Künstliche Intelligenz und Machine Learning

Klausur

Duale Hochschule Baden-Württemberg Mannheim

Kurs:
WWI 22DSB

Datum und Uhrzeit:
09.02.2024, 10:00 Uhr – 11:00 Uhr

Matrikelnummer:
☐☐☐☐☐☐☐

Aufgabe	**Punkte**	**Prozent**	**Erreicht**
1 Principal Component Analysis	13	21.66 %	
2 Hinge Loss Klassifikator	10	16.67 %	
3 Bayes'sche Entscheidungstheorie	10	16.67 %	
4 Evaluation von Machine Learning Modellen	12	20.00 %	
5 Wahr-Falsch Aufgaben	15	25.00 %	
Gesamt	60	100 %	

Dozent: ______________________
(Daniel Wehner)

Note: ______________________

Hinweise

1. Es gilt die Prüfungsordnung der DHBW.
2. Als Hilfsmittel sind erlaubt
   - ein nicht programmierbarer Taschenrechner, sowie ein
   - zweiseitig handbeschriebenes Cheat-Sheet.
3. Die Benutzung weiterer Hilfsmittel ist nicht erlaubt (man beachte § 33 der Prüfungsordnung).
4. Überprüfen Sie zunächst, ob Sie alle Aufgabenblätter erhalten haben, bevor Sie mit der Bearbeitung der Aufgaben beginnen.
5. Die Klausur besteht aus **60 Punkten** und ist innerhalb von **60 Minuten** zu lösen.
6. Notieren Sie Ihre Antworten grundsätzlich auf dem beiliegenden Konzeptpapier. Eine Ausnahme hiervon sind Single- bzw. Multiple-Choice Aufgaben.
7. Es steht Ihnen frei, die Fragen entweder auf Englisch oder auf Deutsch zu beantworten.

§ 33 der Prüfungsordnung (Bewertung bei Sanktionen)
Eine Modulprüfung, eine Prüfungsleistung oder ein Prüfungsteil werden ohne inhaltliche Bewertung mit 'nicht ausreichend' (5,0) benotet (...), wenn die zu prüfende Person (...) gemäß der Regelungen des § 35 versucht, das Ergebnis der Prüfungsleistung durch Täuschung oder Benutzung nicht zugelassener Hilfsmittel zu beeinflussen oder nach § 36 den ordnungsgemäßen Prüfungsablauf stört.

1 Principal Component Analysis

(13 Punkte)

1.1 Es sei die Matrix

$$\boldsymbol{A} := \begin{pmatrix} -1 & 2 \\ 2 & -1 \end{pmatrix} \in \mathbb{R}^{2\times 2}$$

gegeben. Bestimmen Sie die Eigenwerte von $\boldsymbol{A}$ sowie die zugehörigen Eigenräume. Bestimmen Sie ferner die Matrizen $\boldsymbol{\Lambda} \in \mathbb{R}^{2\times 2}$ und $\boldsymbol{U} \in \mathbb{R}^{2\times 2}$, so dass

$$\boldsymbol{A} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^\top$$

gilt.

☐/5

1.2 Handelt es sich bei der Matrix $\boldsymbol{A}$ um eine Kovarianzmatrix? Begründen Sie Ihre Behauptung!

☐/1

1.3 Es sei der zweidimensionale Datensatz

$$\mathcal{D} := \{(1,0), (2,-1), (-1,3), (0,1)\}$$

gegeben. Berechnen Sie die Kovarianzmatrix $\boldsymbol{\Sigma}$ für diesen Datensatz. Benutzen Sie zu diesem Zweck die Formel ohne *Bias*-Korrektur.

<u>**Hinweis:**</u> Runden Sie alle Zwischenergebnisse auf zwei Nachkommastellen.

☐/4

1.4 Die Eigenwerte der Kovarianzmatrix $\boldsymbol{\Sigma}$, welche in Teilaufgabe 1.3 bestimmt wurde, lauten gerundet $\lambda_1 \approx 0.03$ und $\lambda_2 \approx 3.41$. Ferner ist $\boldsymbol{u}^1 \approx \begin{pmatrix} 0.8 & 0.6 \end{pmatrix}^\top$ ein Eigenvektor zum Eigenwert λ_1 und $\boldsymbol{u}^2 \approx \begin{pmatrix} 0.6 & -0.8 \end{pmatrix}^\top$ ein Eigenvektor zum Eigenwert λ_2.

Reduzieren Sie die Dimension des Datensatzes $\mathcal{D}$ mit dem PCA-Algorithmus auf eine Dimension!

☐/2

1.5 Wie viel Prozent der ursprünglichen Varianz des Datensatzes bleibt nach dieser Reduktion (siehe Teilaufgabe 1.4) erhalten?

☐/1

2 Hinge Loss Klassifikator

(10 Punkte)

In der Vorlesung haben wir gesehen, wie wir das lineare Modell $\boldsymbol{\theta}^\top \boldsymbol{x}$ modifizieren können, um einen Klassifikator zu erhalten. Die Ausführungen führten auf das Konzept der logistischen Regression.

Alternativ dazu betrachten wir nun den sogenannten **Hinge Loss Klassifikator**, dessen Modellfunktion gegeben ist durch

$$h_{\boldsymbol{\theta}}(\boldsymbol{x}) := \operatorname{sign}(\boldsymbol{\theta}^\top \boldsymbol{x}). \tag{1}$$

Die Funktion $\operatorname{sign}(z)$ gibt hierbei das Vorzeichen, also -1 oder $+1$, des Arguments z aus. Für das Training des Klassifikators wird der **Hinge Loss** benutzt. Für ein vorgegebenes Trainingsbeispiel $(\boldsymbol{x}, y)$ mit $\boldsymbol{x} \in \mathbb{R}^{M+1}$ und $y \in \{-1, +1\}$ wird dieser gemäß

$$\ell^{\text{Hinge}}(\boldsymbol{x}, y; \boldsymbol{\theta}) := \begin{cases} 1 - y \cdot \boldsymbol{\theta}^\top \boldsymbol{x} & \text{wenn } y \cdot \boldsymbol{\theta}^\top \boldsymbol{x} < 1 \\ 0 & \text{wenn } y \cdot \boldsymbol{\theta}^\top \boldsymbol{x} \geq 1 \end{cases} \tag{2}$$

berechnet, wobei $\boldsymbol{\theta} \in \mathbb{R}^{M+1}$. Beantworten Sie die folgenden Fragen:

2.1 Bestimmen Sie die partielle Ableitung von ℓ^{Hinge} nach dem Parameter θ_m.

<u>**Hinweis:**</u> ℓ^{Hinge} ist für $y \cdot \boldsymbol{\theta}^\top \boldsymbol{x} = 1$ nicht differenzierbar. Setzen Sie daher die Ableitung an dieser Stelle gleich 0.

☐/4

2.2 Wie lautet allgemein der Gradient $\nabla_{\boldsymbol{\theta}} \ell^{\text{Hinge}}(\boldsymbol{x}, y; \boldsymbol{\theta})$ der Fehlerfunktion ℓ^{Hinge}?

<u>**Hinweis:**</u> Der Gradient ist nur in $\boldsymbol{\theta}$ zu bestimmen!

☐/2

2.3 Es seien die beiden Beispiele

$$\left(\boldsymbol{x}^1 := \begin{pmatrix} 1 & 0 & 1 \end{pmatrix}^\top, y_1 := +1\right) \quad \text{und} \quad \left(\boldsymbol{x}^2 := \begin{pmatrix} 1 & -1 & 2 \end{pmatrix}^\top, y_2 := -1\right)$$

gegeben. Wie lautet für diese beiden Beispiele jeweils der stochastische Gradient bei vorgegebenem $\boldsymbol{\theta} := \begin{pmatrix} 1 & 1 & 2 \end{pmatrix}^\top$?

☐/2

2.4 Die Fehlerfunktion (2) führt auf einen *Maximum Margin* Klassifikator. Beschreiben Sie kurz, was darunter zu verstehen ist. Welcher Vorteil besteht gegenüber der logistischen Regression?

☐/2

3 Bayes'sche Entscheidungstheorie

(10 Punkte)

Sie möchten Pilze sammeln. Sie sind sich jedoch nicht sicher, welche Pilze essbar und welche giftig sind. Ihnen steht allerdings ein Trainings-Datensatz zur Verfügung, der für einige Pilze – aufgeschlüsselt nach den Attributen **Geruch**, **Hutfarbe** und **Standort** – aufzeigt, welcher Kategorie diese angehören.

Die folgende Tabelle zeigt bereits in voraggregierter Form, wie viele Beispiele jeweils welche Attributausprägungen aufweisen:

Klasse	**Geruch**			**Hutfarbe**				**Standort**		**Summe**
	neutral	würzig	faul	rot	grau	braun	weiß	Wald	Wiese	
essbar	2389	0	0	366	805	714	504	1495	894	2389
giftig	559	192	720	402	651	358	60	897	574	1471
Summe	2948	192	720	768	1456	1072	564	2392	1468	3860

Nicht für reale Anwendungsfälle benutzen!

3.1 Bei einem Spaziergang treffen Sie auf einen Pilz und ermitteln für ihn die folgenden Attributwerte:

Geruch neutral
Hutfarbe grau
Standort Wald

Bestimmen Sie für dieses Exemplar alle notwendigen Wahrscheinlichkeiten, um anschließend eine Klassifikation mittels des *naïve Bayes* Algorithmus vornehmen zu können. Sagen Sie dabei die wahrscheinlichste Klasse vorher.

☐/4

3.2 Sie möchten unbedingt vermeiden, einen giftigen Pilz als essbar einzustufen *(False Positive)* und führen daher das folgende Kostenmodell ein: Für korrekte Klassifikationen seien keine Kosten vorgesehen ($\ell_{\text{tp}} := 0 =: \ell_{\text{tn}}$). Für einen *False Negative* sollen hingegen Kosten von $\ell_{\text{fn}} := 2$ entstehen.

Wie hoch müssen Sie die Kosten ℓ_{fp} für einen *False Positive* mindestens ansetzen, damit das Beispiel aus Teilaufgabe 3.1 unter Berücksichtigung dieses Kostenmodells nicht als essbar klassifiziert wird?

☐/4

3.3 Sie finden einen Pilz mit gelber Hutfarbe. Auf welches Problem stoßen Sie bei dessen Klassifikation und wie können Sie dieses Problem umgehen?

<u>**Hinweis:**</u> Sie müssen hier keine Berechnungen anstellen.

☐/2

4 Evaluation von Machine Learning Modellen

(12 Punkte)

4.1 Sie haben ein neuronales Netz auf einem Datensatz mit den beiden Klassen $\oplus$ und $\ominus$ trainiert. Das Modell erzielte auf einem separaten Testdatensatz der Größe $N = 185$ einen *Recall* von 0.25 und einen F_1-*Score* von 0.381. Ferner erkannte das Modell 40 Instanzen der positiven Klasse $\oplus$ korrekt.

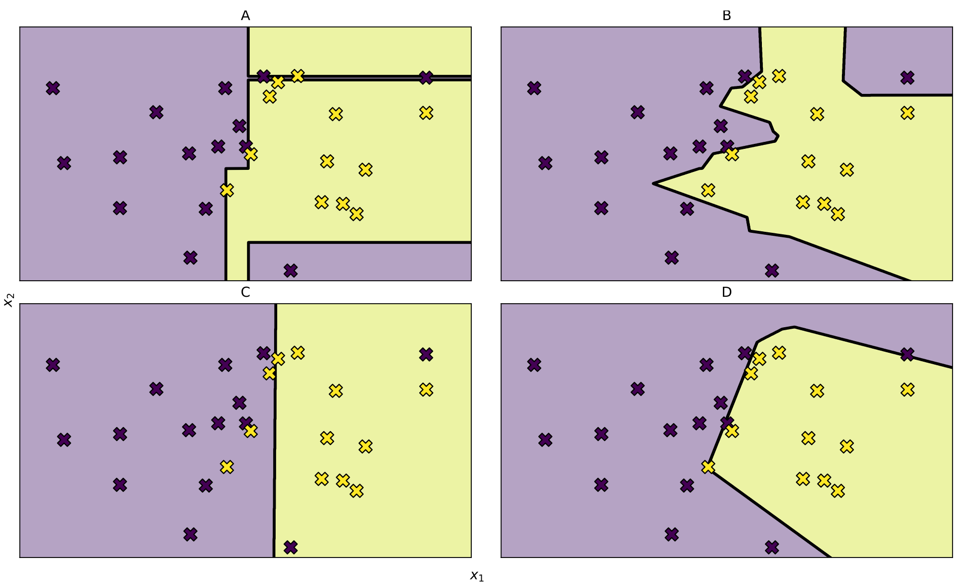
Stellen Sie die Konfusionsmatrix des Modells auf dem Testdatensatz auf.

Hinweis: Runden Sie alle Zwischenergebnisse auf zwei Nachkommastellen.

/5

4.2 Im Folgenden wurde derselbe Datensatz mit vier verschiedenen Klassifikatoren klassifiziert. Geben Sie jeweils an, welcher Klassifikator welche Entscheidungsgrenze erzeugt hat. Zur Auswahl stehen:

– *Logistische Regression* (ohne Verwendung von Basisfunktionen),
– *1-nearest Neighbor*,
– *Entscheidungsbaum* und
– *Multi-Layer Perceptron* (regularisiert).



Zwei der Modelle overfitten auf den Daten. Identifizieren Sie diese Modelle und geben Sie jeweils eine geeignete Gegenmaßnahme an.

/4

4.3 Beschreiben Sie was man unter dem *Bias-Variance-Tradeoff* versteht. Wie hängen diese Begriffe mit *Underfitting* und *Overfitting* zusammen?

/3

5 Wahr-Falsch Aufgaben

(15 Punkte)

Kreuzen Sie jeweils an, ob die folgenden Aussagen wahr (**W**) oder falsch (**F**) sind.

<u>**Hinweis:**</u> Für jede korrekte Antwort erhalten Sie einen Punkt, **für jede falsche Antwort verlieren Sie einen Punkt!** Nicht beantwortete Fragen werden mit 0 Punkten bewertet. Ein möglicherweise negatives Gesamtergebnis für diese Aufgabe wird durch die Punktzahl 0 ersetzt.

Aussage		**W**	**F**
5.1	Der Gradient der Funktion $f : \mathbb{R}^2 \to \mathbb{R}, (x, y) \mapsto \ln(2xy)$ lautet $\left(\frac{1}{2x}, \frac{1}{2y}\right)^\top$.	☐	☐
5.2	Die Newton-Methode zur Optimierung von Funktionen ist, wie *Gradient Descent*, eine *second-order* Methode.	☐	☐
5.3	Das Theorem von Bayes lautet $$p(A\|B) = \frac{p(B\|A)p(A)}{p(A \cap B)}$$ für zwei beliebige Ereignisse A und B.	☐	☐
5.4	Der Normalisierungsterm im Theorem von Bayes ist zum Zwecke der Klassifikation unerheblich, weswegen dieser in praktischen Anwendungen häufig ignoriert wird.	☐	☐
5.5	Die Modellparameter eines *naïve Bayes* Klassifikators werden üblicherweise mithilfe der *Maximum Likelihood* Methode bestimmt.	☐	☐
5.6	Die Abkürzung *i.i.d.* steht für *independent and identically distributed*.	☐	☐
5.7	Die Formel zur Bestimmung der optimalen Koeffizienten eines *Ridge-Regression* Modells lautet $$\boldsymbol{\theta}^\star = (\boldsymbol{\Phi}^{-1}\boldsymbol{\Phi} + \lambda \boldsymbol{I})^\top \boldsymbol{\Phi}^\top \boldsymbol{y}.$$	☐	☐

Aussage		W	F
5.8	Der *Batch Gradient* basiert auf einem einzelnen Trainingsbeispiel und ist daher nur eine grobe Approximation des wahren Gradienten.	☐	☐
5.9	Bei der logistischen Regression handelt es sich um einen linearen Klassifikator, der meistens mit der *Squared Error* Fehlerfunktion trainiert wird.	☐	☐
5.10	Die *Softmax* Aktivierungsfunktion ζ wird üblicherweise in einem *Hidden Layer* eines neuronalen Netzes benutzt.	☐	☐
5.11	Ein zu kleiner Wert für k führt beim *k-nearest Neighbors* Algorithmus zu Underfitting.	☐	☐
5.12	Die Entropie des Datensatzes $$\mathcal{D} := \{A, B, C, A, A, B, B, C\}$$ ist ungefähr 1.56, der Gini-Index hingegen 0.80. A, B und C stellen dabei die Klassenlabels dar.	☐	☐
5.13	Entropie und Gini-Index führen immer auf den gleichen Entscheidungsbaum.	☐	☐
5.14	Der *KMeans* Algorithmus kennt ein Ausreißerkonzept, d. h. ein Datenpunkt kann keinem Cluster zugeordnet werden, wenn der Punkt zu weit von allen Clusterzentroiden entfernt ist.	☐	☐
5.15	Für den *KMeans* Algorithmus hat die verwendete Distanzmetrik keinen Einfluss auf das Ergebnis des Clusterings.	☐	☐

☐/15