UNIVERSITÄT LEIPZIG

LEHRSTUHL INTELLIGENT LANGUAGE TECHNOLOGIES

Probeklausur "Grundlagen des Maschinellen Lernens" WS 2023/24

Antwort

ANSWER KEY COPY

Prof. Dr. Martin Potthast

Dienstag, 23. Januar 2024

Name, Vorname: .....

Matrikelnummer: .....

Studiengang: .....

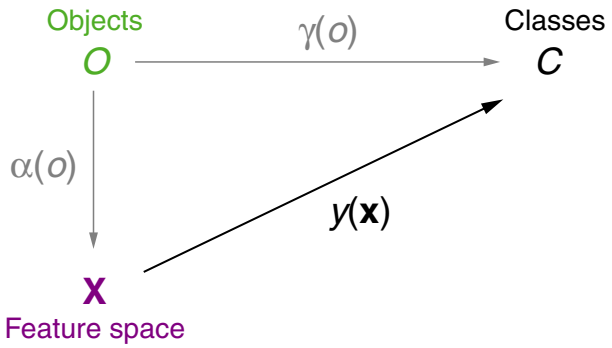
Hinweise:

- Die Bearbeitungszeit beträgt 60 Minuten. Zu Beginn werden 10 Minuten Einlesezeit gewährt.
- Prüfen Sie die Vollständigkeit Ihres Klausurexemplares (Deckblatt, 6 Aufgaben, 13 Blätter).
- Schreiben Sie zu Beginn der Klausur auf das Deckblatt und jedes weitere Blatt Ihren Namen, Ihren Vornamen und Ihre Matrikelnummer.
- Verwenden Sie keinen Rotstift, keinen Bleistift und kein eigenes Konzeptpapier. Benutzen Sie nur den Platz unter den Aufgabenstellungen und die Rückseiten der Blätter für Ihre Lösungen. Sollten Sie zusätzliches Konzeptpapier benötigen, so wenden Sie sich bitte an die Aufsicht.
- Werden zu einer (Teil-)Aufgabe mehrere alternative Lösungen angegeben, gilt die Aufgabe als nicht gelöst. Nicht zu bewertende Bestandteile sind eindeutig zu kennzeichen.
- Als Hilfsmittel ist ein einseitig eigenhändig handbeschriebenes DIN-A4 Blatt zugelassen. Dieses Blatt wird mit eingesammelt; schreiben Sie auch auf dieses Blatt Ihren Namen und ihre Matrikelnummer. Außerdem ist ein nicht-programmierbarer Taschenrechner erlaubt.
- Viel Erfolg!

Aufgabe	1	2	3	4	5	6	$\sum$
Erreichbare Punkte	6	6	8	8	10	10	48
Erreichte Punkte							

Aufgabe 1 : Grundlagen & Evaluation (2+3=5 Punkte)

(a) Gegeben ist die folgende Abbildung, welche die Bestandteile der Spezifikation von Lernaufgaben aufzeigt. Vervollständigen sie die darunter stehende Tabelle am Beispiel des Problems der Klassifikation von Tieren zu verschiedenen Arten anhand von erhobenen Messwerten.



Bestandteil	Beispiel
Objects O	zu klassifizierende Tiere
	numerische Messwerte zu Gewicht, Größe, und Körpertemperatur
	Menge der zu unterscheidenden Tierarten
	fachkundige:r Zoolog:in, welche:r Tiere identifiziert
	Waage, Thermometer, Maßstab
$y(\mathbf{x})$	zu lernender Klassifikator

Antwort

Bestandteil	Beispiel
Objects O	zu klassifizierende Tiere
Feature Space $\mathbf{X}$	numerische Messwerte zu Gewicht, Größe, und Körpertemperatur
Classes C	Menge der zu unterscheidenden Tierarten
$\gamma(o)$	fachkundige:r Zoolog:in, welche:r Tiere identifiziert
$\alpha(o)$	Waage, Thermometer, Maßstab
$y(\mathbf{x})$	zu lernender Klassifikator

(b) Gegeben ist die folgende Menge D von Beispielen, die Tiere (c) durch ihr Körpergewicht in Kilogramm ($\mathbf{X}$) charakterisieren:

Beispiel	$\mathbf{X}$	c
1	3	Katze
2	4	Hund
3	2	Katze
4	1	Hund
5	2	Katze
6	5	Hund

Sie möchten ein Klassifikationsmodell mit k-facher Kreuzvalidierung auf diesem Datensatz evaluieren, wobei der Wert k auf 3 gesetzt ist.

(b1) Wie viele Klassifikatoren werden trainiert (ohne das bereits vorab auf einem großen Datensatz trainierte Klassifikationsmodell, das evaluiert werden soll)?

Antwort

3

(b2) Wie viele Beispiele befinden sich je in der *Trainingsmenge* D_{tr} und *Testmenge* D_{test} für jeden dieser Klassifikatoren?

Antwort

$|D_{tr}| = 4, |D_{test}| = 2$

(b3) Nehmen Sie an, dass Ihr Klassifikationsmodell nach der folgenden Gleichung funktioniert:

$$y(x) = \begin{cases} \text{Katze, wenn} & x < 3 \\ \text{Hund, wenn} & x \geq 3 \end{cases}.$$

Berechnen Sie die wahre Fehlerrate $Err^*(y)$ [*True Misclassification Rate*] dieses Klassifikators unter der Annahme, dass alle Beispiele in D zur Evaluation verwendet werden.

Antwort

Beispiel	$\mathbf{X}$	c	$y(\mathbf{x})$	$c = y(\mathbf{x})$
1	3	Katze	Hund	0
2	4	Hund	Hund	1
3	2	Katze	Katze	1
4	1	Hund	Katze	1
5	2	Katze	Katze	0
6	5	Hund	Hund	1

$$\text{Err}^*(y()) = \frac{|\ \{\mathbf{x} \in X : y(\mathbf{x} \neq c_x)\}\ |}{|\ X\ |} = \frac{2}{6}$$

Aufgabe 2 : Concept Learning (6 Punkte)

Betrachten Sie den gegebenen Merkmalsraum für das Zielkonzept $EnjoySurfing$ mit den folgenden sechs Merkmalen und ihren jeweiligen Domänen:

Sky $= \{$*sunny, rainy*$\}$, *Temperature* $= \{$*warm, cold*$\}$, *Humidity* $= \{$*normal, high*$\}$,
Wind $= \{$*strong, weak*$\}$, *Water* $= \{$*cold, warm*$\}$ und *Forecast* $= \{$*same, change*$\}$.

Mit dem Kandidaten-Eliminierungs-Algorithmus [*Candidate Elimination*] soll das Konzept $EnjoySurfing$ gelernt werden, das den Wert 0 für negative Merkmalsvektoren $\mathbf{x}$ [*feature vectors*] annimmt und den Wert 1 für positive. Der Lernalgorithmus hat bereits Beispiele verarbeitet, die zu den folgenden Mengen H_S und H_G geführt haben:

$H_S = \{$(*sunny, warm, normal, ?, warm, same*), (*sunny, ?, normal, weak, ?, same*)$\}$
$H_G = \{$(*sunny, warm, ?, ?, ?, same*), (*sunny, ?, normal, ?, ?, same*)$\}$

Der nächste Merkmalsvektor hat die folgenden Werte:

$\mathbf{x_1}$ =(*sunny, cold, normal, weak, cold, same*) *EnjoySurfing*$(\mathbf{x_1}) = 0$

Führen Sie den nächsten Schritt des Kandidaten-Eliminierungs-Algorithmus [*Candidate Elimination*] am Beispiel $\mathbf{x_1}$ aus. Schreiben Sie den Endzustand von H_S und H_G auf, sowie jeden Zwischenschritt, um zu begründen, warum Sie Hypothesen zu einer dieser Mengen hinzufügen oder entfernen.

Antwort

(6 POINTS)

Situation:

$H_S = \{$(*sunny, warm, normal, ?, warm, same*), (*sunny, ?, normal, weak, ?, same*)$\}$
$H_G = \{$(*sunny, warm, ?, ?, ?, same*), (*sunny, ?, normal, ?, ?, same*)$\}$
$\mathbf{x_1}$ =(*sunny, cold, normal, weak, cold, same*) *EnjoySurfing*$(\mathbf{x_1}) = 0$

Remove incosistent hypothesis from H_S and develop H_G:

ad a) Remove inconsistent hypotheses from H_S. Update set H_S:

$H_S = \{$(*sunny, warm, normal, ?, warm, same*)$\}$

ad b) Check hypotheses in H_G:

(b1) (*sunny, warm, ?, ?, ?, same*) is consistent with $\mathbf{x_1}$.

(b2) (*sunny, ?, normal, ?, ?, same*) is not consistent with $\mathbf{x_1}$ and is removed from H_G.
Minimal specializations: $\{$(*sunny, warm, normal, ?, ?, same*), (*sunny, cold, normal, ?, ?, same*), (*sunny, ?, normal, strong, ?, same*), (*sunny, ?, normal, weak, ?, same*), (*sunny, ?, normal, ?, cold, same*), (*sunny, ?, normal, ?, warm, same*)$\}$.

Out of those, the following are consistent with $\mathbf{x_1}$: $\{$(*sunny, warm, normal, ?, ?, same*), (*sunny, ?, normal, strong, ?, same*), (*sunny, ?, normal, ?, warm, same*)$\}$.

Add those specializations, which have a more-specfic counterpart in H_S, to H_G:
$H_G = \{$(*sunny, warm, ?, ?, ?, same*), (*sunny, warm, normal, ?, ?, same*), (*sunny, ?, normal, ?, warm, same*)$\}$

Remove from H_G any hypothesis that is less general than another hypothesis in H_G:
$H_G = \{$(*sunny, warm, ?, ?, ?, same*), (*sunny, ?, normal, ?, warm, same*)$\}$

Aufgabe 3 : Lineare Modelle (6+2=8 Punkte)

(a) Betrachten Sie den Incremental-Gradient-Descent-Algorithmus (IGD) für das Trainieren linearer Modelle wie unten dargestellt.

IGD(D, η)

1. *initialize_random_weights*$(\mathbf{w})$, $t = 0$
2. **REPEAT**
3. $t = t + 1$
4. FOREACH $(\mathbf{x}, c) \in D$ DO
5. $y(\mathbf{x}) =$ ☐
6. $\delta = c(\mathbf{x}) - y(\mathbf{x})$
7. $\Delta\mathbf{w} = \eta \cdot \delta \cdot \mathbf{x}$
8. $\mathbf{w} = \mathbf{w} + \Delta\mathbf{w}$
9. ENDDO
10. **UNTIL**(*convergence*$(D, y(), t)$)
11. *return*$(\mathbf{w})$

(a1) Die Modellfunktion $y(\mathbf{x})$ erscheint in Zeile 5 des Algorithmus. Schreiben Sie die Gleichung für die Modellfunktion auf, wenn $y(\mathbf{x})$ ein logistisches Regressionsmodell ist.

Antwort

$y(\mathbf{x}) = \sigma(\mathbf{w}^T\mathbf{x}) = \frac{1}{1+e^{-\mathbf{w}^T\mathbf{x}}}$

(a2) Beschreiben Sie kurz den Zweck der Zeilen 7 und 8 im Algorithmus.

Antwort

For the example $(\mathbf{x}, c(\mathbf{x}))$ from the current iteration, compute its contribution to the gradient of the loss function with respect to the parameter vector $\mathbf{w}$. Invert the sign of the gradient and scale it by η, and use it to update the parameter vector $\mathbf{w}$. This way, the parameter vector is updated approximately in the direction of the gradient of the loss function, and can approach its minimum over time.

(a3) Für die lineare Regression lautet die Modellfunktion $y(\mathbf{x}) = \mathbf{w}^T\mathbf{x}$. Erklären Sie, warum bei der Anpassung eines linearen Regressionsmodells anstelle eines logistischen Regressionsmodells mit IGD der Algorithmus – und insbesondere die Zeilen 6 und 7 – gleich bleibt.

Antwort

For linear regression, we use the squared loss ℓ_2; lines 6 and 7 then result from the derivative $\frac{\partial}{\partial \mathbf{w}} \ell_2(c(\mathbf{x}), \mathbf{w}^T\mathbf{x})$. For logistic regression, we use the logistic loss ℓ_σ; lines 6 and 7 then result from the derivative $\frac{\partial}{\partial \mathbf{w}} \ell_\sigma(c(\mathbf{x}), \sigma(\mathbf{w}^T\mathbf{x}))$. These two combinations of the respective loss and model function have the same derivative with respect to the parameters $\mathbf{w}$.

(b) Betrachten Sie das reellwertige lineare Regressionsmodell $y(\mathbf{x}) = w_0 + \sum_{j=1}^{p} w_j \cdot x_j = \mathbf{w}^T\mathbf{x}$.

(b1) Sei $\gamma()$ eine Zielfunktion mit Werten aus der Menge $\{-1, 1\}$. Überführen Sie $y()$ in einen Klassifikator $\hat{y}()$, der ebenfalls ausschließlich Werte aus dieser Menge annimmt.

$\gamma(\mathbf{x}) \approx \hat{y}(\mathbf{x}) =$ [ ]

Antwort

$\gamma(\mathbf{x}) = \mathsf{sign}(\mathbf{w}^T\mathbf{x})$

(b2) Geben Sie einen mathematischen Term (Name und Formel) an, um die Anpassungsgüte [*goodness of fit*] des linearen Regressionsmodells zu beurteilen.

Antwort

Note: either one of the answers below is correct.

Assess goodness of fit as residual sum of squares: $\mathsf{RSS}(\mathbf{w}) = \sum_{i=1}^{n}(y_i - \mathbf{w}^T\mathbf{x}_i)^2$

Assess goodness of fit as mean absolute deviation: $\mathsf{MAD}(\mathbf{w}) = 1/n \sum_{i=1}^{n} |y_i - \mathbf{w}^T\mathbf{x}_i|$

Aufgabe 4 : Bayesian Learning (1+2+5=8 Punkte)

(a) Zeigen sie den Zusammenhang von Prior-, Posterior-, und Likelihood-Wahrscheinlichkeiten.

Antwort

Durch das Bayes-Theorem ergibt sich

$$P(A_i \mid B_1, \ldots, B_p) = \frac{P(A_i) \cdot P(B_1, \ldots, B_p \mid A_i)}{P(B_1, \ldots, B_p)},$$

wobei $P(A_i)$ die Prior-, $P(A_i \mid B_1, \ldots, B_p)$ die Posterior-, und $P(B_1, \ldots, B_p \mid A_i)$ die Likelihood-Wahrscheinlichkeiten angeben.

(b) Sei $\mathbf{X}$ ein p-dimensionaler Merkmalsraum [*feature space*], sei C die Menge der k Klassen eines Zielkonzepts, und sei D eine Menge von Beispielen über $\mathbf{X} \times C$. Hierbei bezeichnen $A_1, \ldots, A_k$ Ereignisse der Art $\mathbf{C} = c_i$, und $B_j, j = 1, \ldots, p$ bezeichnen Ereignisse der Art $\mathbf{X}_j = x_j$.

Welche Wahrscheinlichkeiten sind zu schätzen, um einen Naive Bayes-Klassifikator für das Zielkonzept zu konstruieren?

Antwort

1. Estimation of the $P(A_i)$, where $A_i := \mathbf{C} = c_i, c_i \in C$.
2. Estimation of the $P(B_{j=x_j} \mid A_i)$, where $B_{j=x_j} := \mathbf{X}_j = x_j$

(c) Gegeben seien die folgenden fünf Trainingsbeispiele in einem zweidimensionalen Merkmalsraum [*feature space*]:

Example	x_1	x_2	c
1	1	0	0
2	0	1	1
3	1	0	1
4	0	1	0
5	1	0	0

Berechnen Sie auf der Grundlage dieses Datensatzes die Prior-Wahrscheinlichkeiten und Likelihoods, die zur Spezifikation eines Naive Bayes-Klassifikators erforderlich sind.

Antwort

$\hat{P}(c = 0) = \frac{3}{5}$ $\quad$ $\hat{P}(c = 1) = \frac{2}{5}$

$\hat{P}(x_1 = 0 \mid c = 0) = \frac{1}{3}$ $\quad$ $\hat{P}(x_1 = 0 \mid c = 1) = \frac{1}{2}$

$\hat{P}(x_1 = 1 \mid c = 0) = \frac{2}{3}$ $\quad$ $\hat{P}(x_1 = 1 \mid c = 1) = \frac{1}{2}$

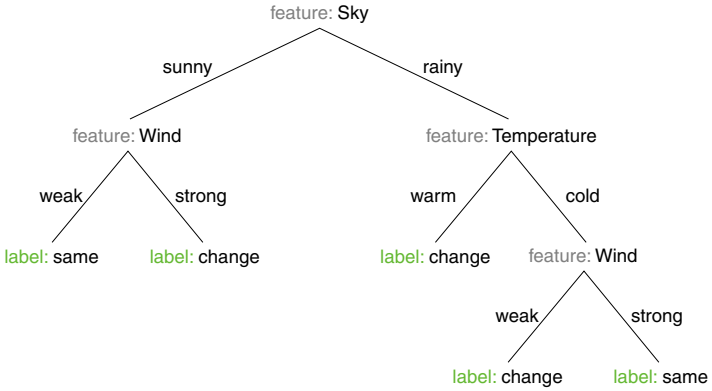
$\hat{P}(x_2 = 0 \mid c = 0) = \frac{2}{3}$ $\quad$ $\hat{P}(x_2 = 0 \mid c = 1) = \frac{1}{2}$

$\hat{P}(x_2 = 1 \mid c = 0) = \frac{1}{3}$ $\quad$ $\hat{P}(x_2 = 1 \mid c = 1) = \frac{1}{2}$

Aufgabe 5 : Entscheidungsbäume (2+5+2+1=10 Punkte)

(a) Gegeben ist der folgende Datensatz sowie ein daraus konstruierter Entscheidungsbaum für Wettervorhersagen.

Sky	Temp.	Wind	Forecast
sunny	warm	weak	same
sunny	warm	strong	change
sunny	cold	weak	same
sunny	cold	strong	change
rainy	warm	weak	change
rainy	warm	strong	change
rainy	cold	weak	change
rainy	cold	strong	same



Geben Sie für diesen Entscheidungsbaum die folgenden Größen an:

(a1) Anzahl der Blattknoten [*leaf node number*]:

Antwort

5

(a2) Höhe des Baumes [*tree height*]:

Antwort

3

(a3) Externe Pfadlänge [*external path length*]:

Antwort

$2 + 2 + 2 + 3 + 3 = 12$

(a4) Gewichtete externe Pfadlänge [*weighted external path length*]:

Antwort

Every attribute splits the examples in half, hence: $2 \cdot (2 + 2 + 2) + 3 \cdot (1 + 1) = 12 + 6 = 18$.

(b) Die Interpretation einer Booleschen Formel hängt von den Wahrheitszuweisungen ihrer booleschen Variablen ab und ist entweder "wahr" (1) [*true*] oder "falsch" (0) [*false*]. Betrachten Sie den Wahrheitswert der Formel als das Klassenlabel für den jeweiligen Vektor $\mathbf{x}$ der Wahrheitszuweisungen für die Booleschen Variablen. Ihre Aufgabe ist es, einen Entscheidungsbaum zu konstruieren, der die Vektoren der Wahrheitszuweisungen für die Boolesche Formel $(A \vee \neg B) \wedge C$ korrekt klassifiziert.

(b1) Stellen Sie die Trainingsmenge für Ihren Entscheidungsbaum [*decision tree*] als eine Tabelle von Beispielen mit den Merkmalen A, B und C und dem jeweiligen Klassenwert zusammen.

Antwort

A	B	C	$(A \vee \neg B) \wedge C$
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

(b2) Identifizieren Sie für die Teilung an der Wurzel [*root split*] des Baums (aus den Merkmalen A, B und C) das Merkmal mit der maximalen Unreinheitsreduktion [*impurity reduction*] $\Delta\iota$. Verwenden Sie das Unreinheitsmaß [*impurity function*] basierend auf der Fehlklassifikationsrate [*misclassification rate*], $\iota_{\textit{misclass}}$, als Teilungskriterium [*splitting criterion*].

Hinweis:

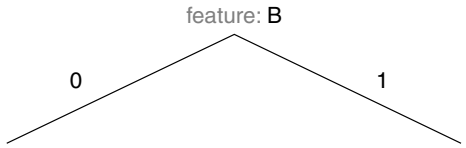
$$\iota_{\textit{misclass}}(D) = 1 - \max\left\{\frac{|\{(\mathbf{x}, c(\mathbf{x})) \in D \ : \ c(\mathbf{x}) = c_1\}|}{|D|}, \frac{|\{(\mathbf{x}, c(\mathbf{x})) \in D \ : \ c(\mathbf{x}) = c_2\}|}{|D|}\right\}$$

$$\Delta\iota = \iota_{\textit{misclass}}(D) - \left(\frac{|D_1|}{|D|} \cdot \iota_{\textit{misclass}}(D_1) \ + \ \frac{|D_2|}{|D|} \cdot \iota_{\textit{misclass}}(D_2)\right)$$

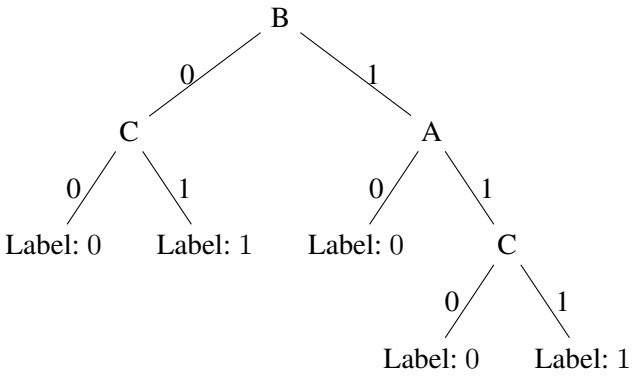
Antwort

$\iota(D) = 1 - \max\{\frac{3}{8}, \frac{5}{8}\} = \frac{3}{8}$
Merkmal A:
$\iota(D_{A=1}) = \frac{1}{2} \qquad \iota(D_{A=0}) = \frac{1}{4}$
$\Delta\iota_A = \frac{3}{8} - (\frac{1}{2} \cdot \frac{1}{2} + \frac{1}{2} \cdot \frac{1}{4}) = 0$
Merkmal B:
$\iota(D_{B=1}) = \frac{1}{4} \qquad \iota(D_{B=0}) = \frac{1}{2}$
$\Delta\iota_B = \Delta\iota_A = 0$
Merkmal C:
$\iota(D_{C=1}) = \frac{1}{4} \qquad \iota(D_{C=0}) = 0$
$\Delta\iota_C = \frac{3}{8} - (\frac{1}{2} \cdot \frac{1}{4} + \frac{1}{2} \cdot 0) = \frac{2}{8}$
Merkmal C yields the maximum impurity reduction.

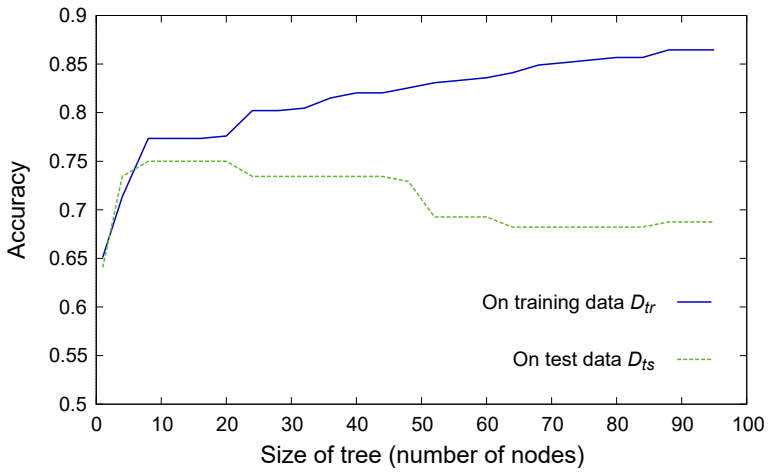
(b3) Der folgende Entscheidungsbaum verwendet Merkmal B für die erste Aufteilung. Vervollständigen Sie die Konstruktion dieses Entscheidungsbaums, so dass er alle Beispiele korrekt klassifiziert, und verwenden Sie dabei Merkmale Ihrer Wahl für die restlichen Aufteilungen. D.h., zeichnen Sie den resultierenden Baum und markieren Sie die Teilungsmerkmale an den inneren Knoten, die Werte der Teilungsmerkmale entlang der entsprechenden Kanten und die Klassen an den Blättern.



Antwort



(c) Bei der Konstruktion eines Entscheidungsbaums können wir die folgende Abbildung erstellen:



(c1) Auf welches Problem weist die zunehmende Differenz zwischen der blauen und der grünen Kurve hin? Nennen Sie den Begriff.

Antwort

Overfitting on training data D_{tr}; noise, conflicting or unseen data in test data split D_{ts}. The larger and more complex the tree, the higher the train accuracy and higher the test error rate.

(c2) Erklären Sie, wie das Stutzen [*pruning*] von Entscheidungsbäumen für die Lösung dieses Problems verwendet werden kann. Verwenden Sie die Abbildung aus der obigen Frage wieder.

Antwort

Evaluating the decision tree on validation data D_{vd} (unseen/contradictory examples) to guard against overfitting on training data D_{tr}. "Reduced Error Pruning" to minimize validation data error $\mathit{Err}(T, D_{vd})$ on constructed decision tree.

(d) Ist es angesichts eines beliebigen endlichen widerspruchsfreien Datensatzes D immer möglich, einen Entscheidungsbaum zu konstruieren, der alle Beispiele aus D korrekt klassifiziert? Erläutern Sie Ihre Antwort.

Antwort

Yes. Because similarly to CART, one can form a space partitioning over X that correctly classifies every sample in D and can be represented as binary decision tree.

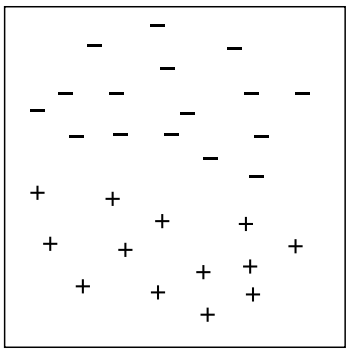
Aufgabe 6 : Neurale Netzwerke (1+1+8=10 Punkte)

(a) Erfordert ein Multilayer-Perzeptron, dass eine Beispielmenge D linear separierbar [*linearly separable*] ist, um sie korrekt zu klassifizieren?

Antwort

No.

(b) Die folgende Abbildung zeigt eine Menge von Punkten in einem zweidimensionalen Merkmalsraum [*feature space*], die zu zwei verschiedenen Klassen gehören.



Wird der Perzeptron-Algorithmus (PT) schlussendlich zu einer Hyperebene konvergieren, die alle Punkte in diesem Datensatz korrekt klassifiziert? Erläutern Sie kurz Ihre Antwort.

Antwort

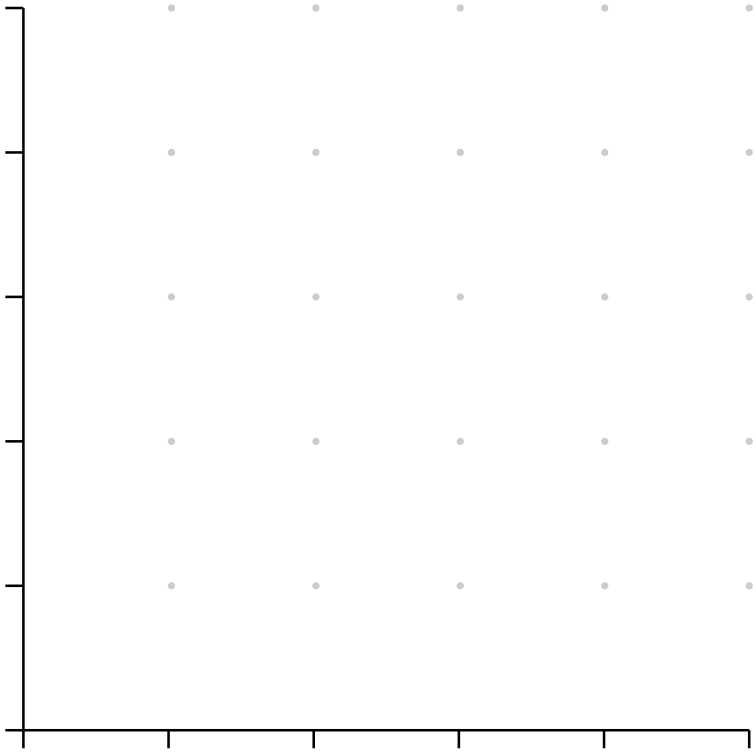
The given set of points is linearly separable. Hence, the PT algorithm will converge in a finite number of steps.

(c) Betrachten Sie ein Perzeptron mit zwei Eingängen und der Heaviside-Funktion H als Aktivierungsfunktion [*activation function*]. Gegeben ist weiterhin der folgende Datensatz, der 4 Punkte $\mathbf{x}_i$ und entsprechende Klassen c_i enthält:

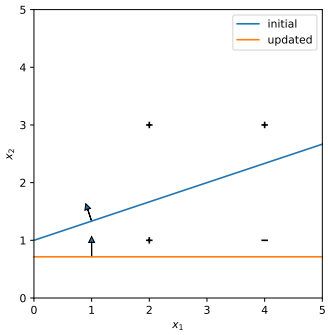
$$\begin{aligned} \mathbf{x}_1 &= (2,1), \quad c_1 = 1 \\ \mathbf{x}_2 &= (4,1), \quad c_2 = 0 \\ \mathbf{x}_3 &= (2,3), \quad c_3 = 1 \\ \mathbf{x}_4 &= (4,3), \quad c_4 = 1 \end{aligned}$$

Die Gewichte des Perzeptrons werden initialisiert durch $(w_0, w_1, w_2)^T = (-3, -1, 3)^T$

(c1) Skizzieren Sie die Datenpunkte und die Hyperebene, wie sie durch das initialisierte Perzeptron definiert sind, in das Koordinatensystem. (Sie können die Klasse 1 mit einem Pluszeichen und 0 mit einem Minuszeichen im Graph eintragen).



Antwort



(c2) Erklären Sie das Bild in Bezug auf die Klassifikation und die Fehlklassifizierungsrate [*misclassification rate*].

Antwort

(It's important to check the direction of the normal vector)
All points above are classified as 1 and all below the hyperplane as 0, so the perceptron misclassifies $\mathbf{x}_1$. (1P)
The misclassification rate is 0.25. (1P)

(c3) Betrachten Sie den Perceptron Training-Algorithmus. Der Punkt $\mathbf{x}_1$ wird ausgewählt. Berechnen Sie einen Aktualisierungsschritt für die Gewichte, wie er durch den Perceptron Training-Algorithmus definiert ist. Nehmen Sie eine Lernrate von $\eta = 0.5$ an.

Antwort

$$\delta = c(\mathbf{x}) - y(\mathbf{x}) = 1 - 0 = 1$$

$$\mathbf{w}_{new} = \mathbf{w} + \eta\delta\mathbf{x}_1 = \begin{pmatrix} -3 \\ -1 \\ 3 \end{pmatrix} + 0.5 \cdot 1 \cdot \begin{pmatrix} 1 \\ 2 \\ 1 \end{pmatrix} = \begin{pmatrix} -2.5 \\ 0 \\ 3.5 \end{pmatrix}$$

$$\mathbf{w}_{new} = \begin{pmatrix} -2.5 \\ 0 \\ 3.5 \end{pmatrix}$$

(c4) Zeichnen Sie die aktualisierte Hyperebene in das Koordinatensystem von Teilaufgabe 1 ein und interpretieren Sie, was sich in Bezug auf die Klassifikation und Fehlklassifikationsrate [*misclassification rate*] geändert hat.

Antwort

Drawing: see a) (1P)

The model now correctly classifies $\mathbf{x}_1$ but missclassifies $\mathbf{x}_2$. Overall misclassification rate stays 0.25.