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

Ecorrection
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	P 10
I										

Working instructions

- This exam consists of **16 pages** with a total of **10 problems**.
  Please make sure now that you received a complete copy of the exam.
- The total amount of achievable credits in this exam is 86 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 10).
- 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 Normalizing flows (8 credits)

You are given the task of density estimation on $\mathbb{R}^2$ and plan on using normalizing flows. In the following we present some candidate transformations that will be used for **reverse parameterization.** For each of the transformations, state if it can be used to define a normalizing flow and justify your answers.
In all cases, the input is a vector $\boldsymbol{x} = \begin{bmatrix} x_1 & x_2 \end{bmatrix}^T$. We denote the output of the transformation with $\boldsymbol{z} \in \mathbb{R}^2$.

0 1 2

a)

$$\boldsymbol{A} = \boldsymbol{W}^T \boldsymbol{W}$$
$$\boldsymbol{z} = \boldsymbol{A}\boldsymbol{x},$$

where $\boldsymbol{W} \in \mathbb{R}^{2\times 2}$.

No.
The transformation may be non-invertible (consider $mW = 0$).
1 point for correct answer.
Full points for showing with a counterexample or Jacobian proof.

0 1 2

b)

$$\boldsymbol{z} = \begin{bmatrix} x_1^2 & x_2^2 \end{bmatrix}^T$$

No.
The transformation is not a bijection.
1 point for correct answer.
Full points for stating a counterexample or proof via Jacobian etc.

0 1 2

c)

$$z_1 = \boldsymbol{V}\text{ReLU}(\boldsymbol{W}x_2 + \boldsymbol{b})$$
$$\boldsymbol{z} = \begin{bmatrix} z_1 & x_2 \end{bmatrix}^T,$$

where $\boldsymbol{W} \in \mathbb{R}^{h\times 1}$, $\boldsymbol{V} \in \mathbb{R}^{1\times h}$, $\boldsymbol{b} \in \mathbb{R}^h$ and ReLU is applied elementwise.

No.
This is similar to a Real NVP architecture as the inverse of x_2 is easy to obtain by asigning $x_2 = z_2$. However, the $x_1 \mapsto z_1$ does not depend on x_1 and all the points are mapped to a single point (which depends on x_2). So the whole transformation is not a bijection.
1 point for correct answer.
Full points for this explanation or counterexample etc.

d)

0
1
2

$$\boldsymbol{z} = \boldsymbol{a} \odot \boldsymbol{x} + \boldsymbol{b},$$

where $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^2$ and $\odot$ is the elementwise product.

Yes if $\boldsymbol{a} \neq 0$.
Full points for saying yes and showing the inverse.
Full points for saying yes and saying it is not invertible when $\boldsymbol{a} = 0$.
Full points for saying no and showing the counterexample, $\boldsymbol{a} = 0$.
No points for only saying yes/no, without correct explanation.

Problem 2 Variational inference (10 credits)

0 1 2 3 4 5 6 7 8 9 10

Suppose we are given a latent variable model for a sequence of observations $x_1, \dots, x_N \in \{0, 1\}$ and latent variables $z_1, \dots, z_N \in [0, 1]$ with

$$p(z_1, \dots, z_N) = \prod_{n=1}^{N} \text{Beta}(z_n \mid \alpha, \beta) = \prod_{n=1}^{N} \frac{1}{B(\alpha, \beta)} z_n^{\alpha-1}(1 - z_n)^{\beta-1}$$

$$p(x_1, \dots, x_N | z_1, \dots, z_N) = \prod_{n=1}^{N} \text{Bern}(x_n \mid z_n) = \prod_{n=1}^{N} z_n^{x_n}(1 - z_n)^{1-x_n}$$

with parameters $\alpha, \beta > 0$ and normalizing constant $B(\alpha, \beta)$. We define the variational distribution

$$q(z_1, \dots, z_N) = \prod_{n=1}^{N} \text{Beta}(z_n \mid \gamma, \delta) = \prod_{n=1}^{N} \frac{1}{B(\gamma, \delta)} z_n^{\gamma-1}(1 - z_n)^{\delta-1}$$

with parameters $\gamma, \delta > 0$.
Assume that α, β are known and fixed. Prove or disprove the following statement:
There **exist** observations $x_1, \dots, x_N \in \{0, 1\}$ and values of $\gamma, \delta > 0$ such that the ELBO is tight, i.e. $\exists x_1, \dots, x_N, \exists \gamma, \delta : \log(p(x_1, \dots, x_N)) = \mathcal{L}((\alpha, \beta), (\gamma, \delta))$.

In the following, we use bold $\boldsymbol{x}$ and $\boldsymbol{z}$ for the two sequences.
The ELBO is tight if $\text{KL}(p(\boldsymbol{z}|\boldsymbol{x})||q(\boldsymbol{z})) = 0$, i.e. $p(\boldsymbol{z}|\boldsymbol{x}) = q(\boldsymbol{z})$. ✓ ✓
We begin by determining $p(\boldsymbol{z}|\boldsymbol{x})$.
Applying Bayes formula shows that

$$p(\boldsymbol{z}|\boldsymbol{x}) \propto p(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z}) \checkmark \checkmark \tag{2.1}$$

$$\propto \left(\prod_{n=1}^{N} z_n^{x_n}(1 - z_n)^{1-x_n}\right)\left(\prod_{n=1}^{N} z_n^{\alpha-1}(1 - z_n)^{\beta-1}\right) \checkmark \tag{2.2}$$

$$= \prod_{n=1}^{N} z_n^{\alpha-1+x_n}(1 - z_n)^{\beta-1+(1-x_n)} \checkmark \tag{2.3}$$

$$= \prod_{n=1}^{N} z_n^{\alpha-1+x_n}(1 - z_n)^{\beta-x_n} \tag{2.4}$$

$$\propto \prod_{n=1}^{N} \text{Beta}(z_n \mid \alpha + x_n, \beta - x_n + 1) \tag{2.5}$$

The last few simplifications are not necessary for getting credits
Alternatively, we can use the fact that the Beta distribution is the conjugate prior of the Bernoulli distribution to immediately arrive at the above result. ✓ ✓ ✓ ✓
Please account for follow-up mistakes from calculating an incorrect $p(z|x)$, e.g. caused by making mistakes in the multiplication
If all observations x_n have the same value ✓ ✓, i.e. $\forall n : x_n = c$ with $c \in \{0, 1\}$, we can ensure that $p(z|x) = q(z)$ by setting $\gamma = \alpha + c$ ✓ and $\delta = \beta - c + 1$ ✓.
You can give 1 credit if someone does not specify x_n, δ, γ (correctly) but says something along the lines of: "We can choose parameters to make exponents match". Or: "Same type of distribution, we can choose parameters s.t. distributions are the same".

Problem 3 Robustness (9 credits)

0 1 2 3 4 5 6 7 8 9

In the lecture, we have derived a convex relaxation for the ReLU activation function. Now, we want to generalize this result to the flexible ReLU (FReLU) activation function

$$FReLU(x) = \begin{cases} x + b & \text{if } x > 0 \\ b & \text{if } x \leq 0 \end{cases}$$

with variable input $x \in \mathbb{R}$ and **constant parameter** $b \in \mathbb{R}$.

Let $y \in \mathbb{R}$ be the variable we use to model the function's output. Now, given input bounds $l, u \in \mathbb{R}$ with $l \leq x \leq u$, provide a set of **linear constraints** corresponding to the convex hull of $\{ \begin{bmatrix} x & FRelu(x) \end{bmatrix}^T \mid l \leq x \leq u \}$.

Hint: You will have to make a case distinction to account for different ranges of l and u.

We have to distinguish three cases:
Case I: $l, u \leq 0$ 1P for identifying this case distinction
A single constraint suffices: $y = b$ 1P for constraint
Case II: $l, u > 0$ 1P for identifying this case distinction
A single constraint suffices: $y = x + b$ 1P for constraint
Case III: $l < 0, u > 0$ 1P for identifying this case distinction
Now, we need three linear constraints:

$$y \geq b$$
$$y \geq x + b$$
$$y \leq \frac{u}{u - l}(x - l) + b$$

1P for constraint I, 1P for constraint II, 2P for (complex) constraint III

Problem 4 Autoregressive models (8 credits)

0 1 2 3 4 5

a)
An autoregressive process of order p, AR(p), is defined as:

$$X_t = c + \sum_{i=1}^{p} \varphi_i X_{t-i} + \varepsilon_t \,,$$

with independently distributed noise variables $\varepsilon_t \sim \mathcal{N}(0, \sigma^2)$.
Provided that the AR(p) is stationary, derive its first moment $\mathbb{E}[X_t]$ as a function of c and φ_i.

$$\mathbb{E}[X_t] = \mathbb{E}\left[c + \sum_{i=1}^{p} \varphi_i X_{t-i} + \varepsilon_t\right] \checkmark \qquad \text{Linearity}$$

$$\mathbb{E}[X_t] = c + \sum_{i=1}^{p} \varphi_i \mathbb{E}\left[X_{t-i}\right] + \mathbb{E}\left[\varepsilon_t\right] \checkmark \qquad \mathbb{E}\left[\varepsilon_t\right] = 0 \;\forall t \text{ by definition}$$

$$\mathbb{E}[X_t] = c + \sum_{i=1}^{p} \varphi_i \mathbb{E}\left[X_{t-i}\right] + 0 \checkmark \qquad \mathbb{E}[X_t] = \mu \;\forall t \text{ by stationarity}$$

$$\mu = c + \sum_{i=1}^{p} \varphi_i \mu \qquad \checkmark$$

$$\mu = c + \mu \sum_{i=1}^{p} \varphi_i$$

$$\mu = \frac{c}{1 - \sum_{i=1}^{p} \varphi_i} \qquad \checkmark$$

0 1 2 3

b) Let us define a process X_t as

$$X_t = \sin^2\left(-\frac{\pi}{2}t\right) + \frac{2}{3}X_{t-1} + \varepsilon_t$$

with independently distributed noise variables $\varepsilon_t \sim \mathcal{N}(0, \sigma^2)$.
Decide if the process X_t is stationary. Justify your answer.

We have

$$\mathbb{E}[X_t] = \mathbb{E}\left[\sin^2\left(-\frac{\pi}{2}t\right)\right] + \mathbb{E}\left[\frac{2}{3}X_{t-1}\right] + \mathbb{E}[\varepsilon_t] = \sin^2\left(-\frac{\pi}{2}t\right) + \mathbb{E}\left[\frac{2}{3}X_{t-1}\right] \checkmark$$

Assume that the process was stationary, i.e. $\mathbb{E}[X_t] = \mu \forall t$ with constant μ. Then, we would have

$$\mu = \sin^2\left(-\frac{\pi}{2}t\right) + \frac{2}{3}\mu \checkmark$$

$$\Longleftrightarrow \frac{1}{3}\mu = \sin^2\left(-\frac{\pi}{2}t\right)$$

This contradicts our assumption that μ is a constant ✓. This is a justify, not a prove task. You can also give full credits for any other reasonable explanation for why the process is not stationary (which should boil down to "we have some time-dependent component").
This is not an *AR*(p) process, i.e. the rule based on characteristic polynomials is not applicable. You can still give 1 point for applying the characteristic polynomial rule correctly.

Problem 5 Hidden Markov Models (9 credits)

Consider a hidden Markov model with 2 states $\{1, 2\}$ and 4 possible observations $\{\mathrm{c}, \mathrm{e}, \mathrm{i}, \mathrm{n}\}$. The initial distribution π, transition probabilities $\boldsymbol{A}$ and emission probabilities $\boldsymbol{B}$ are

$$\pi = \begin{matrix} 1 \\ 2 \end{matrix} \begin{pmatrix} 2/5 \\ 3/5 \end{pmatrix} \qquad \boldsymbol{A} = \begin{matrix} \\ 1 \\ 2 \end{matrix} \begin{matrix} 1 & 2 \\ \end{matrix} \begin{pmatrix} 1/3 & 2/3 \\ 3/5 & 2/5 \end{pmatrix} \qquad \boldsymbol{B} = \begin{matrix} \\ 1 \\ 2 \end{matrix} \begin{matrix} \mathrm{c} & \mathrm{e} & \mathrm{i} & \mathrm{n} \\ \end{matrix} \begin{pmatrix} 2/5 & 1/5 & 1/5 & 1/5 \\ 0 & 1/5 & 3/5 & 1/5 \end{pmatrix},$$

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

a) We have observed the sequence $X_{1:3} = [\mathrm{nic}]$. What is the most likely latent state Z_3 given these observations? Justify your answer. What is this type of inference called?

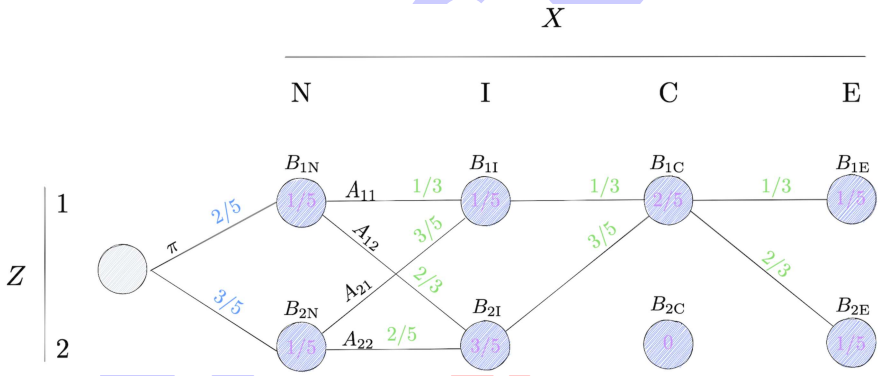
0 1 2 3

This type of inference is called *filtering / smoothing* ✓. To compute $P(Z_3|X_{1:3})$ we can either use the forward algorithm or identify from the emission matrix B that "c" can only be observed form latent state $Z = 1$ ✓ ✓. Hence, we get:

$$P(Z_3|X_{1:3}) = \begin{pmatrix} 1 \\ 0 \end{pmatrix}.$$

b) The full observed sequence is $X_{1:4} = [\mathrm{nice}]$. What is the most likely latent state sequence $Z_{1:4}$ given these observations? Justify your answer.

0 1 2 3 4 5 6



1 We note that "c" can only be observed from latent state $Z = 1$. Hence, all paths going through B_{2C} can be ignored. ✓

2 Since all paths have to go through B_{1C} and we have $B_{1E} = B_{2E}$, the maximising path has to also go along B_{2E} as the edge weight $A_{12} = 2/3$ is larger than $A_{11} = 1/3$. Hence, $Z_{3:4} = [12]$. ✓

3 We note that the probabilities for $B_{\cdot\mathrm{N}}$ are identical. Hence, the paths to $B_{1\mathrm{I}}$ only depend on the edge weights and the path along $Z_1 = 2$ is favoured, its weight is $\propto 9/25$ ✓.

4 We note that the probabilities for $B_{\cdot\mathrm{N}}$ are identical. Hence, the paths to $B_{2\mathrm{I}}$ only depend on the edge weights and the path along $Z_1 = 1$ is favoured, its weight is $\propto 4/15$ ✓.

5 Comparing the paths to $B_{1\mathrm{C}}$, we multiply the paths weights for $B_{1\mathrm{I}}$ and $B_{2\mathrm{I}}$ with $1/5 \times 1/3$ and $3/5 \times 3/5$, respectively. We get $9/375$ for $Z_{1:3} = [211]$ and $36/375$ for $Z_{1:3} = [121]$. Hence, $Z_{1:3} = [121]$ is more likely ✓ ✓.

We conclude that the most likely latent state sequence $\max_Z P(Z|X_{1:4} = [\mathrm{nice}]) = [1212]$.
This is just one possible solution, students could for example also apply the Viterbi algorithm. In general, please give one credit for justifying why $Z_3 = C$, one credit for justifying why $Z_4 = 2$ and four credits for justifying which of the four paths to $B_{1,c}$ is the most likely.
Enumerating all possible paths is also fine. 3✓ if omissions or minor mistakes. 6✓ if correct

Problem 6 Temporal Point Processes (10 credits)

0 1 2 3 4 5 6 7 8 9 10

Assume that we use a Hawkes process to model a discrete event sequence $\{t_1, \dots, t_N\}$ with $t_i \in [0, T]$. Further assume that (like in the lecture) we use an exponential triggering kernel, i.e. $k_\omega(t - t_i) = \exp(-\omega(t - t_i))$. Prove that the log-likelihood-function of the process is

$$\log p_\theta(\{t_1, \dots, t_N\}) = \sum_{i=1}^{N} \log \left(\mu + \alpha \sum_{j<i} \exp(-\omega(t_i - t_j)) \right) - \mu T + \frac{\alpha}{\omega} \sum_{i=1}^{N} (\exp(-\omega(T - t_i)) - 1)$$

The Hawkes process uses the conditional density

$$\lambda^*(t) = \mu + \alpha \sum_{t_j \in \mathcal{H}(t)} k_\omega(t - t_j), \checkmark \tag{6.1}$$

with $k_\omega(t - t_j) = \exp(-\omega(t - t_j))$. The log-likelihood of a TPP is

$$\log p_\theta(\{t_1, \dots, t_N\}) = \sum_{i=1}^{N} \log \lambda^*(t_i) - \int_0^T \lambda^*(u) \mathrm{d}u. \checkmark \tag{6.2}$$

We first insert the conditional density into the sum in Eq. 6.2:

$$\sum_{i=1}^{N} \log \lambda^*(t_i) = \sum_{i=1}^{N} \log \left(\mu + \alpha \sum_{j<i} \exp(-\omega(t_i - t_j)) \right). \checkmark \checkmark \tag{6.3}$$

The result is identical to the first summand of the log-likelihood specified in the problem statement. Next, we evaluate the integral. Since $\lambda^*(u)$ changes after each event, we have to split the integral into a sum of segments. Doing so, we obtain

$$\begin{aligned}
\int_0^T \lambda^*(u) \mathrm{d}u &= \int_0^{t_1} \mu \mathrm{d}u + \sum_{i=1}^{N} \int_{t_i}^{t_{i+1}} \mu + \alpha \sum_{j \le i} \exp(-\omega(t - t_j)) \mathrm{d}u \checkmark \checkmark \\
&= \mu T \checkmark + \alpha \sum_{i=1}^{N} \sum_{j \le i} \int_{t_i}^{t_{i+1}} \exp(-\omega(t - t_j)) \mathrm{d}u \\
&= \mu T - \frac{\alpha}{\omega} \sum_{i=1}^{N} \sum_{j \le i} \left(\exp(-\omega(t_{i+1} - t_j)) - \exp(-\omega(t_i - t_j)) \right) \checkmark \\
&\overset{(1)}{=} \mu T - \frac{\alpha}{\omega} \sum_{i=1}^{N} \left(\exp(-\omega(T - t_i)) - \exp(-\omega(t_i - t_i)) \right) \checkmark \\
&= \mu T - \frac{\alpha}{\omega} \sum_{i=1}^{N} (\exp(-\omega(T - t_i)) - 1) \checkmark
\end{aligned} \tag{6.4}$$

where we denote $t_{N+1} = T$ for convenience and in (1) we used that the double sum cancels out. Overall, we arrive at

The assignment of the last six credits above is to be read as follows:

- 2 credits for correctly splitting up the integral (splitting up the μ term is not necessary)
- 1 credit for correctly integrating the constant μ.
- 1 credit for correctly integrating the exponential function (mind the $\frac{1}{\omega}$ constant)
- 1 credit for the trick with the telescoping sums
- 1 credit for the final simplification

$$\log p_\theta(\{t_1, \dots, t_N\}) = \sum_{i=1}^{N} \log \left(\mu + \alpha \sum_{j<i} \exp(-\omega(t_i - t_j)) \right) - \mu T + \frac{\alpha}{\omega} \sum_{i=1}^{N} (\exp(-\omega(T - t_i)) - 1). \tag{6.5}$$

Problem 7 Graphs – Generative Models (8 credits)

Let $\boldsymbol{A} \in \{0,1\}^{N\times N}$ be the adjacency matrix of a graph generated by a stochastic block model with $\pi = \begin{bmatrix} a & 1-a \end{bmatrix}^T$, $\eta = \begin{bmatrix} p & q \\ q & p \end{bmatrix}$ and parameters $a, p, q \in [0,1]$. Let $\deg(n) = \sum_{j=1}^{N} A_{n,j}$ be the degree of node n.

Derive the expected degree $\mathbb{E}[\deg(n)]$ of an arbitrary node n.

0 1 2 3 4 5 6 7 8

In the following, we assume that we have no self-loops, i.e. $A_{n,n} = 0$ (but allowing self-loops is also fine). Without loss of generality assume that $n = N$. Due to linearity of expectation, we have

$$\mathbb{E}[\deg] = \sum_{j=1}^{N-1} \mathbb{E}\left[A_{N,j}\right] \checkmark\checkmark = \sum_{j=1}^{N-1} \Pr\left[A_{N,j} = 1\right],$$

where the last equality follows from the fact that $A_{N,j}$ is a binary random variable.
We can distinguish four possible community assignments for nodes N and j and apply the law of total probability:

$$\begin{aligned}
\Pr\left[A_{N,j} = 1\right] &= \Pr\left[A_{N,j} = 1 \mid z_N = 1 = z_j\right] \Pr\left[z_N = 1 = z_j\right] \\
&\quad + \Pr\left[A_{N,j} = 1 \mid z_N = 2 = z_j\right] \Pr\left[z_N = 2 = z_j\right] \\
&\quad + \Pr\left[A_{N,j} = 1 \mid z_N = 1 \wedge z_j = 2\right] \Pr\left[z_N = 1 \wedge z_j = 2\right] \\
&\quad + \Pr\left[A_{N,j} = 1 \mid z_N = 2 \wedge = z_j = 1\right] \Pr\left[z_N = 2 \wedge = z_j = 1\right] \\
&= p \cdot (a^2 + (1-a)^2) + q \cdot (2 \cdot a(1-a)). \checkmark
\end{aligned} \quad \checkmark\checkmark\checkmark$$

and thus

$$\mathbb{E}[\deg] = (N-1)\left(p \cdot (a^2 + (1-a)^2) + q \cdot (2 \cdot a(1-a)). \checkmark\right)$$

Please also give full credit if the students assume that we have self-loops, which will lead to a factor N instead of $N-1$.
Please account for follow-up mistakes, i.e. give students the last credit even if they plug an incorrect term for the inner expectation into the sum.
Two of the four credits from the "law of total probability equation" above can be given whenever the students get the high-level idea of making a case distinction based on the cluster assignments of pairs of nodes.
Solving via conditioanl expectations, i.e.

$$\mathbf{E}[\deg(N)] = a \cdot \mathbf{E}[\deg(N) \mid z_N = 1] + (1-a) \cdot \mathbf{E}[\deg(N) \mid z_N = 2] \tag{7.1}$$

is also fine. Same scheme as above applies.

Problem 8 Graphs – Clustering (10 credits)

Let $\boldsymbol{A} \in \{0,1\}^{N \times N}$ be the adjacency matrix of an undirected graph (i.e. symmetric adjacency matrix) generated by a stochastic block model with $\pi = \begin{bmatrix} a & 1-a \end{bmatrix}^T$, $\eta = \begin{bmatrix} p & q \\ q & p \end{bmatrix}$ and parameters $a, p, q \in [0,1]$.

0 1 2

a) Assume that $p, q, a \in [0,1]$ are known and fixed. Can $\Pr(\boldsymbol{z} \mid \boldsymbol{A}, \eta, \pi)$, the probability mass function of community assignments $\boldsymbol{z}$ given $\boldsymbol{A}$, be evaluated in polynomial time? That is, can it be evaluated in $\mathcal{O}(N^c)$, where N is the number of nodes and $c \in \mathbb{R}_+$? Justify your answer.

No. We have

$$\Pr(\boldsymbol{z} \mid \boldsymbol{A}, \eta, \pi) = \frac{\Pr(\boldsymbol{A} \mid \boldsymbol{z}, \eta) \Pr(\boldsymbol{z} \mid \pi)}{\Pr(\boldsymbol{A} \mid \eta, \pi)} = \frac{\Pr(\boldsymbol{A} \mid \boldsymbol{z}, \eta) \Pr(\boldsymbol{z} \mid \pi)}{\sum_{\boldsymbol{z}' \{0,1\}^N} \Pr(\boldsymbol{A} \mid \boldsymbol{z}', \eta) \Pr(\boldsymbol{z}' \mid \pi)}$$

and evaluating the denominator requires exponential time.
Explicitly writing out the posteiror is not necessary. Any argument along the lines of "calculating the normalizing constant takes exponential time because there are exponentially many possible community assignments" should get 2 credits.

0 1 2 3 4 5 6 7 8

b) Now assume that $p = 0$. Further assume that $\boldsymbol{A}$ is a connected graph (i.e. each pair of nodes (i, j) is connected by a path). Propose a procedure for finding the most likely community assignment, i.e.

$$\max_{\boldsymbol{z} \in \{0,1\}^N} \Pr(\boldsymbol{z} \mid \boldsymbol{A}, \eta, \pi)$$

in polynomial time $\mathcal{O}(N^c)$. Justify your answers.

Because $p = 0$, $\boldsymbol{A}$ must be a bipartite graph, i.e. $\mathbb{V} = \mathbb{H} \cup \mathbb{J}$.
Because $\boldsymbol{A}$ is connected, the partitions can be found in polynomial time, e.g. by starting depth-first-search at an arbitrary node, and adding nodes at even depth to $\mathbb{H}$ and at odd depth to $\mathbb{J}$.
Because $\boldsymbol{A}$ is connected and bipartite, there are only two possible community assignments: Either $\boldsymbol{z}_n = 1 \iff n \in \mathbb{H}$ or $\boldsymbol{z}_n = 1 \iff n \in \mathbb{J}$.
Define $\hat{\boldsymbol{z}} \in \{0,1\}^N$ with $\hat{\boldsymbol{z}}_n = 1 \iff n \in \mathbb{H}$.
Then

$$\Pr(\hat{\boldsymbol{z}} \mid \boldsymbol{A}, \eta, \pi) = \frac{\Pr(\boldsymbol{A} \mid \hat{\boldsymbol{z}}, \eta) \Pr(\hat{\boldsymbol{z}} \mid \pi)}{\Pr(\boldsymbol{A} \mid \hat{\boldsymbol{z}}, \eta) \Pr(\hat{\boldsymbol{z}} \mid \pi) + \Pr(\boldsymbol{A} \mid 1 - \hat{\boldsymbol{z}}, \eta) \Pr(1 - \hat{\boldsymbol{z}} \mid \pi)}$$

If the probability is ≥ 0.5, we have found our most likely community assignment. Otherwise, $1 - \hat{\boldsymbol{z}}$ is the most likely community assignment.
The above probability can be computed in polynomial time.
Students don't have to come up with the exact procedure above. Please grade as follows:

- Explaining that we have a bipartite graph (using this specific technical term is not necessary) ✓
- Explaining that there are thus only two possible community assignments ✓ ✓
- Some valid procedure for finding the two parts of the graph ✓ ✓
- Explanation why the procedure for finding the two parts runs in polynomial time ✓
- Finding maximum by testing which of the two possible assignments has higher likelihood ✓
- Some explanation why evaluating likelihood is in polynomial time (writing out the formula as above is not necessary) ✓

It is not necessary for the students to provide detailed pseudocode, a high-level explanation like in the sample solution above is sufficient.

Problem 9 Limitations of Graph Neural Networks (6 credits)

a) What is oversmoothing and what causes it?

0 1 2 3

Oversmoothing refers to the effect that in the limit of infinite layers (under some assumptions), the embedding vectors of each node (in a connected component) converge to the same constant vector and hence, the nodes will get indistinguishable w.r.t. their attributes.
1P for explaining what oversmoothing is. They don't have to write that it is only valid under some assumptions and only in the connected component. It would also be acceptable to explain that oversmoothing refers to the phenomenon that each node representation (in the limit of infinite layers) is only dependent on global graph properties and not the local neighbourhood anymore.

This is caused by the fact that in this limit, the influence of all nodes on an arbitrary node follows the PageRank solution and hence, the final representation is only dependent on the global graph structure and not on the local neighbourhood anymore.
2P for explaining what causes it. If the student argues that iteratively applying averaging operations makes nodes more and more similar causing oversmoothing (instead of argueing with the influence distribution) - also give that point. They do not have to explicitly write that the final representation only depends on the global graph structure (this is an implicit property from being proportional to the stationary distribution of the graph, however, as written above, could be used to explain what oversmoothing is.)
Note that there do exist multiple ways to explain what oversmoothing is and what causes it, a coherent logical argument diverging from the sample solution should of course yield the respective points.

b) The Personalized Propagation of Neural Predictions (PPNP) architecture is designed to overcome the problem of oversmoothing. Briefly explain its two key building blocks.

0 1 2 3

- First, it separates the feature transformation from the propagation along the graph structure. It does so by applying a learnable function on the features of each node, e.g. a neural network before the propagation step. 2 points
- Second, for the propagation step, it uses personalized pagerank applied on the transformed features of each node. This means for each node a personalized teleportation probability is introduced and only the particular node is in the teleportation set. 1 point: writing that personalized pagerank is applied on the transformed features is enough for that point

Problem 10 Page Rank (8 credits)

Recall the spam farm discussed in our exercise. It consists of the spammer's own pages S_{own} with target page t and k supporting pages, as well as links from the accessible pages S_{acc} to the target page. **Different from the exercise, every page within S_{own} has a link to every other page within S_{own}** (see Fig. 10.1).

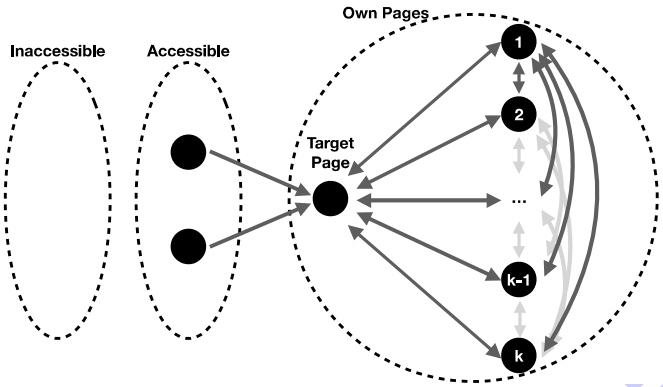


Figure 10.1

Let n be the total number of pages on the web, E the set of all edges, r_p the PageRank score of a page p and d_p the degree of a page p. Let $x_{\text{acc}} = \sum_{p \in S_{\text{acc}}, (p,t) \in E} \frac{r_p}{d_p}$ be the amount of PageRank contributed by the accessible pages. We are using PageRank with teleports, where $(1 - \beta)$ is the teleport probability.

0 1 2 3

a) Derive the PageRank r_s of a supporting page r_s as a function of β, r_t, k, n.

The PageRank score of a supporting page consists of the PageRank coming from the target page r_t and all supporting pages:

$$r_s = \beta \frac{r_t}{k} + \beta * \frac{1}{k} * \sum_{i=1}^{k-1} r_i + \frac{1-\beta}{n} \tag{10.1}$$

1p for correct formulation of PageRank equation.
We can recognize that the supporting pages are all symmetric, i.e. have the same edges. Thus, we can get rid of the sum with:

$$r_s = \beta \frac{r_t}{k} + \beta * \frac{k-1}{k} * r_s + \frac{1-\beta}{n} \tag{10.2}$$

1p for simplification, specifically, replacing the sum with $(k - 1)$.
Lastly, we solve for r_s:

$$r_s = r_t \frac{\beta}{k - \beta k + \beta} + \frac{k - \beta k}{n * (k - \beta k + \beta)} \tag{10.3}$$

1p for correct rewriting of formula w.r.t. r_s / correct result

b) Derive the PageRank r_t of the target page as a function of x_{acc}, k, β, n. You do not have to simplify.

0 1 2 3

The PageRank of a target page consists of the accessible pages as well as the PageRank from all supporting pages:

$$r_t = \beta x_{acc} + \beta \sum_{i=1}^{k} \frac{r_i}{k} + \frac{1-\beta}{n} \tag{10.4}$$

1p for correct formulation of PageRank equation. We can again simplify due to the fact that all supporting pages have the same PageRank and insert the solution from a) for r_s:

$$r_t = \beta x_{acc} + \frac{\beta^2}{k-\beta k+\beta} r_t + \beta \frac{k-\beta k}{n(k-\beta k+\beta)} + \frac{1-\beta}{n} \tag{10.5}$$

1p for replacing the sum and correct insertion of r_s. Lastly, we can rewrite w.r.t. r_t:

$$r_t = (\beta x_{acc} + \beta \frac{k-\beta k}{n(k-\beta k+\beta)} + \frac{1-\beta}{n}) * (1 - \frac{\beta^2}{k-\beta k+\beta})^{-1} \tag{10.6}$$

1p for correct rewriting of formula w.r.t. r_t / correct result
If a student mistakenly writes $r_t = x_{acc} + \beta \sum_{i=1}^{k} \frac{r_i}{k} + \frac{1-\beta}{n}$ (missing β) but the solution is otherwise correct, we give 2p in total (full points -1 for missing β) If the PageRank formulation from r_s from task a) is incorrect but the student otherwise takes the correct step we give full points. This applies only if the rest of the calculation is not significantly simplified by the wrong solution from the previous task.

c) How can the spammer modify the edges of the k supporting pages to increase the PageRank score r_t of the target page? Justify your answer.

0 1 2

Since currently, the is PageRank score is shared by the spam farm due to connections between the supporting pages, the spamer could increase the PageRank of the target page by decreasing the number of edges between supporting pages or completely removing edges between supporting pages. Full points for either removing or decreasing edges, students do not need to name both.

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