

# Maschinelles Lernen I Lernen nach Bayes (Topic 8)

## Was ist Lernen nach Bayes?

Ein statistisches Verfahren, das erlaubt:

- Kombinieren von vorhandenes Wissen (*a priori* Wahrsch.) mit beobachteten Daten
  - Hypothesen können mit einer Wahrscheinlichkeit angegeben werden
  - es werden keine bestehenden Hypothesen ausgeschlossen  
=> denn jeder Bsp. kann Glaubwürdigkeit bestehender Hyp. verringern oder erhöhen
  - mehrere mögliche Hypothesen können gemeinsam ausgewertet werden  
=> zielgerichteter Arbeiten

## Probleme: in der Praxis:

- 1) initiales Wissen über Wahrsch. notwendig (Approx. durch Schätzung auf Basis vorhandener Daten)
  - 2) Rechenaufwand für optimale Bayes'sche Hypothese im Allgemeinen
    - linear mit Anzahl möglicher Hypothesen
    - Aber: manchmal deutliche Reduzierung d. Rechenaufwands möglich

(Rechenregeln siehe Cheat Sheet)

## Theorem von Bayes

$P(h)$  Wahrsch., dass  $h$  aus  $H$  gültig (a priori, d.h. vor Beobachtung von  $D$ )

$P(D)$  Wahrsh., dass  $D$  als Ereignisdatensatz auftritt (ohne Wissen über mögliche Hypothesen)

(likelihood)  $P(D|H)$  Wahrsch., dass  $D$  auftritt in einer Welt wo  $H$  gilt. gute Hypothese

$P(h|D)$  Wahrsc., dass  $h$  wahr ist gegeben die beobachteten Daten  $D$  (a posteriori)

$$P(h|D) = \frac{P(D|h) P(h)}{P(D)} \leftarrow \text{a priori}$$

↑  
a posteriori

Likelihood

Hesler

$$\text{Herleitung (Baye): } P(A \wedge B) = P(B \wedge A)$$

$$P(B|A)P(A) = P(A|B)P(B)$$

$$P(B|A) = \frac{P(A|B)P(B)}{P(A)}$$

Beispiel: Medizinische Diagnose aus Labortest:

$$P(\text{Krebs}) = 0.008 \quad P(\neg \text{Krebs}) = 0.992 \quad P(\oplus) = P(\oplus | \text{Krebs}) \cdot P(\text{Krebs}) + \\ P(\oplus | \neg \text{Krebs}) \cdot P(\neg \text{Krebs})$$

(Falsch-Positiv-R)  $P(\oplus | \neg \text{Krebs}) = 0.03$        $P(\ominus | \neg \text{Krebs}) = 0.97$

$$\text{Gesucht: } P(\text{Krebs} | \oplus) \Rightarrow P(K. | \oplus) = \frac{P(\oplus | \text{Krebs}) \cdot P(\text{Krebs})}{P(\oplus)} = 0,121$$

## Auswahl von Hypothesen

Ziel: finden der Hyp.  $h$  aus  $H$  mit größter w. gegeben die beobachteten Daten  $D$ .

$\rightarrow$  Maximum a posteriori Hypothese (MAP)

$$h_{MAP} = \arg \max_{h \in H} P(h|D) = \arg \max_{h \in H} \frac{P(D|h) P(h)}{P(D)} = \arg \max_{h \in H} P(D|h) P(h) \quad (\text{Bayes}) \quad (P(D) = \text{const.})$$

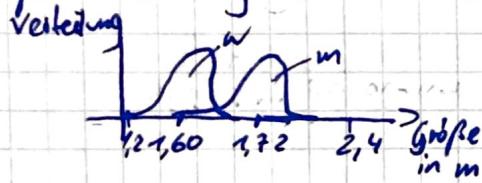
Unter der Annahme, dass untersuchte Hypothesen  $P(h_i)$  (z.B.  $P(\text{Krebs})$  oder  $P(\neg \text{Krebs})$ ) alle gleichverteilt sind oder sie einfach ignoriert werden (unregularisiert)\* wird MAP zur Maximum Likelihood Estimation (MLE):

(2)

$$h_{\text{MLE}} = \arg \max_{h \in H} P(D|h)$$

( $P(D)$  ist hier noch immer konstant!  
=> kann als Normalisierer gesehen werden)

\* Beispiel: Verteilung männlich/weiblich (vereinfacht ohne std. Dev.)



$$\text{MAP} = \arg \max_{\mu_1, \mu_2} P(\mu_1, \mu_2 | D)$$

$$= \arg \max_{\mu_1, \mu_2} \frac{P(D|\mu_1, \mu_2) \cdot P(\mu_1, \mu_2)}{\underbrace{\mu_1, \mu_2}_{\text{likelihood}}} \quad \xrightarrow{\text{P}(D) \text{ const.}} \quad \begin{matrix} \text{empirisch} \\ \text{a priori} \end{matrix}$$

Das a priori - Wissen ( $P(\mu_1, \mu_2)$ ) kann als Regularisierung betrachtet werden, weil damit sehr große und sehr kleine Menschen schwach gewichtet werden (da sie in der Annahme einer Normalverteilung sehr unwahrscheinlich sind)

=> Senkung der Varianz aber Erhöhung von Bias

=>  $P(D|\mu_1, \mu_2)$  = erwartungstreuer Schätzer = unbiased

(> Definition besagt, wenn Anzahl samples ( $|D|$ )  $\rightarrow \infty$  geht, dann konvergiert Erwartungswert gegen wahren EW der zugrundeliegenden Wahrscheinlichkeitsverteilung)

"Biased" wäre der Schätzer, wenn er gegen einen davon abweichenden EW konvergiert

Beispiel: medizinische Diagnose mit Vorwissen aus alten Labortests, nun neuer Patient mit positivem Labortest:

$$P(\oplus | \text{Krebs}) P(\text{Krebs}) = 0.98 \cdot 0.008 = 0.0078$$

$$P(\oplus | \neg \text{Krebs}) P(\neg \text{Krebs}) = 0.03 \cdot 0.992 = 0.0298$$

=> bei einem positiven Labortest hat die Hypothese „nicht Krebs“ die höhere Wahrscheinlichkeit -> sollte gewählt werden nach MAP

### Brute Force Lernen von MAP-Hypothesen

1) Berechne für jede Hypothese  $h \in H$  die a posteriori Wahrsch.:

$$P(h|D) = \frac{P(D|h) \cdot P(h)}{P(D)} \quad (\leftarrow \text{kann weggelassen werden wenn const.})$$

2) Gib die Hypothese  $h_{\text{MAP}}$  mit grösster a posteriori Wahrsch. aus

$$h_{\text{MAP}} = \arg \max_{h \in H} P(D|h) P(h)$$

# 3 Maschinelles Lernen Lernen nach Bayes (Topic 8)

## Konzeptlernen

### Problemstellung:

- endlicher Hypothesenraum  $H$  über Raum der Instanzen  $X$
- Aufgabe: Lernen eines Zielkonzepts  $c: X \rightarrow \{0,1\}$
- Feste Sequenz von Instanzen:  $\langle x_1, \dots, x_m \rangle = X$
- Sequenz der Zielwerte:  $D = \langle d_1, \dots, d_m \rangle$  (Klassenlabels)

### Annahmen:

- Trainingsdaten  $D$  sind nicht verarscht (d.h.  $d_i = c(x_i)$ )
- Zielkonzept  $c$  ist in  $H$  enthalten
- keine a priori - Wissen (alle Hyp. gleich wahrscheinlich)

### Fälle:

$$\text{(likelihood)} \quad P(CD|h) = \begin{cases} 1 & \text{falls } h(x_i) = d_i, \forall i: \in D \\ 0 & \text{sonst} \end{cases}$$

$$P(h) = \frac{1}{|H|} \quad (\text{gleichverteilt})$$

### Berechnung der a posteriori Wahrsch.:

#### 1. Fall (konsistente Hyp.)

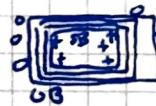
$$P(h|D) = \frac{1 \cdot \frac{1}{|H|}}{P(D)} = \frac{1 \cdot \frac{1}{|H|}}{\sum_{h \text{ const}} P(CD|h) P(h)} = \frac{1 \cdot \frac{1}{|H|}}{\frac{|VS_{H,D}|}{|H|}} = \frac{1}{|VS_{H,D}|}$$

#### 2. Fall (sonst):

$$P(h|D) = \frac{0 \cdot P(h)}{P(D)} = 0$$

### Einschub:

VS<sub>H,D</sub>: Menge der  $h$  aus  $H$ , die konsistent mit  $D$  sind (Versionenraum von  $H$ )



Rechtecke sind die Hyp. im Version Space  
GB: general boundary  
SB: specific boundary

$$\text{damit: } P(D) = \sum_{h \text{ const}} P(d_i|h) \cdot P(h) = \frac{|VS_{H,D}|}{|H|}$$

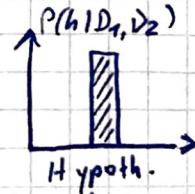
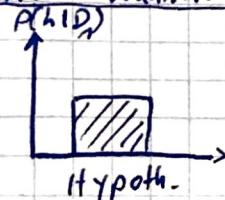
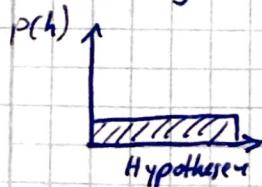
$$\text{also: } P(h|D) = \begin{cases} \frac{1}{|VS_{H,D}|} & \text{falls } h \text{ konsistent mit } D \\ 0 & \text{sonst} \end{cases}$$

- Ein Lernverfahren ist ein konsistenter Lerner, wenn es eine Hypothese liefert, die keine Fehler auf den Trainingsdaten macht

- Unter obigen Voraussetzungen gibt jeder konsistente Lerner eine MAP-Hypothese aus (learning bias)

- Methode um induktiven Bias auszudrücken (IB: Menge der Annahmen die der Lerner nutzt um Outputs zu Inputs zu schätzen  $\Rightarrow$  Version space)

### Entwicklung der a posteriori Wahrsch. mit wachsender Anzahl Trainingdaten:

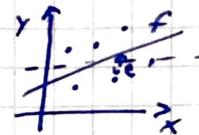


Inkonsistente Hypothesen  $P \rightarrow 0$

## Lernen einer reellwertigen Funktion

Gesucht: reellwertige Zielfunktion  $f$

Gegeben: Beispiele  $(x_i, d_i)$  mit vorausgesetzten Trainingswerten für  $d_i$ :



$$- d_i = f(x_i) + e_i$$

- $e_i$  ist eine Zufallsvariable (Rauschen) die unabhängig für alle  $x_i$  entsprechend einer Normalverteilung mit Mittelwert  $\mu=0$  gezogen wird

Die Maximum Likelihood Hypothese  $h_{ML}$  ist diejenige, welche die Summe der Fehlerquadrate minimiert:

$$h_{ML} = \arg \min_{h \in H} \sum_{i=1}^m (d_i - h(x_i))^2$$

(Herleitung siehe Folien S. 19)

## Klassifikation neuer Instanzen

Bisher: Such nach der Hypothese mit der größten Wahrsch. gegeben die Daten D

Jetzt: Welches ist die wahrscheinlichste Klassifikation  $v_j$  einer neuen Instanz?

Beispiel:  $P(h_1 | D) = 0.4, P(h_2 | D) = 0.3, P(h_3 | D) = 0.3$

$$h_1(x) = \Theta, h_2(x) = \Theta, h_3(x) = \Theta \quad (\text{wahr: } \Theta)$$

$\Rightarrow h_{MAP}(x)$  ist nicht die wahrscheinlichste Klassifikation!

$\rightarrow$  ist ein Punktschätzer einer unbekannten Quantität auf Basis von empirischer Daten

$\rightarrow$  damit: wie wahrsch. ist  $h$  wahr, gegeben  $D \Rightarrow$  notwendig: multiplizieren mit Beispieldaten und aufaddieren um das wahrscheinlichste Label zu bestimmen!  $\Rightarrow$

## Optimale Klassifikation nach Bayes

$$v_{OB} = \arg \max_{v_j \in V} \sum_{h_i \in H} P(v_j | h_i) P(h_i | D)$$

Beispiel:

$$P(h_1 | D) = 0.4, P(\Theta | h_1) = 0, P(\oplus | h_1) = 1$$

$$P(h_2 | D) = 0.3, P(\Theta | h_2) = 1, P(\oplus | h_2) = 0$$

$$P(h_3 | D) = 0.3, P(\Theta | h_3) = 1, P(\oplus | h_3) = 0$$

$$\sum_{h_i \in H} P(\oplus | h_i) P(h_i | D) = 1 \cdot 0.4 + 0 \cdot 0.3 + 0 \cdot 0.3 = 0.4 \quad (\text{Erwartungswert})$$

$$\sum_{h_i \in H} P(\Theta | h_i) P(h_i | D) = \underline{0.6}$$

Vorteil: schneidet im Durchschnitt als bestes Klassifikationsverfahren ab

Nachteil: kostenintensiv bei großer Hypothesenzahl (weil a-posteriori Hyp. für alle Hyp. berechnet)

## Gibbs Algorithmus

- wähle  $h$  aus  $H$  zufällig nach der a-posteriori Wahrscheinlichkeitsverteilung  $P(h | D)$
- Nutze  $h(x)$  für Klassifikation von  $x$
- Bestimme den Erwartungswert wie vorher
- Vorteil: weniger kostenintensiv und Eigenschaft (unter bestimmten Annahmen):

$$E[\text{error}_{\text{Gibbs}}] \leq 2 E[\text{error}_{\text{Baysoptimal}}]$$

Prof. Dillmann fragen warum Ensemble Learning (S.25)

# (5) Maschinelles Lernen 1 Lernen nach Bayes (Topic 8)

## Naiver Bayes Klassifikator

adresiert das Problem der Abhängigkeit in den Trainingsdaten durch Vereinfachung mithilfe der bedingten Unabhängigkeit

### Beispiel

Gegeben: - Instanz  $x$ : Konjunktion vieler Attribute  $\langle a_1, a_2, \dots, a_n \rangle$

- Endliche Menge von Klassen  $V = \{v_1, \dots, v_m\}$

- Menge klassifizierter Beispiele

Gesucht: Wahrscheinlichste Klasse für neue Instanz

$$v_{MAP} = \arg \max_{v_j \in V} P(v_j | a_1, a_2, \dots, a_n)$$

$$= \arg \max_{v_j \in V} \frac{P(a_1, a_2, \dots, a_n | v_j) P(v_j)}{P(a_1, a_2, \dots, a_n)}$$

$$= \arg \max_{v_j \in V} P(a_1, a_2, \dots, a_n | v_j) P(v_j)$$

sicher zu berechnen:

Auszählen aller

Kombinationen über

Attributwerte  $\rightarrow$  riesige

Trainingsmenge notwendig!

a priori: lässt sich leicht aus dem Auftreten der Klasse  $v_i$  in der Trainingsmenge berechnen

Daher vereinfachende Annahme, dass  $a_i$  bedingt unabhängig \*

$$P(a_1, a_2, \dots, a_n | v_j) = \prod_i P(a_i | v_j)$$

Und damit naiver Bayes Klassifikator:

$$v_{NB} = \arg \max_{v_j \in V} P(v_j) \prod_i P(a_i | v_j)$$

z.B.  $P(\text{Wind=stark} | \text{Tennis=ja})$   
 $P(\text{Wind=stark} | \text{Tennis=nein})$

### \* Einschub: Bedingte Unabhängigkeit

$X$  ist bedingt unabhängig von  $Y$  gegeben  $Z$ , wenn die Wahrscheinlichkeitsverteilung von  $X$  bei gegebenem Wert von  $Z$  unabhängig vom Wert von  $Y$  ist:

$$P(X | Y, Z) = P(X | Z)$$

Beispiel:  $P(\text{Donner} | \text{Regen}, \text{Blitz}) = P(\text{Donner} | \text{Blitz})$

### Zusammenfassung

-  $P(v_j)$  und  $P(a_i | v_j)$  werden basierend auf Häufigkeiten in Trainingsdaten geschätzt (gezählt)

- Wahrscheinlichkeiten für Klassifikation entspricht gelernter Hypothese
- Neue Instanzen werden klassifiziert unter Anwendung obiger MAP-Regel
- Wenn Annahme (bedingte Unabhängigkeit der Attribute) erfüllt

$$v_{NB} = v_{MAP}$$

=> Wichtig: keine Suche im Hypothesenraum!

Beispiel (S.30): neue Instanz  $\langle \text{sonnig}, \text{haut}, \text{hoch}, \text{stark} \rangle$

$$P(\text{Tennis} = \text{ja}) = \frac{9}{14} \quad P(\text{Wind=stark} | \text{Tennis} = \text{ja}) = \frac{3}{9}$$

$$P(\text{Tennis} = \text{nein}) = \frac{5}{14} \quad P(\text{Wind=stark} | \text{Tennis} = \text{nein}) = \frac{3}{5}$$

$$P(v_j) = P(\text{sonnig} | \text{ja}) P(\text{haut} | \text{ja}) P(\text{hoch} | \text{ja}) P(\text{stark} | \text{ja}) = 0.0053$$

$$= 0.0206$$

$$P(v_{nein}) = \dots$$

=> Klassifikation: Tennis=nein

$$\text{normierte Wahrsch. } \frac{0.0206}{0.0206 + 0.0053} = 0.795$$

## Schätzen von Wahrscheinlichkeiten

(6)

Problem: Was, wenn für eine Klasse  $v_j$  ein Attribut  $a_i$  einen bestimmten Wert in den Daten gar nicht annimmt?

$$\hat{P}(a_i | v_j) = 0 \rightarrow P(v_j) \neq \hat{P}(a_i | v_j) = 0$$

Lösung:  $\hat{P}(a_i | v_j) \leftarrow \frac{n_c + m_p}{n + m}$  (m-Laplace Schätzer)

n Anzahl der Beispiele mit  $v = v_j$

$n_c$  Anzahl der Beispiele mit  $v = v_j$  und  $a = a_i$

$p$  A priori Wahrscheinlichkeit für  $\hat{P}(a_i | v_j)$ , z.B.  $p = \frac{1}{\text{Werte}(a_i)}$

m Anzahl der „virtuellen Beispiele“ → neu generierte Daten aus vorhandener W-Verteilung (a priori Wissen)

### Eintritt: virtuelle Beispiele

Annahme: Münzwurf: Heads  $\leftrightarrow$  Tails, 50% Wahrscheinlichkeit

$$H \frac{1}{2}$$

$$T \frac{1}{2}$$

„high variance“  $\rightarrow n_H = 0$

$$n_T = 5$$

m-Schätzer

$$m_H = 5$$

$$m_T = 5$$

$$m_H = \frac{5}{10} \cdot 10 = 5$$

$$m_T = \frac{5}{10} \cdot 10 = 5$$

$$\hat{P}_H = \frac{0+5}{5+5} = \frac{1}{2}$$

$$\hat{P}_T = \frac{5+5}{5+5} = 1$$

- ist die angenommene Wahrscheinlichkeitsverteilung schlecht gewählt  
führt man einen Bias ein (m-Laplace s kann als Regularisator gesehen werden)  
starken
- ist sie gut gewählt führt meinen einen schwächeren Bias ein mit dem Ziel der Varianzreduzierung

„Klassifikation von Texten“ ausgelassen (S. 33 - 36, im Mitchell Buch S. 183)

aber Idee: Verwende m-Laplace-Schätzer, um für alle Wörter des Vokabulars (hier Englisch) einen Schätzwert zu erhalten, für die es keine Sample Daten (aber eine a priori Verteilung meistend) gibt:

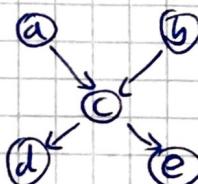
$$P(w_k | v_j) \leftarrow \frac{\text{Anzahl Vorkommen von } w_k \text{ in Text}_j + 1}{\text{Gesamtanzahl Wortpositionen in Text}_j + |\text{Vokabular}|}$$

### Bayes'sche Netze 1

Motivation: Naive Bayes - Annahme der bedingten Unabhängigkeit oft zu restriktiv  
→ ohne vereinfachende Annahmen ist Lernen nach Bayes oft nicht möglich  
Bayes'sche Netze beschreiben bedingte Abhängigkeiten / Unabhängigkeiten bezüglich Untermengen von Variablen

=> erlaubt Kombination von a priori Wissen über bedingte (Un-)Abhängigkeit von Variablen mit den beobachteten Trainingsdaten

Beispiel:



Beragt, dass d von a und b unabhängig ist wenn c gegeben.

Verfahren benötigt lokale Wahrscheinlichkeitstabellen, z.B.

		a,b	a,-b	-a,b	-a,-b	
		c	0.4	0.1	...	
		¬c	0.6	0.9	...	

Inferenz: die Bestimmung von Wahrsh.

tief liegender Zufallsvariablen  $P(z)$  ist

zwar möglich, aber rechenaufwändig (NP-vollst.).

wenn vorherige Wahrscheinlichkeitskombinationen noch unberechnet sind, z.B. müssen für  $P(z)$   $2^n$  (bei binärer Klassifikation) mit n Anzahl Vorgänger berechnet werden

→  $2^n$  Berechnungen, je eine für jede der  $2^n$  mögl. Kombinationen für  $P(z)$   $\hat{P}(z) = P(z|x,y) P(z|\bar{x},y)$   
wenn mehr als eine Kante als Eingang, dann sonst wird nur  $P(z|x)$  berechnet ( $1 - P(z|x)$  braucht nicht gespeichert zu werden)