +1 \mid x)\left(\mathbb{1}_{f(x)=-1} + \mathbb{1}_{f(x)=+1}\right) + (1 - c)\mathbb{1}_{f(x)=r}$.

1.4 Huber Loss

Consider the so-called Huber loss $\ell_{H,\delta} : \mathbb{R} \to \mathbb{R}$ for a fixed constant $\delta > 0$, defined as follows:

$$\ell_{H,\delta}(a) := \begin{cases} \frac{1}{2}a^2 & \text{if } |a| \leq \delta \\ \delta(|a| - \frac{1}{2}\delta) & \text{if } |a| > \delta \end{cases},$$

and represented in the following Figure 1.

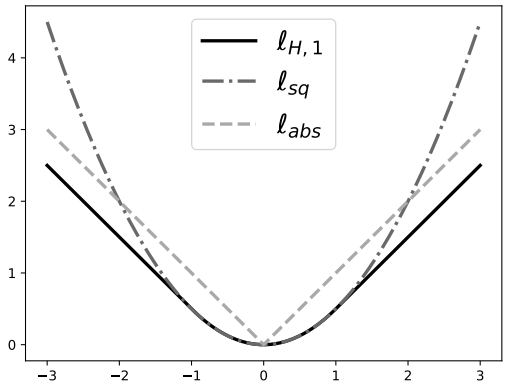


Figure 1: Plots of the ℓ_{abs}, ℓ_{sq} and Huber $\ell_{H,\delta}$ (with $\delta = 1$) loss functions.

This loss is often employed to obtain estimators that are more robust to outliers in the training set. Furthermore, we denote as $\ell_{abs}(a) := |a|$ and $\ell_{sq}(a) := \frac{1}{2}a^2$ the absolute value and squared losses, respectively.

Are the following statements True or False?

Question 13 *(1 point)* For large values of $|a|$, the Huber loss roughly behaves like the ℓ_{abs} loss, in the sense that $\lim_{|a|\to+\infty} \frac{\ell_{H,\delta(a)}}{\ell_{abs}(a)}$ evaluates to a finite constant.

A True B False

Question 14 *(1 point)* Consider running regression on a dataset $\{(x_i, y_i)\}_{i=1}^n$ with the Huber loss for a linear model to obtain

$$\hat{w} := \arg\min_{w\in\mathbb{R}^d} \sum_{i=1}^{n} \ell_{H,\delta}\left(y_i - w^\top x_i\right).$$

We expect the solution vector $\hat{w}$ to be sparse, thus we can use the Huber loss for feature selection.

A True B False

Question 15 *(1 point)* For any fixed value of $\delta > 0$, the Huber loss is convex on $\mathbb{R}$.

A True B False

2 Kernels

Question 16 *(2 points)* This question is regarding the task of regression. We want to compare using a fully connected neural network vs. using kernel regression. Which of the following statements is correct?

A For any choice of kernel, kernel regression methods implicitly operate on *finite-dimensional* feature spaces defined by the corresponding feature map.

B When using a fully connected neural network, we can arrive at a closed-form solution.

C Kernel regression can be considered as a linear model on an implicit feature space characterized by the kernel's feature map.

D Regardless of the kernel we use, kernel regression can only learn a polynomial function.

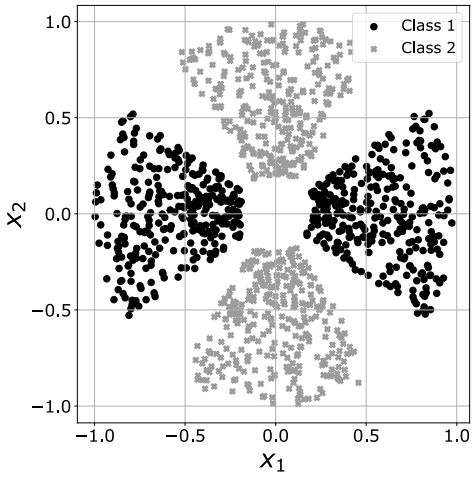


Figure 2: Dataset with two classes.

Question 17 *(4 points)* Consider the dataset in Figure 2. Which of the following feature maps $\Phi : \mathbb{R}^2 \to \mathbb{R}$ makes the classes linearly separable?
Reminder: The classes are linearly separable if there exists a threshold $\theta_0 \in \mathbb{R}$ such $\Phi(x) \geq \theta_0$ for points x from the first class, and $\Phi(x) < \theta_0$ for points x from the second class.

A $\Phi(x) = (x_1 + x_2)^2$

B $\Phi(x) = (x_1 x_2 + 1)^2$

C $\Phi(x) = \left| \frac{x_1 x_2}{\sqrt{x_1^2 + x_2^2}} \right|$

D $\Phi(x) = \left| \frac{x_1}{\sqrt{x_1^2 + x_2^2}} \right|$

E $\Phi(x) = \sqrt{x_1^2 + x_2^2}$

F $\Phi(x) = |x_1 x_2|$

G $\Phi(x) = x_1 - x_2$

H $\Phi(x) = \frac{x_1}{\sqrt{x_1^2 + x_2^2}}$

Question 18 *(3 points)* Let $d \in \mathbb{N}$ be a fixed constant number. Let $h(m) : \mathbb{N} \to \mathbb{N}$ denote the number of possible monomials (terms) of degree less or equal to m over d different variables $x = (x_1, \ldots, x_d)$. As an example for $d = 2, m = 2$, the number of all the possible monomials is 6: $1, x_1, x_2, x_1^2, x_2^2, x_1x_2$. What is the growth rate of $h(m)$?

A $h(m) \in \Theta(m)$. B $h(m) \in \Theta(m^2)$. C $h(m) \in \Theta(d^m)$. D $h(m) \in \Theta(m^d)$.

Question 19 *(3 points)* Let $d \in \mathbb{N}$ be a fixed constant number. Assume that multiplication of two numbers, addition of two numbers, and exponentiation of a number by another number (x^y) all have a constant ($\Theta(1)$) computational cost. What is the computational cost of constructing the kernel matrix for a dataset $\mathcal{D} = \{x_i\}_{i=1}^n$ with n points $x_i \in \mathbb{R}^d$ using the polynomial kernel for degree-m polynomials (we use the kernel $k(x, x') = (1 + x^\top x')^m$)?

A $\Theta(m)$ B $\Theta(nd^m)$ C $\Theta(nm)$ D $\Theta(m^d)$ E $\Theta(n^2)$ F $\Theta(mn^2)$

Are the following statements True or False?

Question 20 *(1 point)* For every valid kernel $k(x, x') : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ there exists a finite dimensional feature map $\Phi : \mathcal{X} \to \mathbb{R}^d$, such that $k(x, x') = \phi(x)^\top \phi(x')$.

A True B False

Question 21 *(1 point)* For $x, x' \in \mathbb{R} \setminus \{0\}$ we define $k(x, x') = \frac{\sin(x)}{(x')^2} + 1$. k is a valid kernel.

A True B False

Question 22 *(1 point)* For $x, x' \in \mathbb{R}^2$ we define $k(x, x') = x^\top M x'$ with $M = \begin{bmatrix} 1 & 2 \\ 2 & 1 \end{bmatrix}$. k is a valid kernel.

A True B False

Question 23 *(1 point)* Let k_1 be a valid kernel. Then, for any polynomial function f, $k(x, x') = f(k_1(x, x'))$ is a valid kernel.

A True B False

3 Dimensionality Reduction with PCA

In principal component analysis (PCA), we map the data points $x_i \in \mathbb{R}^d, i = 1, \ldots, n$, to $z_i \in \mathbb{R}^k$, $k \ll d$, by solving the following optimization problem:

$$C_* = \frac{1}{n} \min_{\substack{W \in \mathbb{R}^{d \times k}, W^\top W = I \\ z_1, \ldots, z_n \in \mathbb{R}^k}} \sum_{i=1}^{n} \|W z_i - x_i\|_2^2 . \tag{4}$$

We denote by $W_*, z_1^*, \ldots, z_n^*$ the optimal solution of Equation (4). Assume the data points are centered, i.e., $\sum_{i=1}^{n} x_i = 0$.

Question 24 *(3 points)* What is the value of $\mathrm{Tr}(W_* W_*^\top)$?
Reminder: For a square matrix $A \in \mathbb{R}^{k \times k}$ we denote its trace by $\mathrm{Tr}(A)$ and it is defined as the sum of its diagonal elements: $\mathrm{Tr}(A) = \sum_{i=1}^{k} A_{ii}$

A n B k C d D $\max(n, d)$

Question 25 *(3 points)* What holds for z_i^*?

A $z_i^* = W_*^\top (W_* W_*^\top)^{-1} x_i$
B $z_i^* = (W_* W_*^\top)^{-1} W_*^\top x_i$
C $z_i^* = (W_*^\top W_*)^{-1} W_*^\top x_i$
D $z_i^* = W_*^\top (W_* W_*^\top)^{-1} W_* x_i$

Question 26 *(4 points)* Let $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d \geq 0$ be the eigenvalues of the empirical covariance matrix $\Sigma = \frac{1}{n} \sum_{i=1}^{n} x_i x_i^\top \in \mathbb{R}^{d \times d}$. Let $v_1, \ldots, v_d \in \mathbb{R}^d$ be the corresponding eigenvectors. Remember that:

$$W z_i^* = (\sum_{j=1}^{k} v_j v_j^\top) x_i.$$

What is the value of C_*?
Hint: *(i.)* $\|x\|_2^2 = x^\top x = \mathrm{Tr}(x^\top x), \forall x \in \mathbb{R}^d$, *(ii.)* $\mathrm{Tr}(ABC) = \mathrm{Tr}(CAB) = \mathrm{Tr}(BCA)$ for matrices A, B, C of appropriate dimensions. For a definition of trace, see the reminder in Question 24.

A $\frac{1}{n} \sum_{i=k+1}^{d} \lambda_i$ B $\frac{1}{n} \sum_{i=1}^{k} \lambda_i$ C $\frac{1}{n} \sum_{i=k+1}^{d} \lambda_i^2$ D $\frac{1}{n} \sum_{i=1}^{k} \lambda_i^2$

Are the following statements True or False?

Question 27 *(1 point)* PCA helps us find a *linear* mapping to a lower dimensional space.

A True B False

Question 28 *(1 point)* Let n be the number of the points and d the dimension of the points in the dataset. In standard PCA, we compute the spectral decomposition (eigenvalues and eigenvectors) of the empirical covariance matrix with size $d \times d$. In kernelized PCA we instead compute the spectral decomposition of a matrix of size $n \times n$.

A True B False

Question 29 *(1 point)* Imagine two features are identical in the whole dataset, i.e., they are identical among all data samples $x_1, \ldots, , x_n$. Then, utilizing PCA, we can strictly reduce the dimension of the dataset by at least one with zero reconstruction error.

A True B False

4 Neural Networks and Optimization

4.1 Regression with Neural Networks

We model a regression problem on the dataset $\{(x_i, y_i)\}_{i=1,\dots,n}$, where inputs x_i and labels y_i are both in $\mathbb{R}^d$, with a fully connected neural network. We use the sigmoid activation function $\varphi(x) \triangleq \frac{1}{1+e^{-x}}$. The sigmoid function is applied element-wise to vectors.

Therefore, the whole neural network is a function $f : \mathbb{R}^d \to \mathbb{R}^d$, $f(x) = W_3\varphi(W_2\varphi(W_1x + b_1) + b_2) + b_3$, where $W_j \in \mathbb{R}^{d\times d}, b_j \in \mathbb{R}^d, j \in \{1, 2, 3\}$. Here, we have dropped the dependence of $f(x)$ on W_j, b_j for simplicity, and f is written only as a function of the input x to the neural network.
Suppose that during training, we only optimize parameters W_3, b_3, while keeping the other parameters fixed. That is, we would like to solve

$$\min_{W_3,b_3} \sum_{i=1}^{n} \|f(x_i) - y_i\|_2^2 . \tag{5}$$

Answer the following questions.

Question 30 *(1 point)* The objective from Equation (5) is convex with respect to W_3, b_3.

A True B False

Question 31 *(1 point)* We are using the cross-entropy loss function in the objective in Equation (5).

A True B False

Question 32 *(3 points)* Assume that $d = 1$. We pass a batch of input samples $\{x_i\}_{i\in S}$ defined by an index set $S \subseteq \{1, \dots, n\}$ through a batch normalization layer with scale and shift parameters $\gamma = 2, \beta = 0$ respectively. For every input x_i, the layer outputs $\bar{x}_i = \gamma\frac{x_i-\mu_S}{\sigma_S} + \beta$, where μ_S and σ_S are the empirical mean and standard deviation of the input batch. What are the empirical mean and standard deviation of $\{\bar{x}_i\}_{i\in S}$?

A (μ_S, σ_S)
B $(\mu_S, 2\sigma_S)$
C $\left(\frac{\mu_S}{|S|}, \frac{\sigma_S}{|S|}\right)$
D $\left(\frac{\mu_S}{|S|}, 2\frac{\sigma_S}{|S|}\right)$
E (0, 1)
F (4, 1)
G (0, 2)
H (4, 2)

4.2 Training Neural Networks

Consider training a multilayer perceptron (a fully connected neural network) using gradient descent. Answer the following questions.

Question 33 *(1 point)* The network weights are updated during forward propagation.

A True B False

Question 34 *(1 point)* For performing a binary classification task, the final output of the neural network is typically passed through a ReLu activation before being compared to the label.

A True B False

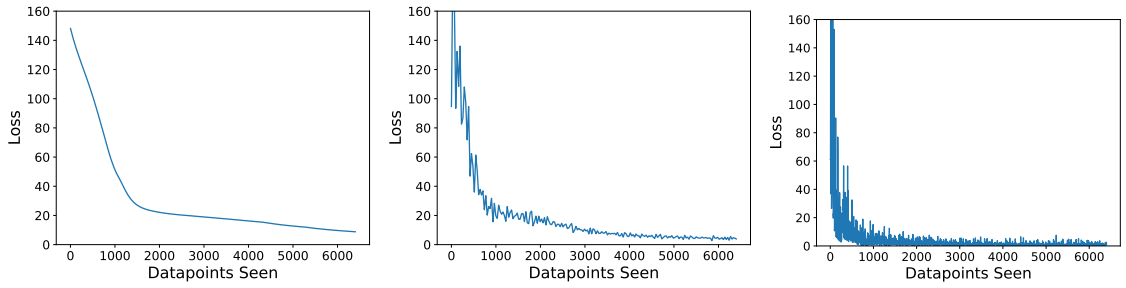
Question 35 *(2 points)* Below are the (smoothed) training loss curves for 3 small identical networks trained with the following optimization algorithms:

A Gradient descent.

B Batch stochastic gradient descent with large batch size.

C Batch stochastic gradient descent with small batch size.

We use the same constant learning rate in all 3 cases. The x-axis corresponds to the batch size multiplied by number of (stochastic) gradient descent iterations. Match the images (left to right) with the optimization method used.



A (A, B, C) B (A, C, B) C (B, A, C) D (B, C, A) E (C, A, B) F (C, B, A)

Question 36 *(3 points)*
We have a one-layer fully connected neural network with input nodes v_i, $i = 1, \ldots, d$, and a single output node v_{out}. The activation function of the output node v_{out} is the identity. We initialize every weight independently with a standard Gaussian distribution $w_i \sim \mathcal{N}\left(0, \sigma^2\right), i \in \{1, \ldots, d\}$. To avoid overfitting, we use dropout and thus independently set each node v_j to zero with probability $1 - p$.
Assume we give as input to the network d independent random variables $X_i, i = 1, \ldots, d$, with $\mathbb{E}[X_i] = 0, \mathbb{E}[X_i^2] = 1$. How should we choose the variance σ^2 so that we have $\mathbb{E}[v_{\text{out}}] = 0$ and $\mathbb{E}[v_{\text{out}}^2] = 1$? Here the randomness is over the random variables X_i, weights w_i, and node dropout events.

A $\sigma^2 = \frac{2}{dp(1-p)}$ B $\sigma^2 = \frac{2}{dp}$ C $\sigma^2 = \frac{1}{dp}$ D $\sigma^2 = \frac{1}{dp(1-p)}$

Question 37 *(2 points)* Figure 3 displays a training dataset and the learned function of 3 different neural networks that are trained on the dataset. The dataset consists of 100 scalar input-output pairs. All neural networks have one hidden layer with 20 units but differ in the choice of the activation function that are either the sigmoid, ReLu or identity function. Match the learned output function with the activation function that is used in the corresponding neural network.

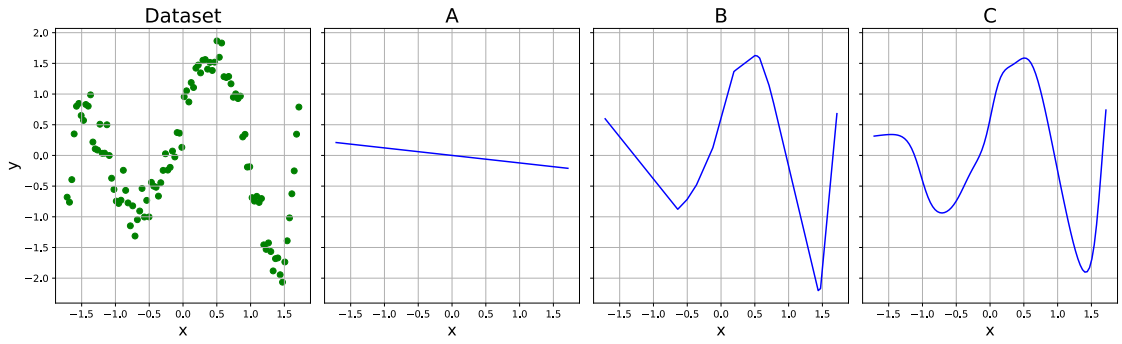


Figure 3: Effect of different activation functions.

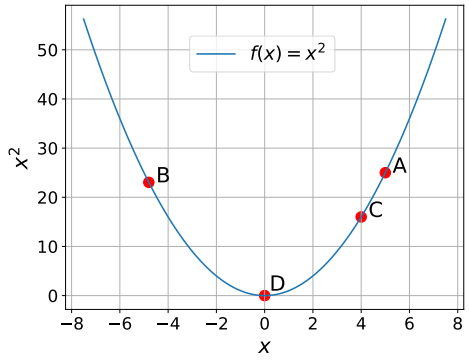
A (A, B, C) = (sigmoid, ReLu, identity)
B (A, B, C) = (sigmoid, identity, ReLu)
C (A, B, C) = (ReLu, sigmoid, identity)
D (A, B, C) = (ReLu, identity, sigmoid)
E (A, B, C) = (identity, ReLu, sigmoid)
F (A, B, C) = (identity, sigmoid, ReLu)

Question 38 *(3 points)* Assume the input to a convolutional layer is a 16×19 matrix and you perform a convolution with kernel size 4×3 with padding equal to 0 and horizontal and vertical strides equal to 2. After performing the convolution, you flatten the resulting matrix to a vector. What is the dimension of the resulting vector?

A 56
B 58
C 63
D 64

4.3 Optimization

Question 39 *(2 points)* You are using gradient descent to find the minimum of the function $f(x) = x^2$. You experiment by fixing the number of iterations T and trying different learning rates to find the one which leads to a better (smaller) solution. There are different possible settings, e.g., you choose a learning rate of zero (1), the optimal learning rate (2), a learning rate that is too large (3) or too small (4).
When initializing at point A on the plot below, which point of the plot are you most likely to reach with each learning rate setting? Match the settings 1, 2, 3, and 4 to the points on the plot indicated by letters A, B, C, and D.



1: learning rate of zero

2: optimal learning rate

3: too large learning rate

4: too small learning rate

A 1: A, 2: B, 3: C, 4: D
B 1: A, 2: B, 3: D, 4: C
C 1: A, 2: C, 3: B, 4: D
D 1: A, 2: D, 3: B, 4: C
E 1: D, 2: A, 3: C , 4: B
F 1: D, 2: A, 3: B, 4: C
G 1: C, 2: D, 3: B, 4: A
H 1: C, 2: B, 3: D, 4: A

Question 40 *(2 points)* Now we use gradient descent with momentum to find the minimum of the function $f(x) = x^2$. We set both the momentum parameter m and learning rate parameter η to some strictly positive (nonzero) value. At the 2020'th time step we move from $x_{2019} \neq 0$ to $x_{2020} = 0$. We perform one more update to obtain x_{2021}. Which of the following is true?

A $f(x_{2020}) < f(x_{2021})$ B $f(x_{2020}) > f(x_{2021})$ C $f(x_{2020}) = f(x_{2021})$

5 Frequentist and Bayesian Inference

Question 41 *(2 points)* What is the advantage of using bootstrap parameter estimates in comparison with distribution-dependent parameter estimates? Choose the correct statement among the following.

A It is possible to compute the closed-form solution for bootstrap parameter estimates.

B Bootstrap parameter estimates require less computational resources than distribution-dependent parameter estimates.

C Bootstrap parameter estimates can be computed for any black-box predictor.

D Distribution-based parameter estimates have asymptotic guarantees while bootstrap estimates do not.

Question 42 *(2 points)* Consider a dataset $D = \{x_i\}_{i=1}^n$ and assume that the likelihood function $P(D \mid \theta)$ depends on some parameters θ to be estimated. Furthermore, we assume that we have access to the true prior $P(\theta)$ over the parameters. Choose the correct statement among the following.

A The maximum a posteriori (MAP) estimate and the maximum likelihood estimate (MLE) coincide, since they both involve a maximization of the likelihood $P(D \mid \theta)$.

B The posterior distribution over the parameters is given by $P(\theta \mid D) = P(D \mid \theta)P(\theta)$.

C Both the MLE and the MAP estimates maximize $P(D \mid \theta)$.

D The maximum likelihood estimate is a point estimate, whereas the maximum a posteriori estimate outputs a distribution.

E Frequentist inference results in a point estimate for θ, whereas Bayesian inference naturally results in a distribution over θ.

6 Expectation Maximization Algorithm

6.1 Clustering Movies Using Soft EM

We utilize the (soft) expectation maximization (EM) algorithm for clustering movies into two clusters based on the actors who star in them. We abbreviate each of the movies '(S)tar Wars', '(T)itanic', 'The (G)odfather', '(I)nterstellar', and 'The (M)atrix' with the first letter of their names. For simplicity, we focus on only four important actors and we represent each movie as a binary (zero-one) feature vector $X \in \{0,1\}^4$, where the i'th, $i \in \{1,2,3,4\}$, element is equal to one if the actor is in the movie and zero otherwise. Assume there are two clusters $C \in \{0,1\}$.

Feature vectors X for each movie are independently generated in the following way:

- Sample a cluster from the distribution $P(C)$. The distribution $P(C)$ is Bernoulli with unknown parameter q, i.e.,

$$P(C=c) = \begin{cases} q & \text{if } c=1 \\ 1-q & \text{if } c=0 \end{cases}$$

- X_i is generated by $p(X_i \mid C)$ and X_i is conditionally independent of X_j given C for all $i \neq j$. This means that the ith feature is conditionally independent of the jth feature given the cluster assignment of the movie. The distribution $p(X_i \mid C), i \in \{1,2,3,4\}$, is also Bernoulli with unknown parameters. Note that this gives rise to 8 unknown parameters, four for each cluster.

Hint: All questions below can be solved independently of each other.

Question 43 *(3 points)* Which of the following describes the likelihood $p(X)$ of a single data point?

A $q\sum_{i=1}^4 p(X_i \mid C=1) + (1-q)\sum_{i=1}^4 p(X_i \mid C=0)$
B $(1-q)\sum_{i=1}^4 p(X_i \mid C=1) + q\sum_{i=1}^4 p(X_i \mid C=0)$
C $q\prod_{i=1}^4 p(X_i \mid C=1) + (1-q)\prod_{i=1}^4 p(X_i \mid C=0)$
D $(1-q)\prod_{i=1}^4 p(X_i \mid C=1) + q\prod_{i=1}^4 p(X_i \mid C=0)$

The 5 movies above have the following feature vectors, respectively, where the bracket notation is used to represent a vector:
$(1,1,1,0)$: G $(0,1,0,0)$: S $(1,0,1,0)$: T $(0,1,1,0)$: I $(1,0,1,0)$: M

Question 44 *(3 points)* **E-Step**
You initialize $\hat{p}(X \mid C=1) = \left(\frac{1}{8}, \frac{1}{4}, \frac{3}{4}, \frac{1}{2}\right)$ and $\hat{p}(X \mid C=0) = \left(\frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}\right)$ and $\hat{q} = \frac{1}{2}$. You use the soft expectation maximization (EM) algorithm to cluster the movies. After performing one E-step with this initialization, what is the assignment probability of G to cluster 1?

A 0 B $\frac{3}{19}$ C $\frac{3}{16}$ D $\frac{3}{8}$ E $\frac{5}{8}$ F $\frac{13}{16}$ G $\frac{16}{19}$ H 1

Question 45 *(4 points)* **M-Step**
Assume that after one E-step, you get the following estimated assignment probability to cluster 1 for each movie, respectively: $\frac{1}{4}$, $\frac{1}{3}$, $\frac{1}{3}$, $\frac{2}{3}$, $\frac{3}{4}$. Now you would like to update the estimate of the parameters. You have $\hat{q} = 3/5$. The next step is an M-step. What will $\hat{q}$ be after this M-step?

A 0 B $\frac{1}{15}$ C $\frac{2}{5}$ D $\frac{7}{15}$ E $\frac{8}{15}$ F $\frac{3}{5}$ G $\frac{14}{15}$ H 1

Question 46 *(2 points)* You decide to try hard EM instead of soft EM. You use the same initialization of $\hat{p}$ and $\hat{q}$ as in Question 44. At convergence, you obtain the following clusters: cluster 0 : {I}, cluster 1: {G, S, T, M}. Suppose you instead initialized with $\hat{p}(X \mid C = 1) = \left(\frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}\right)$ and $\hat{p}(X \mid C = 0) = \left(\frac{1}{8}, \frac{1}{4}, \frac{3}{4}, \frac{1}{2}\right)$. Assume that each E-step results in a unique hard clustering. Which of the following describes the clusterings you would expect to see after convergence?

A cluster 0 : {T, M}, cluster 1: {G, S, I}
B cluster 0 : {G, S, I}, cluster 1: {T, M}
C cluster 0 : {I}, cluster 1: {G, S, T, M}
D cluster 0 : {G, S , T, M}, cluster 1: {I}
E cluster 0 : {G, I}, cluster 1: {S, T, M}
F cluster 0 : {S, T, M}, cluster 1: {G, I}
G cluster 0 : {}, cluster 1: {G, S, T, I, M}
H cluster 0 : {G, S, T, I, M}, cluster 1: {}

6.2 Gaussian Mixture Models

You have data points $x_1, \ldots, x_n$ that you want to split into two clusters ($y = 1$ or $y = 2$). You assume a Gaussian mixture model generated by a mixture of two Gaussians: $\mathcal{N}(\mu_1, \Sigma_1)$ and $\mathcal{N}(\mu_2, \Sigma_2)$. The mean vectors μ_i and covariance matrices Σ_i of each of the two clusters are unknown. You assume that the prior probability for a point being in class i is known and equal to w_i ($w_1 + w_2 = 1$). You perform the **hard** expectation maximization (EM) algorithm to cluster the data points.

Question 47 *(1 point)* The estimated cluster centers at convergence are independent of the initialization we use for the cluster centers.

A True B False

Question 48 *(1 point)* In the E-step, you update your estimates of μ_i and Σ_i.

A True B False

Question 49 *(1 point)* We want to fix $\Sigma_i, i = 1, 2$, to be some diagonal matrix. In this case, the only unknown parameters to estimate would be the cluster means μ_i. Is it True or False that for **any** choice of diagonal Σ_i, the hard EM algorithm is equivalent to (outputs the same means as) Lloyd's heuristic for k-means?
Note: A diagonal matrix is by definition a matrix that has nonzero elements only on its diagonal.

A True B False

Question 50 *(3 points)* **Hard EM**
Let $z_t \in \{1, 2\}$ be the cluster that point x_t is currently assigned to, and n_i be the number of points assigned to cluster i, and $n = n_1 + n_2$. You estimate Σ_1 and Σ_2 separately and do not assume any particular structure for them. To estimate Σ_1 and Σ_2, you maximize the data log-likelihood fixing the current cluster assignments. The notation "$t : A_t$" means "the set of all t for which statement A_t is true". Let $\widehat{\mu}_i = \frac{1}{n_i} \sum_{t:z_t=i} x_t$. When updating the estimate of Σ_1 in hard EM, the update can be written as $\hat{\Sigma}_1 = ...$

A $\frac{1}{n_1+1} \sum_{t:z_t=1} (x_t - \hat{\mu}_1)(x_t - \hat{\mu}_1)^\top$.

B $\frac{1}{nw_1} \sum_{t:z_t=1} (x_t - \hat{\mu}_1)(x_t - \hat{\mu}_1)^\top$.

C $\frac{1}{n_1} \sum_{t:z_t=1} (x_t - \hat{\mu}_1)(x_t - \hat{\mu}_1)^\top$.

D $\frac{1}{n_1+1} \sum_{t:z_t=1} (x_t - \hat{\mu}_1)(x_t - \hat{\mu}_1)^\top + \frac{1}{n_2+1} \sum_{t:z_t=2} (x_t - \hat{\mu}_2)(x_t - \hat{\mu}_2)^\top$.

E $\frac{1}{n_1} \sum_{t:z_t=1} (x_t - \hat{\mu}_1)(x_t - \hat{\mu}_1)^\top + \frac{1}{n_2} \sum_{t:z_t=2} (x_t - \hat{\mu}_2)(x_t - \hat{\mu}_2)^\top$.

F $\frac{1}{nw_1} \sum_{t:z_t=1} (x_t - \hat{\mu}_1)(x_t - \hat{\mu}_1)^\top + \frac{1}{nw_2} \sum_{t:z_t=2} (x_t - \hat{\mu}_2)(x_t - \hat{\mu}_2)^\top$.

G $\frac{1}{nw_1} \sum_{t:z_t=1} x_t x_t^\top$.

H $\frac{1}{n_1} \sum_{t:z_t=1} x_t x_t^\top$.

7 Generative Adversarial Networks

You train a generative adversarial network (GAN) with neural network discriminator D and neural network generator G. Let $z \sim \mathcal{N}(0, I)$, where I is the identity matrix, represent the random Gaussian input for G. The objective during training is given by

$$\min_G \max_D \mathbb{E}_{x \sim p_{\text{data}}}[\log D(x)] + \mathbb{E}_{z \sim \mathcal{N}(0,I)}[\log(1 - D(G(z)))],$$

where p_{data} is the data generating distribution.

Please answer the following questions.

Question 51 *(1 point)* If D and G both have enough capacity, i.e., if they can model arbitrary functions, the optimal G will be such that $G(z) \sim p_{\text{data}}$.

A True B False

Question 52 *(1 point)* The objective above can be interpreted as a two-player game between G and D.

A True B False

Question 53 *(2 points)* Suppose that the probability of a training sample x is $p_{\text{data}}(x) = \frac{1}{100}$ and the probability of x under G is $p_G(x) = \frac{1}{50}$. Suppose that the discriminator D is the globally optimal discriminator for G with the above loss.

What is the probability of D classifying x as being from the generator?

A $\frac{1}{2}$ B $\frac{1}{3}$ C $\frac{2}{3}$ D $\frac{1}{4}$ E $\frac{3}{4}$ F $\frac{1}{6}$ G 0 H 1