Data Analytics and Machine Learning (I26)
Department of Computer Science
Technical University of Munich

E0003
Place student sticker here

Note:

- During the attendance check a sticker containing a unique code will be put on this exam.
- This code contains a unique number that associates this exam with your registration number.
- This number is printed both next to the code and to the signature field in the attendance check list.

Maschinelles Lernen

Exam:	IN2064 / Endterm	**Date:**	Friday 24th February, 2023
Examiner:	Prof. Günnemann	**Time:**	17:00 – 19:00

	P 1	P 2	P 3	P 4	P 5	P 6	P 7	P 8	P 9
I									

Working instructions

- This exam consists of **16 pages** with a total of **9 problems**.
  Please make sure now that you received a complete copy of the exam.
- The total amount of achievable credits in this exam is 36 credits.
- Detaching pages from the exam is prohibited.
- Allowed resources:
  - Two-sided DIN A4 sheet of handwritten notes (a print of digitally handwritten notes is allowed).
- **No other material (e.g. books, cell phones, calculators) is allowed!**
- Physically turn off all electronic devices, put them into your bag and close the bag.
- There is scratch paper at the end of the exam (after problem 9).
- Write your answers only in the provided solution boxes or the scratch paper.
- If you solve a task on the scratch paper, clearly reference it in the main solution box.
- All sheets (including scratch paper) have to be returned at the end.
- **Only use a black or a blue pen (no pencils, red or greens pens!)**
- **For problems that say "Justify your answer" you only get points if you provide a valid explanation.**
- **For problems that say "Derive" you only get points if you provide a valid mathematical derivation.**
- **For problems that say "Prove" you only get points if you provide a valid mathematical proof.**
- If a problem does not say "Justify your answer", "Derive" or "Prove", it is sufficient to only provide the correct answer.

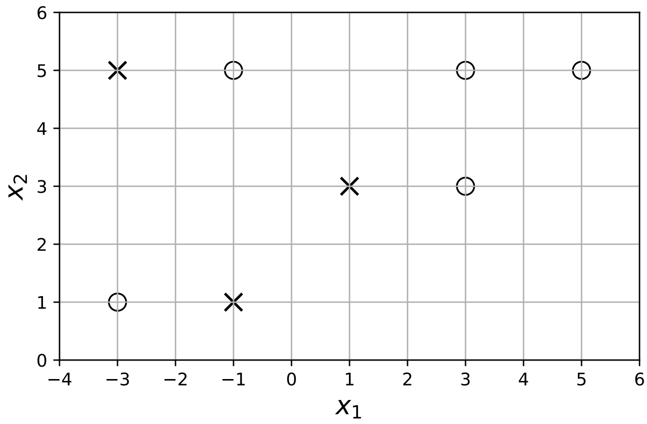
Left room from ___________ to ___________ / Early submission at ___________

☐ Exam empty

cit-ml-1-20230224-E0003-01
cit-ml-1-20230224-E0003-01

Problem 1 Decision trees (4 credits)

Consider the following two-dimensional classification dataset with the classes "0" (○ marker) and 1 (x marker).



0
1
2

a) Draw a decision tree of maximum depth 3 that correctly classifies all datapoints. Each decision node must be of the form $x_d \leq c$ with $d \in \{1, 2\}$ and $c \in \mathbb{R}$. Also annotate each edge with "True" or "False" and each leaf node with "0" or "1"

cit-ml-1-20230224-E0003-02
cit-ml-1-20230224-E0003-02
cit-ml-1-20230224-E0003-02

b) We now consider a modified form of decision trees that **only** allows for decision nodes of the form $a \cdot x_1 + b \leq x_2$ with $a, b \in \mathbb{R}$. In particular, nodes of the form $x_1 \leq c$ are **not allowed**.
Draw such a decision tree of maximum depth 2 that correctly classifies all datapoints. Also annotate each edge with "True" or "False" and each leaf node with "0" or "1"

0
1
2

Problem 2 Probabilistic inference (3 credits)

Consider an infinite number of barns arranged on a regular grid $\mathbb{Z}^2$, with $\mathbb{Z}$ being the set of all integers. An owl starts exploring the world at an unknown location $\mathbf{x}^{(0)} \in \mathbb{Z}^2$. Each day, it moves in one of four directions, according to the following distribution:

$$\Pr\left[\mathbf{x}^{(t+1)} = \mathbf{x}^{(t)} + \begin{bmatrix}-1\\0\end{bmatrix} \mid \mathbf{x}^{(t)}\right] = \frac{2}{8} \quad \Pr\left[\mathbf{x}^{(t+1)} = \mathbf{x}^{(t)} + \begin{bmatrix}+1\\0\end{bmatrix} \mid \mathbf{x}^{(t)}\right] = \frac{2}{8}$$
$$\Pr\left[\mathbf{x}^{(t+1)} = \mathbf{x}^{(t)} + \begin{bmatrix}0\\-1\end{bmatrix} \mid \mathbf{x}^{(t)}\right] = \frac{3}{8} \quad \Pr\left[\mathbf{x}^{(t+1)} = \mathbf{x}^{(t)} + \begin{bmatrix}0\\+1\end{bmatrix} \mid \mathbf{x}^{(t)}\right] = \frac{1}{8}$$

0
1

a) After two days, you find the owl sleeping in $\mathbf{x}^{(2)} = \begin{bmatrix}6 & 8\end{bmatrix}^\top$. List all possible starting locations, i.e. all $\mathbf{s} \in \mathbb{Z}^2$ such that $\Pr\left[\mathbf{x}^{(2)} = \begin{bmatrix}6 & 8\end{bmatrix}^\top \mid \mathbf{x}^{(0)} = \mathbf{s}\right] > 0$.

0
1
2

b) **Derive** the maximum likelihood estimate for the starting location $\mathbf{x}^{(0)}$, i.e. $\operatorname{argmax}_{\mathbf{s}} \Pr\left[\mathbf{x}^{(2)} = \begin{bmatrix}6 & 8\end{bmatrix}^\top \mid \mathbf{x}^{(0)} = \mathbf{s}\right]$.

cit-ml-1-20230224-E0003-04
cit-ml-1-20230224-E0003-04
cit-ml-1-20230224-E0003-04

Problem 3 Linear regression (5 credits)

We want to perform regularized linear regression (without bias) on a dataset with N samples $\mathbf{x}_i \in \mathbb{R}^d$ with corresponding targets y_i (represented compactly as $\mathbf{X} \in \mathbb{R}^{N\times d}$ and $\mathbf{y} \in \mathbb{R}^N$). You assume that your targets are normal distributed, i.e.,

$$p(y_i|\mathbf{x}_i, \mathbf{w}) = \mathcal{N}(\mathbf{x}_i^\top \mathbf{w}, 1) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(y_i - \mathbf{x}_i^\top \mathbf{w}\right)^2\right). \tag{3.1}$$

To add regularization, you choose a Laplace prior on the parameters $\mathbf{w} \in \mathbb{R}^d$, i.e.,

$$p(\mathbf{w}) = \frac{1}{2\lambda} \prod_{i=1}^{d} \exp\left(-\frac{|w_i|}{\lambda}\right). \tag{3.2}$$

with hyperparameter $\lambda > 0$.

a) **Derive** the negative logarithm of the posterior distribution, i.e., $-\log p(\mathbf{w}|\mathbf{X}, \mathbf{y})$ (up to some normalization constant).

0 1 2 3 4

b) What is the advantage/difference of having such a Laplace prior over a Gaussian prior? **Justify your answer!**

0 1

Problem 4 Optimization (6 credits)

Below, you are given two different functions and asked to **prove** convexity.

0
1
2
3

a) Prove that the subsequent function is convex in $\mathbf{x} \in \mathbb{R}^d_{>0}$, i.e., over the set of vectors solely consisting of positive entries $x_i > 0$ for all $i = 1, \ldots, d$:

$$f(\mathbf{x}) = \sum_{i=1}^{d} x_i \log x_i$$

Hint: *Remember that a matrix* $\mathbf{A} \in \mathbb{R}^{d \times d}$ *is positive semi-definitive, if* $\forall z \in \mathbb{R}^d : \mathbf{z}^\top \mathbf{A} \mathbf{z} \geq 0$

cit-ml-1-20230224-E0003-06
cit-ml-1-20230224-E0003-06
cit-ml-1-20230224-E0003-06

b) Let $f_1 : \mathbb{R}^d \to \mathbb{R}$ and $f_2 : \mathbb{R}^d \to \mathbb{R}$ be two convex functions. **Prove** that

$$h(\mathbf{x}) = \max\{f_1(\mathbf{x}), f_2(\mathbf{x})\}$$

is a convex function.
Note: For this, you are not allowed to use any convexity rules from the lecture without proving them.

0 1 2 3

Page empty

Problem 5 Deep Learning (5 credits)

The following code snippets all contain **exactly one error**. Your task is to spot the mistakes and explain how to fix it. **Justify your answer!**
We omitted variable initializations to avoid clutter. Assume that all variables were appropriately initialized.

0 1 2

a) Given an input $\mathbf{x} \in \mathbb{R}^d$, the subsequent class implements the ReLU layer $f(x) = \max(0, x)$ and the corresponding backward pass.

```
import numpy as np

class ReLU:
    def forward(self, inputs):
        self.cache = inputs
        out = np.maximum(inputs, 0)
        return out
    def backward(self, d_out):
        inputs = self.cache
        d_inputs = d_out * (inputs < 0)
        return d_inputs

relu = ReLU()
z = relu.forward(x)
d_x = relu.backward(1.0)
```

b) We have trained a model to perform multiclass classification over c classes on a dataset $\mathbf{X} \in \mathbb{R}^{n \times d}$ and one-hot encoded targets $y \in \{0,1\}^{n \times c}$ with $\sum_{j=1}^{c} y_{i,j} = 1 \quad \forall i \in [1, \dots, n]$. The model is defined as: outputs = ReLU($x@w_1 + b_1$)$@w_2 + b_2$. The model was trained to minimize the Cross Entropy between the one-hot encoded target and the prediction. Now, we want to obtain the normalized class probabilities as well as the predicted class.

0 1 2 3

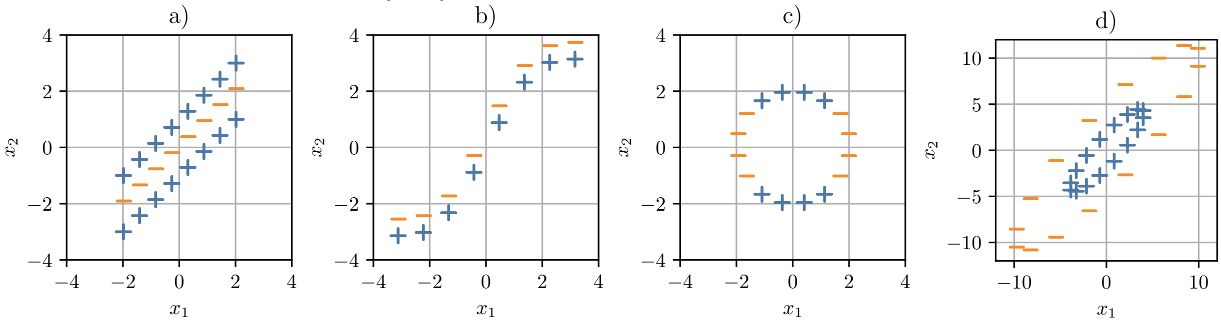
```
import torch

model.eval()
outputs = model.forward(x)
classprobs = torch.sigmoid(outputs)
y_hat = torch.argmax(classprobs, axis=1)
for i, (cp, yh) in enumerate(zip(classprobs, y_hat)):
    print(f"predicted class {yh} for sample {i} with probability {cp[yh]}")
```

Problem 6 Linear classification (4 credits)

0 1 2 3 4

We want to perform binary classification on four different datasets, $t \in \{a, b, c, d\}$, each consisting of N_t samples $\mathbf{x}_i \in \mathbb{R}^2$ with labels $y_i \in \{-, +\}$:



You already came up with transformations $\phi_1, ..., \phi_4$ that transform the respective datasets such that they are linearly separable:

$$\phi_1(\mathbf{x}) = \hat{x}_1 \hat{x}_2$$

$$\hat{\mathbf{x}} = \mathbf{x} \begin{bmatrix} \cos\frac{\pi}{4} & -\sin\frac{\pi}{4} \\ \sin\frac{\pi}{4} & \cos\frac{\pi}{4} \end{bmatrix} \tag{6.1}$$

$$\phi_2(\mathbf{x}) = x_2 - \sin(x_1) - x_1 \tag{6.2}$$

$$\phi_3(\mathbf{x}) = \left\| \begin{bmatrix} \frac{x_1}{2} \\ x_2 - x_1 \end{bmatrix} \right\|_2 \tag{6.3}$$

$$\phi_4(\mathbf{x}) = |x_1 - x_2| \tag{6.4}$$

Unfortunately, you forgot which transform belongs to which dataset. Assign the transformations $\phi_1, \phi_2, \phi_3, \phi_4$ to the datasets a, b, c, d such that the transformed datasets are linearly separable. **Justify your answer!**

Problem 7 Support Vector Machines and Kernels (4 credits)

You are given a dataset with N datapoints $\{(\mathbf{x}_i, y_i)\}_{i=1}^N$ with $\mathbf{x}_i \in \mathbb{R}^d$ and $y_i \in \{-1, 1\}$ representing the class of datapoint i. We use the augmentation trick $\mathbf{x} \mapsto \tilde{\mathbf{x}} = (\mathbf{x}, 1)$ to turn the affine decision function of an SVM classifier $h(\mathbf{x}) = \mathbf{w}^\top \mathbf{x} + b$ (with explicit bias term) into a linear function $\tilde{h}(\mathbf{x}) = \tilde{\mathbf{w}}^\top \tilde{\mathbf{x}}$ with $\tilde{\mathbf{w}} = (\mathbf{w}, b) \in \mathbb{R}^{d+1}$.

Now, we want to solve the adapted (maximum-margin) optimization problem

$$\begin{aligned} \min_{w} \quad & \frac{1}{2}\tilde{\mathbf{w}}^\top \tilde{\mathbf{w}} \\ \text{subject to} \quad & y_i \tilde{\mathbf{w}}^\top \tilde{\mathbf{x}}_i - 1 \geq 0 \qquad i = 1, \dots, N \end{aligned}$$

a) What is the Lagrangian function $L(\tilde{\mathbf{w}}, \boldsymbol{\alpha})$ associated to the above problem, with $\alpha_i \geq 0$ corresponding to the Lagrangian multipliers.

0
1

b) **Derive** the corresponding dual function $g(\boldsymbol{\alpha})$. It suffices to simplify $g(\boldsymbol{\alpha})$ such that it does not contain any minimization or maximization term.

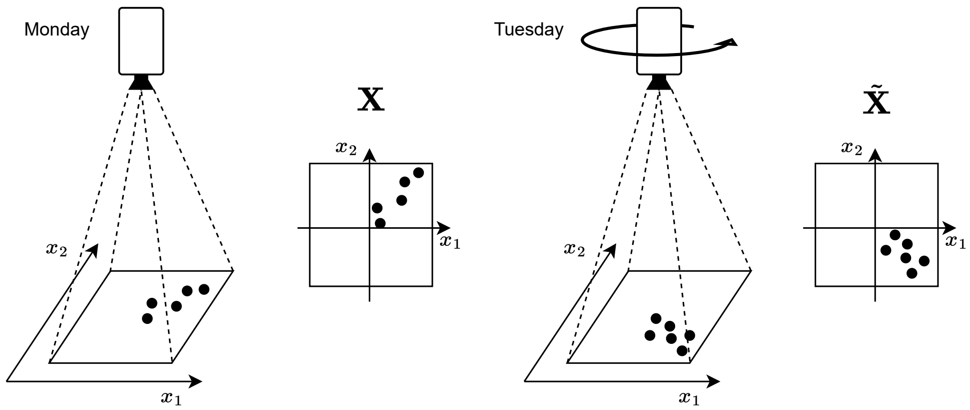
0
1
2
3

Hint: *The Lagrangian function $L(\tilde{\mathbf{w}}, \boldsymbol{\alpha})$ is convex in $\tilde{\mathbf{w}}$.*

Problem 8 PCA (3 credits)

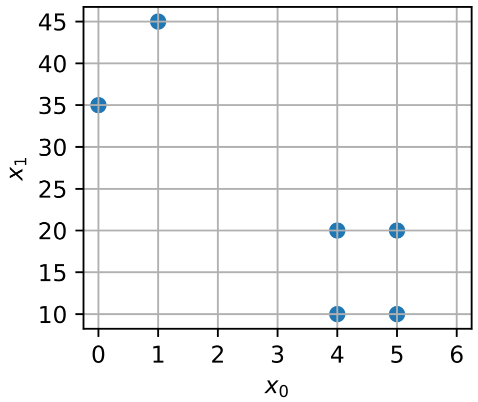
0
1
2
3

On Monday, you experimented with growing bacteria and took a photo of the result. You recorded the positions of bacteria as illustrated below. Each position is a two-dimensional coordinate, with the origin in the middle of the camera's frame. The positions are saved in a data matrix $\mathbf{X} \in \mathbb{R}^{N\times 2}$. On Tuesday, you repeated the experiment but did not set up the camera at the same angle. Tuesday's measurements are denoted with $\tilde{\mathbf{X}} \in \mathbb{R}^{M\times 2}$.
Since you assume the positions will follow the **same distribution** every day, you want to rotate the data recorded on Tuesday to **match the direction and shape** of the data from Monday. Unfortunately, the only data processing technique you know is PCA. Fortunately, this is enough to solve this problem. **Propose a solution and justify your answer.**
You have a function `PCA(D)` at your disposal, which takes data matrix $\mathbf{D} \in \mathbb{R}^{a\times b}$ and returns $\mathbf{\Gamma} \in \mathbb{R}^{b\times b}$ corresponding to the principal components, and $\mathbf{\Lambda} \in \mathbb{R}^{b}$ corresponding to the eigenvalues. You also know the commands for basic matrix manipulation: addition, subtraction, multiplication and transpose.
Assume that PCA always gives you the desired eigenvectors, that is, ignore the potential sign flips in $\mathbf{\Gamma}$.
Note: Figure below is just for illustration purposes. The angle and the values N and M are not given.



Problem 9 Clustering (2 credits)

You are given the following two-dimensional dataset $\mathbf{X} \in \mathbb{R}^{n \times 6}$:



a) What are the globally optimal cluster centers μ that minimize the k-means objective

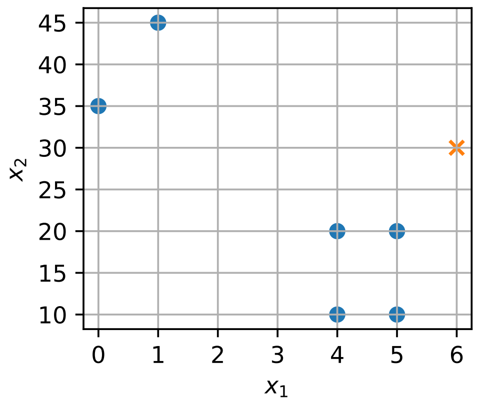
0
1

$$J(\mathbf{X}, \mathbf{Z}, \mu) = \sum_{i=1}^{N} \sum_{k=1}^{K} \mathbf{z}_{ik} ||\mathbf{x}_i - \mu_k||_2^2 \tag{9.1}$$

with the assignment to the closest cluster centers $\mathbf{Z}$.

b) Now assume you want to infer the corresponding cluster for a new datapoint without updating the cluster centers μ. To what cluster center in μ does the new point (x) correspond to? **Justify your answer!**

0
1



Additional space for solutions–clearly mark the (sub)problem your answers are related to and strike out invalid solutions.

cit-ml-1-20230224-E0003-14
cit-ml-1-20230224-E0003-14
cit-ml-1-20230224-E0003-14

cit-ml-1-20230224-E0003-16
cit-ml-1-20230224-E0003-16
cit-ml-1-20230224-E0003-16