KIT-Fakultät für Informatik
Prof. Dr. G. Neumann
Prof. Dr. P. Friederich

Karlsruher Institut für Technologie

Aufgabenblätter

zur Klausur

„Grundlagen der Künstlichen Intelligenz mit Extra Leistungen“

am 7. März 2024, 10:30 – 12:30 Uhr

- Beschriften Sie bitte gleich zu Beginn jedes Lösungsblatt deutlich lesbar mit Ihrem Namen und Ihrer Matrikelnummer.
  Please mark your name and student ID number clearly legibly on each solution sheet immediately at the beginning of the exam
- Diese Aufgabenblätter werden nicht abgegeben. Tragen Sie Ihre Lösung deshalb ausschließlich in die für jede Aufgabe vorgesehenen Bereiche der Lösungsblätter ein.
  These exercises sheets will not be handed in. Therefore, please write your solution for each exercise exclusively in the designated areas of the solution sheets.
- Außer Schreibmaterial sind während der Klausur keine Hilfsmittel zugelassen. Täuschungsversuche durch Verwendung unzulässiger Hilfsmittel führen unmittelbar zum Ausschluss von der Klausur und zur Note „nicht bestanden“.
  Except for writing materials, no aids are allowed during the exam. Attempts to cheat by using unauthorized aids will result in immediate expulsion from the exam and the grade "failed".
- Die Gesamtpunktzahl beträgt 92 Punkte. Zum Bestehen der Klausur müssen mindestens 46 Punkte erreicht werden.
  The total number of points is 92. A minimum number of 46 points is necessary to pass the exam.
- Falls Sie den Bonus für die Klausur in einem früheren Semester (vor Wintersemester 23/24) erreicht haben, notieren Sie bitte das entsprechende Semester auf das Deckblatt der Lösungsblätter.
  If you collected bonus points in an earlier semester (before winter semester WS 23/24), please note the semester on the title page of the solution sheet.
- Verwenden Sie weder Rotstift noch Bleistift. Alle Lösungen in rot oder mit Bleistift werden nicht gewertet.
  Do not use neither pencil nor red color to write down your solutions. Solutions written in red color or with pencil will not be evaluated.

Viel Erfolg!

Good luck!

Aufgabe 1 *Allgemeine Fragen* (15 Punkte)
Question 1 *General Questions* (15 Points)

1. Das Gradientenabstiegsverfahren (‘gradient descent’) wird zur numerischen Lösung von Minimierungsproblemen eingesetzt.
   The gradient descent method is used to numerically solve minimization problems.

   (a) Beschreiben Sie das Verfahren und geben Sie ein Beispiel an, wofür es im Bereich des maschinellen Lernens verwendet wird! 2 P.
   Describe the method and give an example where it is used in the field of machine learning!

   (b) Was ist der Unterschied zwischen ‘batch gradient descent’ und ‘stochastic gradient descent’? Diskutieren Sie die Vor- und Nachteile der beiden Varianten! 3 P.
   What is the difference between 'batch gradient descent' and 'stochastic gradient descent'? Discuss the advantages and disadvantages of both variants!

2. Logik
   Logic

   (a) Erklären Sie den Unterschied zwischen Syntax und Semantik im Bereich der Logik. 2 P.
   Explain the difference between syntax and semantics in the field of logic.

   (b) Sie möchten wissen, ob Sie eine Prüfung bestehen, welche aus zwei Aufgaben besteht, die jeweils schwierig sein können oder nicht. Dazu kennen Sie die folgenden Aussagen und Formeln: 2 P.
   You want to know whether you will pass an exam consisting of two tasks, each of which may or may not be difficult. You know the following statements and premises:

   S_n : Aufgabe n ist schwierig.
   Task n is difficult
   L_n : Sie können Aufgabe n lösen
   You can solve task n
   B : Sie bestehen die Prüfung
   You pass the exam
   $\neg S_1 \to L_1, \quad S_2 \to \neg L_2, \quad L_1 \wedge L_2 \to B$

   Können Sie mit diesen Informationen für eine Prüfung mit zwei leichten Aufgaben eine Aussage darüber treffen, ob Sie die Prüfung bestehen? Wenn ja, welche und warum? Wenn nein, welche Information fehlt Ihnen dafür? Begründen Sie in beiden Fällen Ihre Antwort!
   With this information, can you conclude whether you will pass the exam if it consists of two easy tasks? If yes, which conclusion can you draw and why? If not, what information is missing to conclude? Justify your answer in both cases!

3. Beschreiben Sie kurz (in jeweils maximal drei Sätzen) die folgenden Methoden des unüberwachten Lernens und nennen Sie je ein Anwendungsgebiet! 6 P.
   Briefly describe (in a maximum of three sentences each) the following methods of unsupervised learning and name one area of application for each!

   (1) Clustering

   (2) Principal Component Analysis

   (3) Autoencoder

Aufgabe 2 *Klassifikation* (12 Punkte)
Question 2 *Classification* (12 Points)

Sie möchten ein Modell zur Klassifizierung von Obst entwickeln, wobei in Ihrem Datensatz jedes Stück Obst durch einen i-dimensionalen Vektor x dargestellt ist. Im ersten Fall beinhaltet Ihr Datensatz nur zwei Obstsorten mit den Klassenlabels $c_0 = 0$ (Äpfel) und $c_1 = 1$ (Birnen). Sie wissen, dass der Datasenatz linear separierbar ist und verwenden zunächst einen linearen Klassifikator der Form

$$f(\tilde{x}) = \tilde{w}^T \tilde{x} \quad ,$$

wobei Funktionswerte von $f(\tilde{x}) < 0.5$ der Klasse c_0 und Funktionswerte $f(\tilde{x}) \geq 0.5$ der Klasse c_1 zugeordnet werden. Dabei ist $\tilde{x}$ der um eine 1 erweiterte Inputvektor und $\tilde{w}$ der um den Biasterm b erweiterte Gewichtsvektor.

You want to develop a model for the classification of fruit and each fruit in your dataset is represented as an i-dimensional vector x. In the first case, your dataset only contains two types of fruit with class labels $c_0 = 0$ (apples) and $c_1 = 1$ (pears). You know that the dataset is linearly separable and start by using a linear classifier of the form

$$f(\tilde{x}) = \tilde{w}^T \tilde{x} \quad ,$$

where values of $f(x) < 0.5$ correspond to class c_0 and values of $f(x) \geq 0.5$ to class c_1 with $\tilde{x}$ being the input vector extended by 1 and $\tilde{w}$ the weight vector extended by the bias term b.

1. Erklären Sie, warum dieser lineare Klassifikator zusammen mit dem mittleren quadratischen Fehler ('mean square error', MSE) als Verlustfunktion nicht gut für Klassifikationen geeignet ist, und zeichnen Sie zur Veranschaulichung einen linear separierbaren Beispieldatensatz in 2D, bei dem die Datenpunkte zweier Klassen mit diesem Verfahren nicht getrennt werden können! 3 P.
   Explain why a linear classifier together with a 'mean square error' (MSE) loss function is not suited for classification tasks. As an illustration, draw an example of a linearly separable dataset in which the data points cannot be separated using this method.

2. Geben Sie die Funktionsgleichung der Sigmoid-Funktion an, und erklären Sie, warum diese besser als Klassifikator geeignet ist! Welche Verlustfunktion wird für Klassifikationen anstelle von MSE verwendet? Geben Sie den Namen und die Formel an! 3 P.
   Write down the equation of the sigmoid function and explain why it is better suited as a classifier. Which loss function is used for classifications instead of MSE? Give its name and the equation.

3. Im zweiten Fall beinhaltet Ihr Datensatz zusätzlich Bananen und Orangen. (Multiklassen-Klassifikation). Welche Funktion wird in diesem Fall als Klassifikator verwendet? Geben Sie den Namen und die Gleichung an! 2 P.
   In the second case, your dataset additionally contains bananas and oranges (multiclass classification). Which function is used as the classifier in this case? Give its name and the equation.

4. Wie würden Sie vorgehen, wenn die Klassen nicht linear separierbar wären? Zeichnen Sie einen Beispieldatensatz in 2D und erklären sie daran Ihr Vorgehen! 2 P.
   How would you proceed if the classes were not linearly separable? Draw an example dataset in 2D to explain your procedure.

5. Nachdem Sie Ihr Klassifikationsmodell trainiert haben, möchten Sie bewerten, wie gut es die korrekten Klassen vorhersagen kann. Erklären Sie, warum (und in welchen Fällen) die Genauigkeit als Metrik dafür nicht ausreicht! 2 P.

   After you have trained your classification model, you want to evaluate how well it can predict the correct classes. Explain why (and in which cases) accuracy is not a sufficient metric.

Aufgabe 3 *Hyperparameter und Ensembles* (12 Punkte)
Question 3 *Hyperparameters and ensembles* (12 Points)

Luisa möchte ein feed-forward neural network für ein Regressionsproblem verwenden, wobei Werte $x \in \mathbb{R}$ vorhergesagt werden sollen.
Luisa wants to use a feed-forward neural network for a regression problem, where values $x \in \mathbb{R}$ are to be predicted.

1. Welche Aktivierungsfunktion sollte sie für das Ausgabeneuron des neuronalen Netzes verwenden? Begründen Sie Ihre Antwort kurz. 2 P.
   Which activation function should she use for the output neuron of the neural network? Briefly justify your answer.

2. Luisa hat einen Datensatz für ihr Problem im Internet gefunden und teilt diesen zufällig in drei gleich große Teile: P1, P2 und P3. Sie beschließt, ihr neuronales Netz auf P1 zu trainieren und auf P3 den mittleren absoluten Fehler zu bestimmen. Nun ist sie sich nicht sicher, welchen der drei Teile sie zum Optimieren der Hyperparameter verwenden soll.
   Luisa found a dataset for her problem on the internet and randomly divides it into three equally sized parts: P1, P2, and P3. She decides to train her neural network on P1 and to determine the mean absolute error on P3. Now she is not sure which of the three parts she should use to optimize the hyperparameters.

   (a) Wir betrachten die drei Szenarien, in denen Luisa jeweils P1, P2 oder P3 zum Optimieren der Hyperparameter (Anzahl und Größe der hidden Layer) wählt. Ordnen Sie die drei Szenarien (P1, P2 und P3) nach dem zu erwartenden mittleren absoluten Fehler (gemessen auf P3) der Reihe nach aufsteigend. 2 P.
   We consider the three scenarios in which Luisa chooses P1, P2, or P3, respectively, to optimize the hyperparameters (number and size of hidden layers). Arrange the three scenarios (P1, P2, and P3) with ascending expected mean absolute error (measured on P3).

   (b) Welches der Szenarien sollte Luisa wählen? Begründen Sie Ihre Antwort. 2 P.
   Which of the scenarios should Luisa choose? Justify your answer.

3. Luisa beschließt, dass sie nicht nur daran interessiert ist, genaue Vorhersagen zu machen, sondern dass sie auch wissen möchte, wie verlässlich die Vorhersage ist. Dazu verwendet sie ein Ensemble mit $m = 5$ neuronalen Netzen, jedes davon macht eine Vorhersage für ihr Regressionsproblem. Den Mittelwert der Ausgaben des Ensembles verwendet sie als Vorhersage, die Standardabweichung des Ensembles verwendet sie als Unsicherheitsabschätzung der Vorhersage.
   Luisa decides that she is not only interested in making accurate predictions but also wants to know how reliable the prediction is. For this purpose, she uses an ensemble of $m = 5$ neural networks, each making a prediction for her regression problem. She uses the average of the outputs of the ensemble as the prediction, and the standard deviation of the ensemble as an uncertainty estimate of the prediction.

   (a) Erklären Sie die Bagging Strategie und begründen Sie, warum es sinnvoll ist, diese beim Trainieren eines Ensembles anzuwenden. Was ist der Vorteil gegenüber 2 P.

dem Szenario, in dem alle Modelle auf dem gesamten Datensatz trainiert werden?

Explain the bagging strategy and justify why it is sensible to apply it when training an ensemble. What is the advantage over the scenario in which all models are trained on the entire dataset?

(b) Luisa möchte die Varianz der Vorhersage ihres Ensembles abschätzen. $\hat{f}_i$ sei die Einzelvorhersage des Modells i mit $\mathbb{E}\left[\hat{f}_i\right] = \mu$ und $\mathrm{Var}\left[\hat{f}_i\right] = \sigma^2$. Luisa stellt die folgende Rechnung auf: 2 P.

Luisa wants to estimate the variance of the prediction of her ensemble. Let $\hat{f}_i$ be the prediction of model i with $\mathbb{E}\left[\hat{f}_i\right] = \mu$ and $\mathrm{Var}\left[\hat{f}_i\right] = \sigma^2$. Luisa calculates the Variance in the following way:

$$\mathrm{Var}\left[\hat{f}_{\text{ens}}\right] = \mathrm{Var}\left[\frac{1}{m}\sum_{i=1}^{m}\hat{f}_i\right] = \frac{1}{m^2}\sum_{i=1}^{m}\mathrm{Var}\left[\hat{f}_i\right] = \frac{1}{m}\sigma^2 \tag{1}$$

Entscheiden Sie, ob diese Rechnung für das Trainieren des Ensembles mit der Bagging Strategie richtig ist. Begründen Sie Ihre Antwort.

Decide if her calculation is correct when using the bagging strategy to train the ensemble. Justify your answer.

(c) Luisa überlegt, ob sie für jedes Modell im Ensemble eine andere Modell-Architektur (Anzahl und Größe der hidden Layer) verwenden kann. Sie ist sich nicht sicher, ob das die Ensemble-Vorhersage verfälschen könnte. Begründen Sie, ob bzw. warum das Benutzen unterschiedlicher Modell-Architekturen im Ensemble sinnvoll sein könnte. 2 P.

Luisa is considering whether she can use a different model architecture (number and size of the hidden layers) for each model in the ensemble. She is unsure whether this could distort the ensemble prediction. Justify whether or why using different model architectures in the ensemble could make sense.

Aufgabe 4 *Suchprobleme* (12 Punkte)
Question 4 *Search* (12 Points)

Wir betrachten folgendes Suchproblem: Gegeben seien n übereinander liegende Pfannkuchen mit Durchmesser 1 bis n auf einem Stapel. Ziel ist es, die Pfannkuchen der Größe nach zu sortieren. Wir können jeden Zustand mit einem Vektor aller Durchmesser beschreiben, der Zielzustand wäre beispielsweise

We consider the following search problem: Given are n stacked pancakes with diameters 1 through n. The goal is to sort the pancakes by size. We can describe each state with a vector of all diameters; for example, the target state would be

$$\text{Goal} = \begin{bmatrix} 1 \\ 2 \\ \vdots \\ n \end{bmatrix}.$$

Wir können die Pfannkuchen manipulieren, indem wir mit einem Pfannenwender zwischen zwei beliebigen Pfannkuchen stechen (bzw. unter alle Pfannkuchen gehen) und den aufgegabelten oberen Teil umdrehen. Somit haben wir im oberen Teil die **umgedrehte Reihenfolge** der Pfannkuchen. Die Kosten für diesen Übergang entspricht der **Anzahl umgedrehten Pfannkuchen** (siehe folgende Abbildung).

We can manipulate the pancakes by inserting a spatula between any two pancakes (or going under all pancakes) and flipping the upper part that has been forked. Thus, in the upper part, we have the **reversed order** of the pancakes. The cost for this transition corresponds to the **number of flipped pancakes** (see the following figure).

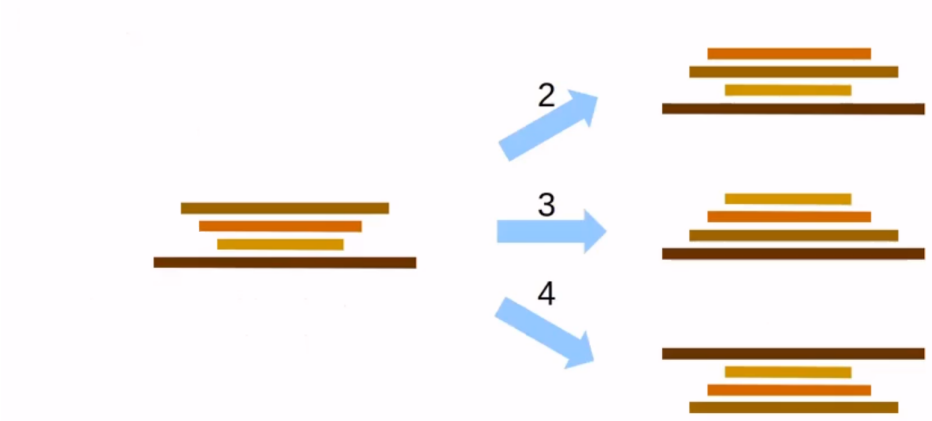


Abbildung: Visuelle Darstellung des Zustandsübergangs und dessen Kosten im Pfannkuchenproblem.

Figure: Visual representation of the state transition and its costs in the pancake sorting problem.

1. Wie ist ein Suchproblem im Allgemeinen definiert? Geben Sie die 3 Bausteine an. 2 P.
   How is a search problem generally defined? Provide the 3 components.
2. Wie viele Zustände hat das Pfannkuchenproblem mit $n = 4$? 1 P.
   How many states does the pancake problem with $n = 4$ have?

3. Erklären Sie den „Curse of Dimensionality“ im Kontext von Suchproblemen anhand des Pfannkuchenproblems. 1 P.
   Explain the "Curse of Dimensionality" in the context of search problems using the pancake problem as an example.

Wir definieren nun eine Heuristik h: Für einen Zustand s sei $h(s)$ der Durchmesser des größten Pfannkuchens, der noch falsch sortiert ist. In der obigen Abbildung hätte der linke Ausgangszustand also eine Heuristik von $h(s) = 3$, da der Pfannkuchen mit Durchmesser 4 schon an der richtigen Position ist.
We now define a heuristic h: For a state s, let $h(s)$ be the diameter of the largest pancake that is still incorrectly sorted. In the above illustration, the initial state on the left would have a heuristic of $h(s) = 3$, since the pancake with diameter 4 is already in the correct position.

4. Ist die Heuristik zulässig? Begründen Sie oder geben Sie ein Gegenbeispiel an. 2 P.
   Is the heuristic admissible? Justify or provide a counterexample.

5. Sei nun $n = 4$. Führen Sie eine A^*-Suche mit der obigen Heuristik durch. Zeichnen Sie dazu den Suchbaum in das vorgegebene Skelett ein. Starten Sie mit dem Zustand 6 P.
   Now, let $n = 4$. Perform an A^* search using the above heuristic. To do this, draw the search tree in the given skeleton. Start with the state

$$\text{Start} = \begin{bmatrix} 4 \\ 3 \\ 1 \\ 2 \end{bmatrix}.$$

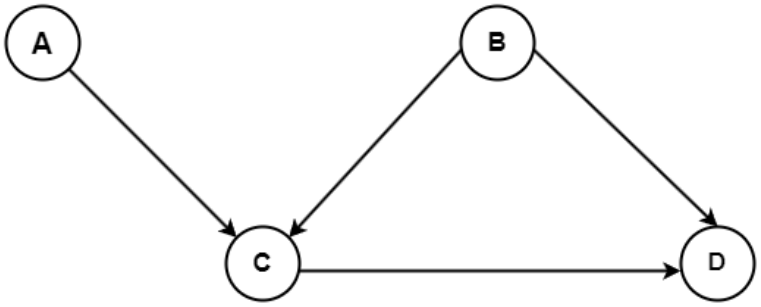
   Explorieren Sie nicht erneut den Vorgänger des aktuellen Zustands. **Markieren Sie die Reihenfolge** der explorierten Zustände in den kreisförmigen Feldern. Geben Sie außerdem in jedem Zustand die **aktuellen Forward- und Backward-Kosten** an (h bzw. g). Falls 2 Zustände die **gleichen Gesamtkosten** haben, explorieren Sie zunächst den Zustand mit niedrigeren Backward Kosten (g-Kosten). Der Algorithmus terminiert, wenn der **Zielzustand zum Explorieren ausgewählt** wird.
   Do not re-explore the predecessor of the current state. **Mark the order** of the explored states in the circular fields. Also enter the **current forward and backward costs** in each state (h and g respectively). If 2 states have the **same total cost**, first explore the state with lower backward cost (g cost). The algorithm terminates when the **target state is selected for exploration**.

   Hinweis: Es müssen nicht alle vorgezeichneten Felder für die Zustände ausgefüllt werden.
   Note: It is not necessary to fill in all of the predefined fields for the states.

Aufgabe 5 *Bayesian Networks und MDPs* (12 Punkte)
Question 5 *Bayesian Networks and MDP* (12 Points)

Betrachten Sie das folgende Bayesian Network, in dem die Variablen A bis D alle boolesche Werte haben. Benutzen Sie für die Aufgabe 1. und 2. Formeln. Sie müssen das Endergebnis nicht ausrechnen, es genügt, die Herleitung anzugeben sowie die richtigen Zahlen aus der Tabelle einzusetzen.

Consider the following Bayesian Network, where variables A–D are all Boolean valued. Use formulas for tasks 1. and 2. You don't need to calculate the final result; it is sufficient to provide the derivation and insert the correct numbers from the table.



A	**P**(**A**)
false	0.8
true	0.2

B	**P**(**B**)
false	0.3
true	0.7

A	**B**	**P**(**C** = true | **A**, **B**)
false	false	0.2
false	true	0.3
true	false	0.1
true	true	0.6

B	**C**	**P**(**D** = true | **B**, **C**)
false	false	0.9
false	true	0.8
true	false	0.4
true	true	0.3

1. Was ist die Wahrscheinlichkeit, dass alle vier dieser booleschen Variablen falsch sind? 3 P.
   What is the probability that all four of these Boolean variables are false?

2. Was ist die Wahrscheinlichkeit, dass A wahr, C wahr und D falsch ist? 4 P.
   What is the probability that A is true, C is true and D is false?

Betrachten Sie nun einen Roboter, der sich in einer Umgebung bewegt. Das Ziel des Roboters ist es, von einem Startpunkt zu einem Zielpunkt so schnell wie möglich zu gelangen. Der Roboter hat jedoch die Einschränkung, dass sein Motor überhitzen und den Roboter am Weiterfahren hindern kann. Der Roboter kann sich mit zwei verschiedenen Geschwindigkeiten bewegen: langsam (**slow**) und schnell (**fast**). Wenn er sich schnell (**fast**) bewegt, erhält er einen Reward von 10; wenn er sich langsam (**slow**) bewegt, erhält er einen Reward von 4. Wir können dieses Problem als MDP modellieren, indem wir drei Zustände haben: kühl (**cool**), warm (**warm**) und kaputt (**broken**). Die Übergänge sind unten aufgeführt. Nehmen Sie an, dass der Discount Faktor $0,9$ beträgt, und gehen Sie auch davon aus,

dass wir beim Erreichen des Zustands kaputt (**broken**) dort bleiben und keine weiteren Belohnung erhalten.

Consider now a robot that is moving in an environment. The goal of the robot is to move from an initial point to a destination point as fast as possible. However, the robot has the limitation that if it moves fast, its engine can overheat and stop the robot from moving. The robot can move at two different speeds: **slow** and **fast**. If it moves **fast**, it gets a reward of 10; if it moves slowly, it gets a reward of 4. We can model this problem as an MDP by having three states: **cool**, **warm**, and **broken**. The transitions are shown below. Assume that the discount factor is 0.9 and also assume that when we reach the state **broken**, we remain there without getting any reward.

s	a	s'	$P(s' \mid a, s)$
cool	slow	cool	1
cool	fast	cool	1/4
cool	fast	warm	3/4
warm	slow	cool	1/2
warm	slow	warm	1/2
warm	fast	warm	7/8
warm	fast	broken	1/8

3. Verwenden Sie die oben gegebenen Informationen, um das MDP zu zeichnen, das die Zustände, Übergänge und Belohnungen des Roboters beschreibt (ähnlich dem Beispiel mit dem Autorennen aus der Vorlesung). Zeichnen Sie jeden Zustand (**cool**, **warm**, **broken**) und verbinden Sie sie mit den Aktivitäten (**slow**, **fast**), um die Übergänge auszudrücken. Notieren Sie auch die zugehörigen Belohnungen (Sie müssen die Übergangswahrscheinlichkeiten nicht erwähnen, da sie bereits in der Tabelle angegeben sind). 3 P.

   Using the information given above, draw the MDP which describes the states, transitions and rewards of the robot (similar to the racing example from the lecture). Draw each state (**cool**, **warm**, **broken**) and connect them to the activities (**slow**, **fast**) to express the transitions. Also, note down the associated rewards (you don't need to mention the transition probabilities as they are already given in the table).

4. Geben Sie die optimale Policy für jeden Zustand an. (Keine Begründung erforderlich.) 2 P.

   Provide the optimal policy for each state. (No explanation needed.)

Aufgabe 6 *Gaussian Mixture Models* (12 Punkte)

Question 6 *Gaussian Mixture Models* (12 Points)

Sie möchten einen Datensatz $\mathcal{D} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$ mit $\boldsymbol{x}_i \in \mathbb{R}^D$ mit einem Gaussian Mixture Model (GMM) der Form

$$p(\boldsymbol{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{x} \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

modellieren. Hierbei sind $\boldsymbol{\mu}_k \in \mathbb{R}^D, \boldsymbol{\Sigma}_k \in \mathbb{R}^{D \times D}$, und es gilt $0 \leq \pi_k \leq 1$, sowie $\sum_k \pi_k = 1$.
You want to model a dataset $\mathcal{D} = \boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$ with $\boldsymbol{x}_i \in \mathbb{R}^D$ using a Gaussian Mixture Model (GMM) of the form

$$p(\boldsymbol{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{x} \mid \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

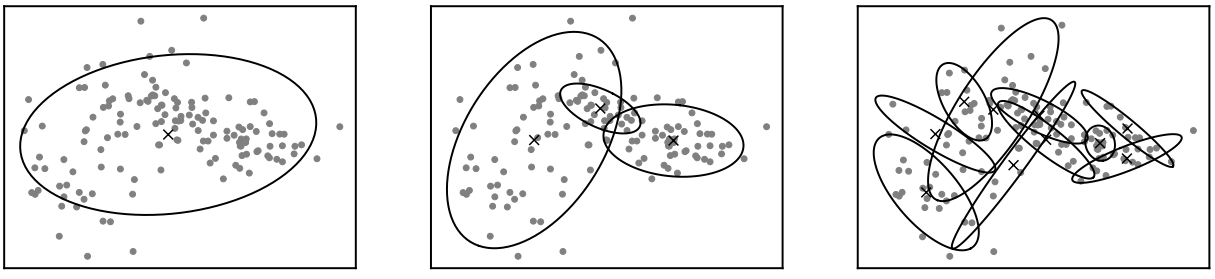
where $\boldsymbol{\mu}_k \in \mathbb{R}^D, \boldsymbol{\Sigma}_k \in \mathbb{R}^{D \times D}$, and $0 \leq \pi_k \leq 1$, with $\sum_k \pi_k = 1$.

1. Geben Sie die Formel der (Marginal) Log-Likelihood des Datensatz $\mathcal{D}$ unter dem Modell $p(\boldsymbol{x})$ an. Nehmen Sie hierbei an, dass die Datenpunkte stochastisch unabhängig sind. [1 P.]
   Provide the formula for the (marginal) log-likelihood of the dataset $\mathcal{D}$ under the model $p(\boldsymbol{x})$. Assume that the data points are stochastically independent.

2. Warum ist die direkte Optimierung der Log-Likelihood schwierig? Nennen Sie ein konkretes Problem! [1 P.]
   Why is direct optimization of the log-likelihood difficult? Provide a specific problem!

3. **Beschreiben und benennen** Sie die 2 Schritte des EM-Algorithmus. Geben Sie insbesondere an, welche Größen in jedem Schritt berechnet werden (keine Formeln verlangt). [2 P.]
   Describe and name the 2 steps of the EM algorithm. Specify which quantities are computed in each step (no formulas required).

Sie experimentieren mit dem EM Algorithmus für GMMs auf einem 2-dimensionalen Datensatz und erhalten folgende Ergebnisse mit jeweils 1, 3, und 10 Mixture Komponenten:
You experiment with the EM algorithm for GMMs on a 2-dimensional dataset and obtain the following results with 1, 3, and 10 mixture components:



(a) $K = 1$ (b) $K = 3$ (c) $K = 10$

4. Welches Modell hat auf dem gegebenen Datensatz die größte Log-Likelihood? Welches Modell würden Sie im Allgemeinen wählen? Begründen Sie. [2 P.]
   Which model has the highest log-likelihood on the given dataset? Which model would you generally choose? Justify your choice.

Gegeben sei nun der eindimensionale Datensatz $\mathcal{D} = \{-1, 1, 5\}$, den Sie mit einem GMM mit 2 Komponenten modellieren möchten. Sie initialieren das GMM wie folgt:
Given is now the one-dimensional dataset $\mathcal{D} = -1, 1, 5$, which you want to model with a GMM with 2 components. You initialize the GMM as follows:

$$p(x) = \frac{1}{2}\mathcal{N}(x \mid \mu_1 = 0, \Sigma_1 = [2]) + \frac{1}{2}\mathcal{N}(x \mid \mu_2 = 2, \Sigma_2 = [1]).$$

5. Führen Sie eine Iteration des EM-Algorithmus durch und **bestimmen Sie damit den neuen Mean** μ_1. Sie müssen im E-Step nur die dafür notwendigen Größen berechnen. Insbesondere ist es **nicht** verlangt, alle anderen neuen Parameter des GMMs bis auf μ_1 zu bestimmen! 6 P.
   Perform one iteration of the EM algorithm and **determine the new mean** μ_1. In the E-step, you only need to calculate the necessary quantities for this. Specifically, it is **not** required to determine all other new parameters of the GMM except for μ_1!

Hinweise:
Hints:

- Nutzen Sie folgende Tabelle für die (gerundeten) Berechnungen der Gaussian Likelihoods:
  Use the following table for the (rounded) calculations of the Gaussian likelihoods:

	$x = -1$	$x = 1$	$x = 5$
$\mathcal{N}(x\|\mu_1 = 0, \Sigma_1 = [2])$	$\frac{25}{100}$	$\frac{25}{100}$	0
$\mathcal{N}(x\|\mu_2 = 2, \Sigma_2 = [1])$	$\frac{5}{100}$	$\frac{35}{100}$	$\frac{5}{100}$

- Hier sind einige Terme, die für die Berechnung hilfreich sein könnten. (Nicht alle werden für die Berechnung benötigt!)
  Here are some terms that could be helpful for the calculation. (Not all will be needed for the computation!)

$$\frac{\sum_i q_{ik}(\boldsymbol{x}_i - \mu_k)(\boldsymbol{x}_i - \mu_k)^T}{\sum_i q_{ik}} \qquad \frac{\sum_i q_{ik}\boldsymbol{x}_i}{\sum_i q_{ik}} \qquad \frac{\pi_k \mathcal{N}(\boldsymbol{x}_i|\mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(\boldsymbol{x}_i|\mu_j, \Sigma_j)} \qquad \frac{\sum_i q_{ik}}{N}$$

- Benutzen Sie Brüche. Der finale Bruch von μ_1 muss nicht gekürzt werden, allerdings sollen alle Doppelbrüche und andere Operationen aufgelöst sein.
  Use fractions. The final fraction for μ_1 does not need to be simplified, but all compound fractions and other operations should be resolved.

Aufgabe 7 *Rekurrente neuronale Netze* (8 Punkte)
Question 7 *Recurrent neural networks* (8 Points)

Gerda und Peter haben einen neuen revolutionären Staubsauger entwickelt, den sie über ihre Webseite vertreiben. Da sie auf ihrer Webseite in den letzten Wochen gehäuft negative Rezensionen erhalten haben, beschließen sie das Problem an der Wurzel zu packen und zu lösen. Dazu möchten sie ein rekurrentes neuronales Netz (RNN) verwenden, welches negative Bewertungen erkennt, sodass diese automatisiert gelöscht werden können. Das RNN verwendet die folgende Updatefunktion, um die hidden representation im nächsten Zeitschritt t zu bestimmen:

Gerda and Peter have developed a new revolutionary vacuum cleaner, which they distribute via their website. Since they have received an increased number of negative reviews on their website in the last few weeks, they decide to tackle and solve the problem at its root. For this purpose, they want to use a recurrent neural network (RNN) that recognizes negative reviews, so that these can be automatically deleted. The RNN uses the following update function to determine the hidden representation in the next time step t:

$$h_t = \sigma\left(W^H h_{t-1} + W^X x_t + b\right) \tag{2}$$

mit

- h_t, h_{t-1}: Hidden representations, Dimension $n_h \times 1$.
- W^H: Gewichtmatrix mit Dimension $n_h \times n_h$.
- W^X: Gewichtmatrix mit Dimension $n_h \times n_x$.
- x_t: Vordefinierte Vektor-Darstellung des Wortes an der Stelle t, Dimension $n_x \times 1$.
- b: Bias Vektor, Dimension $n_h \times 1$.
- σ: Sigmoid Aktivierungsfunktion.

with

- h_t, h_{t-1}: Hidden representations, dimension $n_h \times 1$.
- W^H: Weight matrix with dimension $n_h \times n_h$.
- W^X: Weight matrix with dimension $n_h \times n_x$.
- x_t: Predefined vector representation of the word at position t, dimension $n_x \times 1$.
- b: Bias vector, dimension $n_h \times 1$.
- σ: Sigmoid activation function.

.

In jedem Zeitschritt wird eine Ausgabe des RNNs bestimmt:

In each time step, an output of the RNN is determined:

$$y_t = \sigma(W^Y h_t + c) \tag{3}$$

mit

- W^Y: Gewichtmatrix mit Dimension $n_y \times n_h$.
- c: Bias vector, Dimension $n_y \times 1$.

with

- W^Y: Weight matrix with dimension $n_y \times n_h$.
- c: Bias vector, dimension $n_y \times 1$.

.

1. Zeichnen Sie den unfolded computational graph des RNN für Sätze der Länge 4 mit einem Output y_t in jedem Zeitschritt. 2 P.
   Draw the unfolded computational graph of the RNN for sentences of length 4 with an output y_t at each time step.
2. Wie viele Parameter hat dieses RNN (inkl. Bias Parameter)? 2 P.
   How many parameters does this RNN have (incl. bias parameters)?
3. Um die binäre Klassifikation (positive oder negative Bewertung) durchzuführen müssen die Outputs des RNNs über die verschiedenen Zeitschritte aggregiert werden. Beschreiben Sie zwei unterschiedliche Aggregationsmöglichkeiten. Nach der Aggregierung verwenden wir ein feed-forward neural network für die Klassifikation. Welche Aktivierungsfunktion sollten die beiden für das Ausgabeneuron dieses feed-forward neural networks nutzen? Begründen Sie Ihre Antwort. 3 P.
   To perform the binary classification (positive or negative review), the outputs of the RNN over the different time steps must be aggregated. Describe two different aggregation methods. After aggregation, we use a feed-forward neural network for the classification. Which activation function should the two use for the output neuron of this feed-forward neural network? Justify your answer.
4. Was ist beim Klassifizieren von Rezensionen der entscheidende Vorteil eines RNN gegenüber feedforward neural networks oder convolutional neural networks? 1 P.
   What is the decisive advantage of an RNN in classifying reviews compared to feed-forward neural networks or convolutional neural networks?

Aufgabe 8 *Reinforcement Learning* (9 Punkte)

1. Erklären Sie das Ziel eines Reinforcement Learning Agenten. (Keine Formeln verlangt) 1 P.
   Explain the goal of a Reinforcement Learning agent (no formulas required).

Betrachten Sie die unten dargestellte Gridworld in der Pacman versucht, die optimale Policy zu erlernen. Wenn eine Aktion dazu führt, dass er in einen der schattierten Zustände gelangt, wird der entsprechende Reward während dieses Übergangs vergeben. Alle schattierten Zustände sind Endzustände, d. h. das MDP wird beendet, sobald er in einem schattierten Zustand angekommen ist. In den anderen Zuständen stehen die Aktionen “North (N)”, “East (E)”, “South (S)” und “West (W)” zur Verfügung, die Pacman deterministisch in den entsprechenden Nachbarzustand bewegt. Pacman bleibt an Ort und Stelle, wenn die Aktion versucht, sich aus dem Raster zu bewegen. Nehmen Sie für alle Berechnungen den Discount Faktor $\gamma = 0,5$ und die Q-Lerning Rate $\alpha = 0,5$ an. Pacman startet im State $(1, 3)$. Beachten Sie, dass State (a, b) die a-te Spalte und die b-te Zeile im Raster bedeutet.

Consider the grid world given below and Pacman who is trying to learn the optimal policy. If an action results in landing into one of the shaded states, the corresponding reward is awarded during that transition. All shaded states are terminal states, that is, the MDP terminates once it has arrived in a shaded state. The other states have the North (N), East (E), South (S), and West (W) actions available, which deterministically move Pacman to the corresponding neighbouring state (or have Pacman stays in place if the action tries to move out of the grid). Assume the discount factor $\gamma = 0.5$ and the Q learning rate $\alpha = 0.5$ for all calculations. Pacman starts in state (1, 3). Note that state (a,b) means the a-th column and b-th row in the grid.

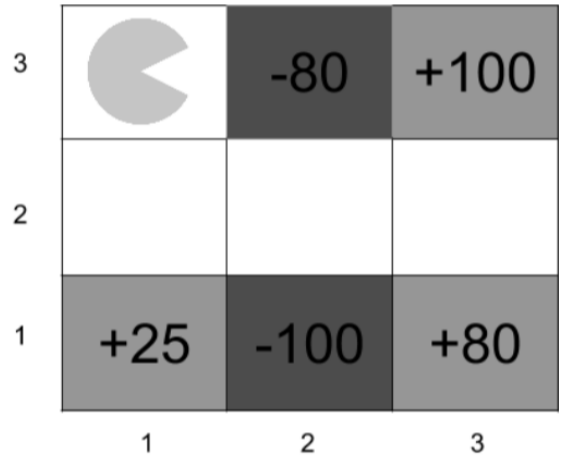


Abbildung: Gridworld

Nehmen Sie nun an, der Agent hat keine Informationen über die Gitterwelt und entscheidet sich, über sie Erfahrungen zu sammeln. Der Agent startet dabei immer in der linken oberen Ecke. Ihnen liegen die folgenden Episoden des Agenten durch diese Gitterwelt vor. Jede Zeile in einer Episode ist ein Tupel, das (state, action, next state, reward) enthält.
Assume the agent does not have any information about the grid-world and decides to collect them via experience. The agent starts from the top left corner and you are given the following episodes from runs of the agent through this grid-world. Each line in an episode is a tuple containing (s, a, s', r).

Episode 1	Episode 2	Episode 3
$(1,3), \mathrm{S}, (1,2), 0$	$(1,3), \mathrm{S}, (1,2), 0$	$(1,3), \mathrm{S}, (1,2), 0$
$(1,2), \mathrm{E}, (2,2), 0$	$(1,2), \mathrm{E}, (2,2), 0$	$(1,2), \mathrm{E}, (2,2), 0$
$(2,2), \mathrm{S}, (2,1), -100$	$(2,2), \mathrm{E}, (3,2), 0$	$(2,2), \mathrm{E}, (3,2), 0$
	$(3,2), \mathrm{N}, (3,3), +100$	$(3,2), \mathrm{S}, (3,1), +80$

2. Sie verwenden den Tabular Q-Learning-Algorithmus mit der aus der Vorlesung bekannten Regel für die Aktualisierung des Q-Werts: 2 P.

$$Q(\boldsymbol{s}, \boldsymbol{a}) \leftarrow Q(\boldsymbol{s}, \boldsymbol{a}) + \alpha\delta$$

wobei α die Lernrate und δ der temporal difference error sind. Geben Sie die Formel für den temporal difference error δ an, wenn s'

(a) ein terminaler Zustand ist

(b) ein nicht-terminaler Zustand ist.

Consider a Q-learning algorithm with the following Q-value update rule:

$$Q(\boldsymbol{s}, \boldsymbol{a}) \leftarrow Q(\boldsymbol{s}, \boldsymbol{a}) + \alpha\delta$$

where α is the learning rate and δ is the temporal difference error. Give the formula for the temporal difference error δ if s' is

(a) a terminal state

(b) a non-terminal state.

3. Unter Verwendung der Q-Learning-Aktualisierungen, berechnen Sie die Q-Werte nach den oben genannten drei Episoden für 6 P.

$$Q((3,2), N),$$
$$Q((1,2), S), \text{ und}$$
$$Q((2,2), E).$$

Hinweis: Sie müssen nicht alle gegebenen Übergänge verwenden, um zu den benötigten Q-Werten zu gelangen. Es wird angenommen, dass die anfänglichen Q-Werte für alle Zustände Null sind.
Using Q-Learning updates, write the Q-values after the above three episodes for

$$Q((3,2), N),$$
$$Q((1,2), S), \text{ and}$$
$$Q((2,2), E).$$

Hint: You don't have to use all given transitions to arrive at the answers for the needed Q values. The initial Q values for all states are assumed to be zero.

KIT-Fakultät für Informatik
Prof. Dr. G. Neumann
Prof. Dr. P. Friederich

Karlsruher Institut für Technologie

Musterlösungen

zur Klausur
„Grundlagen der Künstlichen Intelligenz mit Extra Leistungen“
am 7. März 2024

Name:	Vorname:	Matrikelnummer:

Aufgabe 1	15 von 15 Punkten
Aufgabe 2	12 von 12 Punkten
Aufgabe 3	12 von 12 Punkten
Aufgabe 4	12 von 12 Punkten
Aufgabe 5	12 von 12 Punkten
Aufgabe 6	12 von 12 Punkten
Aufgabe 7	8 von 8 Punkten
Aufgabe 8	9 von 9 Punkten
Gesamtpunktzahl:	92 von 92 Punkten

Note:	**1.0**

Aufgabe 1 *Allgemeine Fragen*

Solution 1 *General Questions*

1. (a) Beim Gradientenverfahren wird ein Extrempunkt einer Funktion gesucht, in dem mit einem Satz von Startparametern θ_0 begonnen wird und der Gradient der Funktion $f(\theta)$ berechnet wird. Danach wird der Gradient mit einer Schrittweite („learning rate“: η) multipliziert und von den Startparametern subtrahiert (für eine Minimumsuche) oder addiert (Maximumsuche). Dies wird so oft wiederholt, bis sich der Funktionswert nicht mehr oder nur noch unterhalb eines Grenzwertes ändert:

$$\theta_{t+1} = \theta_t - \eta \nabla_\theta f(\theta)$$

Anwendung: Optimierung von Modellparametern

(b) Bei der Variante „batch gradient descent“ wird der Mittelwert der Ableitungen aller Datenpunkte verwendet, während bei „stochastic gradient descent“ immer nur ein (zufälliger) Datenpunkt verwendet wird.
Vor- und Nachteile: Bei „batch gradient descent“ werden weniger Optimierungsschritte benötigt, und das Verfahren konvergiert zuverlässiger zum Optimum, da der nächste Punkt bei „stochastic gradient descent“ aufgrund des Einflusses der anderen Parameter nicht immer besser sein muss. Andererseits ist jeder Optimierungsschritt hier weniger aufwendig, so dass diese Variante insgesamt weniger Rechenzeit braucht.

2. (a) **Syntax:** Beschreibung gültiger Symbole, Ausdrücke und Formeln der Logik.
Semantik: Bedeutung bzw. Interpretation der Aussagen

(b) Aus $\neg S_1 \rightarrow L_1$ folgt, dass Aufgabe 1 gelöst werden kann. $S_2 \rightarrow \neg L_2$ erlaubt keine Aussage darüber, ob Aufgabe 2 gelöst werden kann. Da wir den Wert von L_2 nicht kennen, können wir aus $L_1 \wedge L_2 \rightarrow B$ **keine Aussage** über das Bestehen der Prüfung machen.
Fehlende Informationen: Wir müssten zusätzlich wissen, ob Aufgabe 2 gelöst werden kann oder ob auch nur das Lösen von Aufgabe 1 zum Bestehen reicht.

3. - Beim **Clustering** werden Daten in Cluster mit ähnlichen Eigenschaften unterteilt, wobei ein Ähnlichkeitsmaß defniert und die Anzahl der Cluster vorgegeben wird.
Anwendung: Als „preprocessing“-Schritt, Kunden- und Marktanalyse, ...
- **PCA** ist ein Verfahren zur linearen Dimensionalitätsreduktion, bei dem Datenpunkte aus einem p-dimensionalen Raum so auf einen q-dimensionalen Unterraum ($q < p$) abgebildet werden, dass möglichst viel Information erhalten bleibt. Die Hauptkomponenten sind dabei Linearkombination den Input-Dimensionen, wobei die erste Hauptkomponente derjenigen Richtung enspricht, entlang der die Varianz am größten ist. Alle weiteren Hauptkomponenten sind orthogonal

zu den vorherigen und erklären jeweils den maximalen Anteil an verbleibender Varianz.
Anwendung: Bestimmung vor Korrelationen in Daten, als „preprocessing“-Schritt, (2D-)Darstellungen von Daten, ...

- Ein **Autoencoder** ist ein Verfahren zur nicht-linearen Dimensionalitätsreduktion, und besteht in der Regel aus zwei neuronalen Netzen. Der Encoder tranformiert einen hochdimensionalen Input (z.B. ein Bild) in einer niederdimensionale Form (den Code), und der Decoder transformiert den Code zurück zu eine Vektor der Inputgröße.
  Anwendung: Bildinterpolation, Bildsuche, Außreißererkennung, ...

Aufgabe 2 *Klassifikation*

Solution 2 *Classification*

1. Da die Funktionswerte des linearen Klassifikators kontinuierlich sind, aber die Labels nur zwei diskrete Werte annehmen können, ist die Abweichung der Funktionswerte von den Labels bei Datenpunkten, die sehr weit von der Trennlinie liegen, sehr hoch. Dadurch fließen die Punkte weit von Trennlinie mit hohen Fehlern in die Verlustfunktion ein, wodurch die Trennlinie verfälscht wird. Das ist insbesondere bei Datensätzen mit Ausreißern relevant.
   Beispieldatensatz: enthält Ausreißer.

2. $\sigma(f(x)) = \frac{1}{1+e^{-f(x)}}$

   Die logistische Funktion ist besser für Klassifikationen geeignet, weil sie nur Funktionswerte zwischen 0 und 1 annehmen kann, und bei großem Abstand von der Trennlinie gegen diese beiden Werte konvergiert.
   Verlustfunktion: Kreuzentropie:

$$L = \sum_i c_i \log \sigma(f(x_i)) + (1 - c_i) \log[1 - \sigma(f(x_i))]$$

3. Die Softmax-Funktion (für jede Klasse i bei K Klassen):

$$\sigma_i(f(x)) = \frac{e^{f_i(x)}}{\sum_{k=1}^{K} e^{f_k(x)}}$$

4. Es muss eine passende Tranformation der Inputvektoren gefunden, werden, so dass die Datenpunkt in mit den tranformierten Inputvektoren linear separierbar sind. Ein Beispiel in 2D sind kreisförmig angeordnete Klasse mit der Inputtranformation $x‘ = x_1^2 + x_2^2$.
   Beispieldatensatz: nicht-lineare Trennlinie
   Außerdem soll die Formel für die Trennlinie angegeben werden.

5. Bei einem stark unausgewogenen Datensatz, in dem die meisten Punkte einer einzigen Klasse angehören, reicht es für eine hohe Genauigkeit aus, für alle Punkte die Überschussklasse vorhersagen, so dass für alle Punkt der anderen Klasse eine falsche Vorhersage gemacht wird.

Aufgabe 3 *Hyperparameter und Ensembles*

1. Lineare (keine) Aktivierungsfunktion, da Ausgabebereich unbeschränkt.

2. a) P3, P2, P1.
   Erklärung (nicht gefragt!): Wir testen auf P3, daher erhält man den geringsten Fehler wenn man auf P3 optimizert. Auf P1 zu optimieren würde sehr stark auf P1 overfitten, was den größten Fehler ergeben würde.

   b) P2 sollte als validation set verwendet werden, da nur so die Generalisierung auf ungesehene Daten richtig getestet werden kann.

3. a) Der ursprüngliche Datensatz habe die Größe N. Für jedes Modell im Ensemble zieht man N samples mit Zurücklegen aus diesem ursprünglichen Datensatz. Dadurch kann es vorkommen, dass manche Datenpunkte mehrfach vorkommen, und andere gar nicht. Der Vorteil ist dabei, dass die Datensätze der einzelnen Modelle im Ensemble nicht vollständig überlappen und damit unabhängiger sind, als wenn man den gesamten Datensatz für jedes Modell verwendet. Damit verringert man die Varianz im Vergleich zum Szenario, in dem man jedes Modell auf dem gesamten Datensatz trainiert.

   b) Die Rechnung ist nicht richtig, da die Datensätze und damit die Modelle beim Bagging nicht vollständig unabhängig sind.

   c) Das kann durchaus sinnvoll sein, um die Diversität im Ensemble zu erhöhen. Ansonsten kann es vorkommen, dass die Modelle sehr ähnliche Vorhersagen machen, auch für Punkte außerhalb des Trainingsbereiches.

Aufgabe 4 *Suchprobleme*

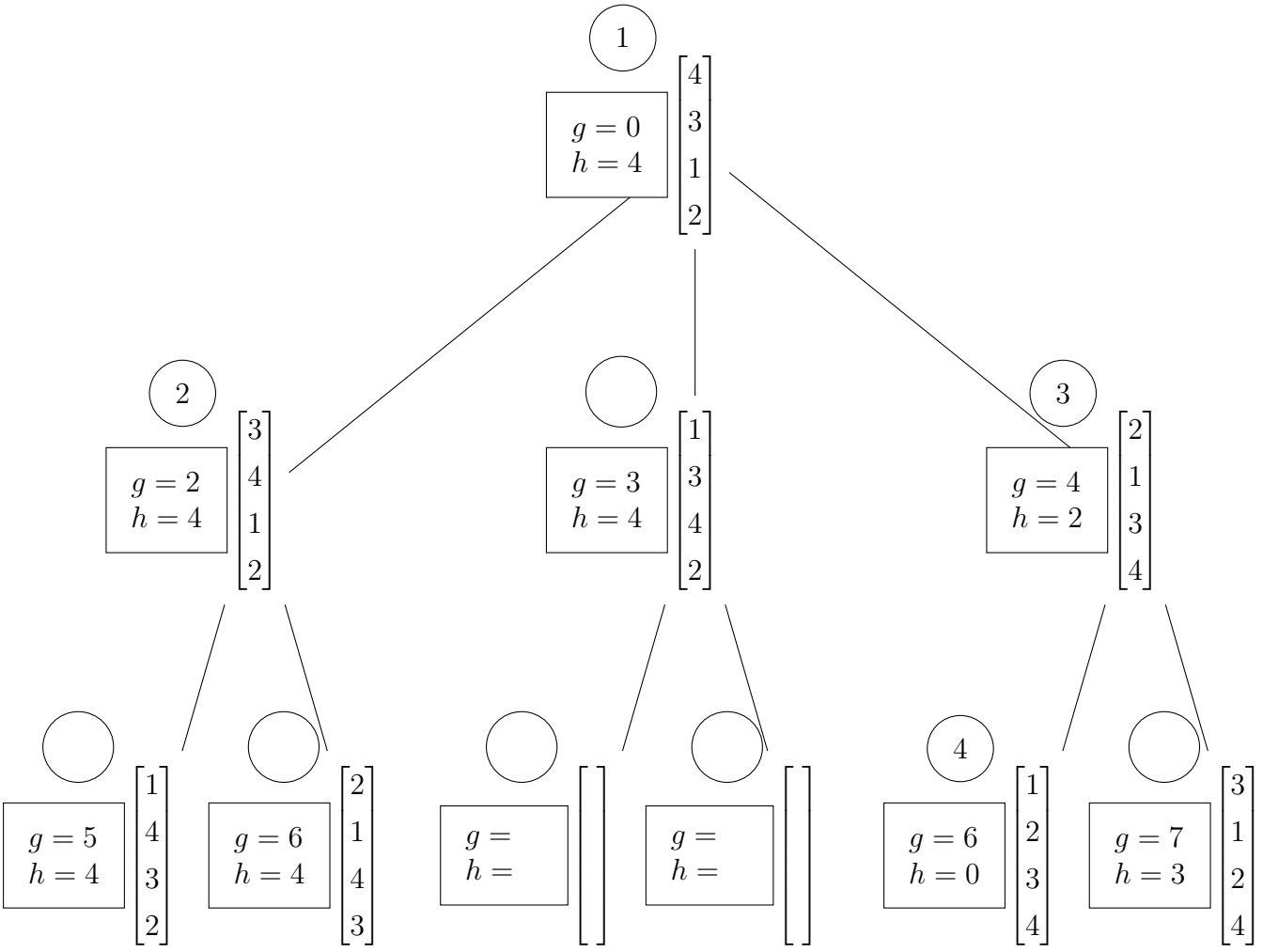
1. Ein Suchproblem besteht aus einem Zustandsraum, einer Nachfolgerfunktion (mit Aktionen, Kosten), ein Startzustand und ein Zieltest/Zielzustand.

2. Es hat $4! = 4 * 3 * 2 * 1 = 24$ Zustände.

3. Mit linear wachsenden Anzahl an Pfannkuchen gibt es exponentiell viele Zustände. Dies führt zu extrem langen Berechnungszeiten und der Computer kann alle Zustände nicht mehr aufzählen.

4. Die Heuristik ist zulässig. Dazu müssen wir zeigen, dass die Heuristik die wahren Kosten niemals überschätzt. Um den größten Pfannkuchen richtig zu sortieren, müssen wir $h(s)$ viele Pfannkuchen umdrehen (da er so weit nach unten einsortiert werden muss). Damit sind die wahren Kosten immer größer als $h(s)$.

5. Suchbaum:



Aufgabe 5 *Bayesian Networks and MDP*

1. The joint pdf of the given Bayesian Network is

$$P(A, B, C, D) = P(A) \cdot P(B) \cdot P(C \mid A, B) \cdot P(D \mid B, C)$$

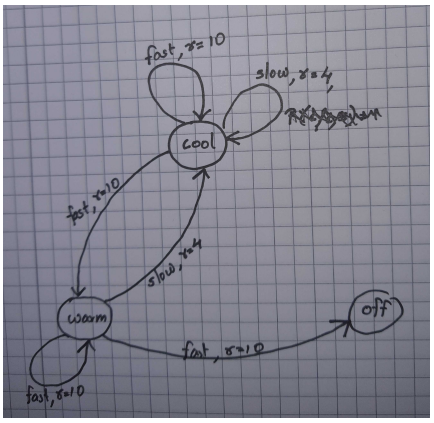
The probability that all are False (F) is

$$\begin{aligned} P(A = F, B = F, C = F, D = F) =& P(A = F) \cdot P(B = F) \cdot P(C = F \mid A = F, B = F) \\ & \cdot P(D = F \mid B = F, C = F) \\ =& 0.8 \cdot 0.3(1 - 0.2)(1 - 0.9) \\ =& 0.8 \cdot 0.3 \cdot 0.8 \cdot 0.1 \\ =& 0.0192 \end{aligned}$$

2.

$$\begin{aligned} P(A, C, \neg D) =& P(A, B, C, \neg D) + P(A, \neg B, C, \neg D) \\ =& P(A) \cdot P(B) \cdot P(C \mid A, B) \cdot P(\neg D \mid B, C) \\ & + P(A) \cdot P(\neg B) \cdot P(C \mid \neg B, A) \cdot P(\neg D \mid \neg B, C) \\ =& 0.2 \cdot 0.7 \cdot 0.6 \cdot (1 - 0.3) + 0.2 \cdot 0.3 \cdot 0.1 \cdot (1 - 0.8) \\ =& 0.06 \text{(no need for exact final answer the step till above is okay too)} \end{aligned}$$

3.



4. If in state cool then move fast. If in state warm then move slow.

Aufgabe 6 *Gaussian Mixture Models*

1. Marginal Log-Likelihood:

$$\mathcal{L} = \sum_{i=1}^{N} \log \left(\sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{x}_i | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right)$$

2. Reasons could be:
   - Gradient depends on all other components
   - Cyclic dependency
   - There exist no closed form solution
   - (Typically very slow convergence)
   - Sum does not go well with the log

3. - E-Step: This is the Expectation Step. Here, we compute the cluster probabilities aka responsibilities for each sample q_{ik}
   - M-Step: This is the Maximization Step. We compute the maximum likelihood estimate of the Coefficients/Weights π_k, means $\boldsymbol{\mu}_k$ and covariances $\boldsymbol{\Sigma}_k$.

4. - Modell mit der größten Likelihood: (c).
   - Welches Modell würden Sie im Allgemeinen wählen: (b)
     Begründung: (c) Overfitted auf das Noise im aktuellen Datensatz. Zwar hat es die größte Train Likelihood, aber (b) gibt die generelle Struktur des Datensatzes besser wieder. (a) Hingegen ist zu allgemein, und beschreibt nicht die Multimodalität des Datensatzes.

5. - E-Step: Berechnung der Responsibilites $q_{1,1}, q_{2,1}, q_{3,1}$:

$$q_{1,1} = \frac{\pi_1 \mathcal{N}(\boldsymbol{x}_1 | \mu_1, \Sigma_1)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(\boldsymbol{x}_1 | \mu_j, \Sigma_j)} = \frac{\frac{1}{2}\frac{25}{100}}{\frac{1}{2}\left(\frac{25}{100} + \frac{5}{100}\right)} = \frac{25}{30}$$

$$q_{2,1} = \frac{\frac{1}{2}\frac{25}{100}}{\frac{1}{2}\left(\frac{25}{100} + \frac{35}{100}\right)} = \frac{25}{60}$$

$$q_{3,1} = \frac{0}{\frac{5}{100}} = 0$$

   - M-Step:

$$\mu_1 = \frac{\sum_i q_{i,1} x_i}{\sum_i q_{i,1}} = \frac{\frac{25}{30} \times (-1) + \frac{25}{60} \times 1 + 0 \times 5}{\frac{25}{30} + \frac{25}{60}} = \frac{-\frac{25}{60}}{\frac{75}{60}} = -\frac{1}{3}.$$

Aufgabe 7 *Recurrent Neural Networks*

1. See lecture slides
2. $n_h * n_h + n_h * n_x + n_h + n_y * n_h + n_y$
3. Option 1: Nutze die letzte Ausgabe als Input des MLP
   Option 2: Nutze sum aggregation aller Ausgaben
   Sigmoid activation, da binäre Klassifikation.
4. Sequenzen unterschiedlicher Länge können klassifiziert werden.

Aufgabe 8 *Reinforcement Learning*

1. The goal of RL is to find a policy π maximizes the expected discounted sum of future rewards.

2. if s' is terminal:
$$\delta = r(\boldsymbol{s}, \boldsymbol{a}) - Q(\boldsymbol{s}, \boldsymbol{a})$$
else:
$$\delta = r(\boldsymbol{s}, \boldsymbol{a}) + \gamma \max_{\boldsymbol{a}'} Q\left(\boldsymbol{s}', \boldsymbol{a}'\right) - Q(\boldsymbol{s}, \boldsymbol{a})$$

3. $$Q((3,2), \mathrm{N}) = \underline{50} \quad Q((1,2), \mathrm{S}) = \underline{0} \quad Q((2,2), E) = \underline{12.5}$$
2 points, 1 point and 3 points respectively for the correct answers wit steps.

Steps

Q-values obtained by Q-learning updates
$$\begin{aligned} Q(s,a) &\leftarrow (1-\alpha)Q(s,a) + \alpha\left(R\left(s,a,s'\right) + \gamma \max_{a'} Q\left(s',a'\right)\right) \\ &= Q(s,a) + \alpha\left(R\left(s,a,s'\right) + \gamma \max_{a'} Q\left(s',a'\right) - Q(s,a)\right). \end{aligned}$$

- To calculate $Q((3,2), \mathrm{N})$, we only need transitions $(3,2), \mathrm{N}, (3,3), +100$. Using this in above equation,
$$\begin{aligned} Q((3,2), N) &= Q(s,a) + \alpha\left(R\left(s,a,s'\right) + \gamma \max_{a'} Q\left(s',a'\right) - Q(s,a)\right) \\ &= 0 + 0.5\left(100 + 0.5 \cdot 0 - 0\right) \\ &= 50. \end{aligned}$$
- There is no transitions for state action pair in $Q((1,2), \mathrm{S})$, so it retains the initial value of 0.
- To calculate $Q((2,2), E)$, we only have 2 transitions available and hence do Q update two times.
  First with **Episode 2** transitions $(2,2), \mathrm{E}, (3,2), 0$. Using this in the Q update equation,
$$\begin{aligned} Q((2,2), E) &= Q(s,a) + \alpha\left(R\left(s,a,s'\right) + \gamma \max_{a'} Q\left(s',a'\right) - Q(s,a)\right) \\ &= 0 + 0.5\left(0 + 0.5 \cdot 0 - 0\right) \\ &= 0. \end{aligned}$$
  Second with **Episode 3** transitions $(2,2), \mathrm{E}, (3,2), 0$. Note that by this time $Q(3,2)$ values have changed as we seen in the previous question and $\max_{a'} Q\left(3,a'\right) = Q((3,2), \mathrm{N}) = 50$. Using this in the Q update equation,
$$\begin{aligned} Q((2,2), E) &= Q(s,a) + \alpha\left(R\left(s,a,s'\right) + \gamma \max_{a'} Q\left(s',a'\right) - Q(s,a)\right) \\ &= 0 + 0.5\left(0 + 0.5 \cdot 50 - 0\right) \\ &= 12.5. \end{aligned}$$