Data Analytics and Machine Learning
Department of Informatics
Technical University of Munich

Eexam
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.

Machine Learning for Graphs and Sequential Data

Exam:	IN2323 / Endterm	**Date:**	Friday 19th August, 2022
Examiner:	Prof. Dr. Stephan Günnemann	**Time:**	08:15 – 09:30

	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 72 credits.
- Detaching pages from the exam is prohibited.
- Allowed resources:
  - one A4 sheet of handwritten notes (two sides, not digitally written and printed).
- **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 _____

Problem 1 Generative models (6 credits)

Recall the variational autoencoder (VAE), which can be summarized by the following pseudocode

$$\begin{aligned}\mu, \sigma &= f_\theta(\mathbf{x}) \\ \epsilon &\sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \\ \mathbf{z} &= \epsilon * \sigma + \mu \\ \tilde{\mathbf{x}} &= g_\phi(\mathbf{z}),\end{aligned}$$

and is trained to model a distribution $p(\mathbf{x})$ via maximization of the evidence lower bound.
We now want to develop a VAE that can model a distribution of images conditioned on a label, i.e. $p(\mathbf{x} \mid y)$ where $\mathbf{x} \in \mathbb{R}^d$ is the image and y is the label, for example, “dog” or “cat”.

0 1 2

a) Modify the above pseudocode for the VAE to condition the model on the label y. You can change the dimensions of functions’ domains and codomains if necessary.

0 1 2 3 4

b) After training is completed we want to sample new images from our variational autoencoder. Write the pseudocode to generate an image given a label y. You should use the solution to the previous problem as a starting point.

Problem 2 **Robustness (10 credits)**

0 1 2 3 4 5 6 7 8 9 10

We are interested in robustness certification for a model with discrete input data $\mathbf{x} \in \{0, 1, \dots, C\}^N$ and an adversary that changes exactly $\delta \in \mathbb{N}$ elements of $\mathbf{x}$.
The perturbation set can be expressed as

$$\mathcal{P}(\mathbf{x}) = \left\{ \tilde{\mathbf{x}} \in \{0, 1, \dots, C\}^N \middle| ||\mathbf{x} - \tilde{\mathbf{x}}||_0 = \delta \right\} \tag{2.1}$$

with $||\mathbf{x}||_0 = \sum_{n=1}^{N} \mathbb{I}[x_n \neq 0]$.
Specify a set of **linear constraints** on $\tilde{\mathbf{x}}$ to model the perturbation set in Eq. (2.1). You may introduce at most $\mathcal{O}(N)$ constraints and $\mathcal{O}(N)$ variables. You are allowed to use integer-valued variables.
Note: A linear constraint is an equality or inequality between two expressions that are **linear functions** of the variables.

$$y \in \{0,1\}^N$$

$$s = \sum_{i=1}^{N} y_i$$

$$\|x - \tilde{x}\|_0 \leq y_i c$$

$$\|x - \tilde{x}\|_0 \geq y_i c$$

Problem 3 **Autoregressive models (8 credits)**

You are given an AR(3) model according to the formula

$$X_t = 17 + 4X_{t-1} + \frac{1}{4}X_{t-2} - X_{t-3} + \varepsilon_t \,,$$

with independently distributed noise variables $\varepsilon_t \sim \mathcal{N}(0, \sigma)$.

0 1 2 3

a) Write down the characteristic polynomial $\Phi(z)$ and show that it can be factorised according to $(2+z)\left(z^2 - \frac{9}{4}z + \frac{1}{2}\right)$.

0 1 2 3 4 5

b) Decide if the process X_t is stationary. Justify your answer.

$$X_t = 17 + 4z X_t + \frac{1}{4} z^2 X_t - z^3 X_t + \varepsilon_t$$

$$(1 - 4z - \frac{1}{4}z^2 + z^3) = 0$$

$$(z+2)\left(z^2 - \frac{9}{4}z + \frac{1}{2}\right)$$

$$z^3 - \frac{9}{4}z^2 + \frac{1}{2}z + 2z^2 - \frac{9}{2}z + 1$$

$$= z^3 - \frac{1}{4}z^2 - 4z + 1$$

$$\frac{32}{64}$$

$$(z+2)\left(z^2 - \frac{9}{4}z + \frac{1}{2}\right) = 0$$

$z = -2$ or $\left(z - \frac{9}{8}\right)^2 = \frac{49}{64} = \left(\frac{7}{8}\right)^2$

$$\frac{81}{64}$$

$$z - \frac{9}{8} = \frac{7}{8} \quad = -\frac{7}{8}$$

$$z = \frac{16}{8} = 2 \quad = \frac{2}{4}$$

$$= \frac{1}{4}$$

no any study.

Problem 4 Hidden Markov Models (10 credits)

Consider a hidden Markov model with 2 states $\{1, 2\}$ and 6 possible observations $\{\mathsf{p}, \mathsf{a}, \mathsf{n}, \mathsf{e}, \mathsf{r}, \mathsf{t}\}$. The initial distribution π, transition probabilities **A** and emission probabilities **B** are

$$\pi = \begin{array}{c} 1 \\ 2 \end{array} \begin{pmatrix} 1/5 \\ 4/5 \end{pmatrix} \qquad \mathbf{A} = \begin{array}{c} \\ 1 \\ 2 \end{array} \begin{array}{c} \begin{array}{cc} 1 & 2 \end{array} \\ \begin{pmatrix} 1/5 & 4/5 \\ 3/5 & 2/5 \end{pmatrix} \end{array} \qquad \mathbf{B} = \begin{array}{c} \\ 1 \\ 2 \end{array} \begin{array}{c} \begin{array}{cccccc} \mathrm{p} & \mathrm{a} & \mathrm{n} & \mathrm{e} & \mathrm{r} & \mathrm{t} \end{array} \\ \begin{pmatrix} 0 & 1/5 & 0 & 2/5 & 0 & 2/5 \\ 1/5 & 1/5 & 1/5 & 1/5 & 1/5 & 0 \end{pmatrix}, \end{array}$$

where $\mathbf{A}_{ij}$ specifies the probability of transitioning from state *i* to state *j*.

a) You have observed the sequence $X = [\mathrm{pattern}]$. Specify all probability distributions $\mathbb{P}()$ that correspond to smoothing / offline inference on X.
Note: You do not need to perform any calculations or insert parameter values.

☐ 0
☐ 1
☐ 2

b) Write down the MAP objective given the observed sequence $X = [\mathrm{pattern}]$.

☐ 0
☐ 1

c) In another instance, you observe the sequence $X = [\mathrm{tea}]$. Given X, what is $\mathbb{P}(Z_3|X)$? [An unnormalised vector suffices]. Justify your answer. What is this type of inference called?

☐ 0
☐ 1
☐ 2
☐ 3
☐ 4
☐ 5
☐ 6
☐ 7

$p(z_i \mid x_{1:T})$ $i \in \{1,2,3,4,5,6\}$

find the max probability sequence of $z_{1:T}$ give $x_{1:T}$

$$\text{argmax}\ p(z_{1:T} \mid x_{1:T})$$

---

online

$$\alpha_1 = \pi \odot B = \begin{pmatrix} \frac{1}{5} \\ \frac{4}{5} \end{pmatrix} \odot \begin{pmatrix} \frac{2}{5} \\ 0 \end{pmatrix} = \begin{pmatrix} \frac{2}{25} \\ 0 \end{pmatrix}$$

$$\alpha_2 = B \odot (A^T \alpha_1)$$

$$= \begin{pmatrix} \frac{2}{5} \\ \frac{1}{5} \end{pmatrix} \odot \begin{pmatrix} \frac{1}{5} & \frac{3}{5} \\ \frac{4}{5} & \frac{2}{5} \end{pmatrix} \cdot \begin{pmatrix} \frac{2}{25} \\ 0 \end{pmatrix}$$

$$= \begin{pmatrix} \frac{2}{5} \\ \frac{1}{5} \end{pmatrix} \odot \begin{pmatrix} \frac{2}{125} \\ \frac{8}{125} \end{pmatrix}$$

$$= \begin{pmatrix} \frac{4}{625} \\ \frac{8}{625} \end{pmatrix}$$

$$\alpha_3 = \begin{pmatrix} \frac{1}{5} \\ \frac{1}{5} \end{pmatrix} \odot \begin{pmatrix} \frac{1}{5} & \frac{3}{5} \\ \frac{4}{5} & \frac{2}{5} \end{pmatrix} \begin{pmatrix} \frac{4}{625} \\ \frac{8}{625} \end{pmatrix}$$

$4 + 24 = 28$

$16 + 16 = 32$

$$p(z_3 \mid x_{1:3}) = \begin{pmatrix} \frac{28}{\cdot\cdot} \\ \frac{32}{\cdot\cdot} \end{pmatrix}$$

Problem 5 Graph learning & Variational inference (10 credits)

Consider the following probabilistic model for generating a **directed, weighted** graph with N nodes, continuous adjacency matrix $\mathbf{A} \in \mathbb{R}^{N \times N}$ and two communities, represented by vector $\mathbf{z} \in \{0,1\}^N$:

$$p_\lambda(\mathbf{A} \mid \mathbf{z}) = \prod_{n=1}^{N} \prod_{m=1}^{N} p_\lambda(A_{n,m} \mid z_n, z_m) \tag{5.1}$$

$$p_\theta(\mathbf{z}) = \prod_{n=1}^{N} \text{Bern}(z_n \mid \theta) = \prod_{n=1}^{N} \theta^{z_n} \cdot (1-\theta)^{1-z_n} \tag{5.2}$$

with $\theta \in [0,1]$. The conditional density $p_\lambda(A_{n,m} \mid z_n, z_m)$ will be specified later.
In the following, assume that we have observed a single graph $\mathbf{A} \in \mathbb{R}^{N \times N}$. We want to perform **mean-field variational inference** with variational family

$$q_\phi(\mathbf{z}) = \prod_{n=1}^{N} \text{Bern}(z_n \mid \phi_n) = \prod_{n=1}^{N} \phi_n^{z_n} \cdot (1-\phi_n)^{1-z_n}. \tag{5.3}$$

Note that $\phi \in [0,1]^N$, i.e. we have one parameter per node.

0 1 2

a) Why is evaluating the ELBO $\mathcal{L}((\lambda,\theta),\phi) = \mathbf{E}_{\mathbf{z}\sim q_\phi}\left[\log p_{\lambda,\theta}(\mathbf{A},\mathbf{z}) - \log q_\phi(\mathbf{z})\right]$ not tractable for large graphs (e.g. $N > 1000$)?

0 1 2 3 4

b) Assume that we approximate the ELBO with a single Monte Carlo sample $\mathbf{z} \in \{0,1\}^N$, i.e.

$$\mathcal{L}((\lambda,\theta),\phi) \approx \log p_{\lambda,\theta}(\mathbf{A},\mathbf{z}) - \log q_\phi(\mathbf{z}). \tag{5.4}$$

Let

$$p_\lambda(A_{n,m} \mid z_n, z_m) = \begin{cases} \lambda_1 \exp(-\lambda_1 A_{n,m}) & \text{if } A_{n,m} \geq 0 \wedge z_n = z_m, \\ \lambda_2 \exp(-\lambda_2 A_{n,m}) & \text{if } A_{n,m} \geq 0 \wedge z_n \neq z_m, \\ 0 & \text{else.} \end{cases}$$

with $\lambda_1, \lambda_2 > 0$. Assume that λ_2, θ and ϕ are fixed.
Prove that the optimal value of λ_1, i.e. the value that maximizes $\log p_{\lambda,\theta}(\mathbf{A},\mathbf{z}) - \log q_\phi(\mathbf{z})$ is

$$\lambda_1^* = \frac{|\{n,m \mid z_n = z_m\}|}{\sum_{n,m|z_n=z_m} A_{n,m}}.$$

Note: You may also write on the next page.

c) To allow optimization w.r.t. ϕ, we want to apply the reparameterization trick. Specify a base distribution $b(\epsilon)$ and a transformation $T(\epsilon, \phi)$ such that

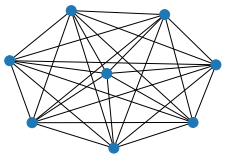
$$\mathbf{E}_{\mathbf{z}\sim q_\phi}\left[\log p_{\lambda,\theta}(\mathbf{A},\mathbf{z}) - \log q_\phi(\mathbf{z})\right] = \mathbf{E}_{\epsilon\sim b}\left[\log p_{\lambda,\theta}(\mathbf{A}, T(\epsilon,\phi)) - \log q_\phi(T(\epsilon,\phi))\right]. \tag{5.5}$$

0
1
2
3
4

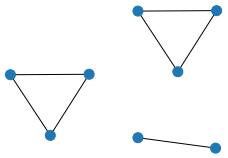
Problem 6 Graphs – Laws & patterns (8 credits)

0 1 2 3 4 5 6 7 8

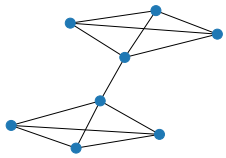
You are given four graphs (a-d), each consisting of eight nodes. You are further given four eigenspectra (1-4), i.e. eigenvalues of the graph Laplacian ordered in ascending order. Assign each of the graphs (a-d) to an eigenspectrum (1-4). Justify your answer.



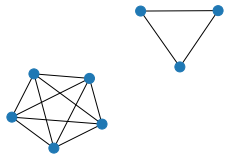
(a)



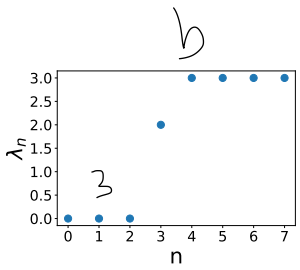
(b)



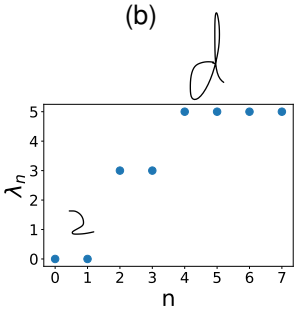
(c)



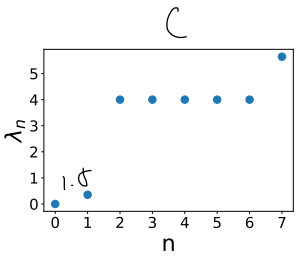
(d)



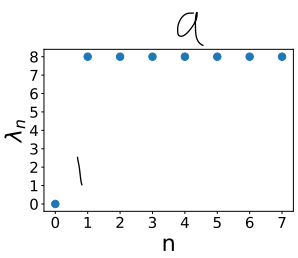
(1)



(2)



(3)



(4)

Problem 7 **Page Rank (8 credits)**

The PageRank scores (without teleports) of the graphs a-d have been computed with power iteration. Match the graphs a-d with the results 1-4. Justify your answer.

0 1 2 3 4 5 6 7 8

1. Does not converge.
2. Does not converge.
3. Converges to $r_A = 0.167$, $r_B = 0.167$, $r_C = 0.167$, $r_D = 0.5$.
4. Converges to $r_A = 0.125$, $r_B = 0.375$, $r_C = 0.25$, $r_D = 0.25$.

Problem 8 Graph Neural Networks (6 credits)

Below, you can find three different types of Graph Neural Network modules. The node embedding $h_u^{(t+1)}$ of node u at layer $t+1$ is calculated with:

- Network Propagation (NP): $h_u^{(t+1)} = \sum_{v \in N(u) \cup \{u\}} h_v^{(t)}$
- Graph Convolution (GCN): $h_u^{(t+1)} = \phi_{gcn}(h_u^{(t)}, \oplus_{v \in N(u)} \psi_{gcn}(h_v^{(t)}))$
- Message Passing (MP): $h_u^{(t+1)} = \phi_{mp}(h_u^{(t)}, \oplus_{v \in N(u)} \psi_{mp}(h_v^{(t)}, h_u^{(t)}))$

where $\oplus$ is some permutation invariant function without learnable parameters, the functions ψ_{gcn}, ψ_{mp} transform hidden features, functions ϕ_{gcn}, ϕ_{mp} are update functions and $N(u)$ is the neighbourhood of node u.

0 1 2 3

a) Prove that network propagation is a special case of graph convolution.
Hint: You can do this by providing specific realizations of $\oplus$, ψ_{gcn} and ϕ_{gcn}.

0 1 2 3

b) Prove that graph convolution is a special case of message passing.
Hint: You can do this by providing specific realizations of ψ_{mp} and ϕ_{mp}.

$\phi_{gcn}(a, b) = a + b$

$\varphi_{gcn}(x) = x$

$\oplus = \sum_{v \in \mathcal{N}(u)}$

$\phi_{mp} = \phi_{gcn}$

$\varphi_{mp}(h_v^{(t)}, h_u^{(t)})$

$= \varphi_{gcn}(h_v^{(t)})$

Problem 9 Limitations of Graph Neural Networks (6 credits)

a) Briefly explain two challenges when attacking GNNs using adversarial attacks.

0 1 2 3 4

b) We model the absence or presence of an edge in a graph with N nodes using a binary vector $\mathbf{x} \in \{0,1\}^{N^2}$. Now, we want to use randomized smoothing to certify that a smoothed classifer using GNNs as base-classifers is robust against attacks on the graph structure.

0 1 2

Recall that a smoothed classifier $g(\mathbf{x})_c$ returns the probability that the base classifier f classifies a smoothed sample $\tilde{\mathbf{x}} \sim \phi(\mathbf{x})$ as class c, i.e. $g(\mathbf{x})_c := \mathbb{P}(f(\phi(\mathbf{x})) = c)$ with a randomization scheme $\phi(\mathbf{x})$.

What is the problem when we want to use Gaussian noise as our randomization scheme? How could that problem be solved?

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