Grundprinzipien des Bayes'schen Lernens

und

Der naive Bayes-Klassifikator

im Vergleich zum

Maximum-Likelihood-Klassifikator

von

Andreas Schätzle

Inhalt

2.:Wahrscheinlichste Klassifikation

- Funktionsweise der Bayes' Klassifikation
- Der optimale Bayes-Klassifikator (BOC)
- Beispiel zur intuitiven Beschreibung
- Nachteile

Praktische Abwandlungen des optimalen Bayes Klassifikators

- Der Gibbs Algorithmus

Der naive Bayes-Klassifikator/naiver Bayes' Lerner

- Anforderungen
- Aufgaben
- Beispiel: Tennisspieler

Verbesserungen

- m-estimate of probability

Textklassifikation

Der Maximum-Likelihood-Klassifikator(MLK)

- Klassifikation Mittels MLK

Literaturangaben

Bayes’sches Lernen

Bayes’sches Lernen geht von der Annahme aus, dass verschiedene Attribute von Klassen durch ihre gemeinsame Warhscheinlichkeitsverteilungsfunktion zusammenhängen und unterschiedliche Klassen sich in ihrer Warhscheinlichkeitsverteilungsfunktion unterscheiden, und dass folglich die besten Klassifizierungsentscheidungen durch Betrachten dieser Wahrscheinlichkeiten unter Einbeziehung von beobachteten Daten zu treffen sind.

Mittels Bayer’schem Lernen können Beweise für verschiedene Hypothesen gewichtet werden. Des Weiteren ist es die Grundlage für Lernalgorithmen, die auf Wahrscheinlichkeiten operieren und kann sogar zur Analyse von Algorithmen herangezogen werden, die nicht auf Wahrscheinlichkeiten operieren.

- Bayes’sche Lernalgorithmen, welche auf Wahrscheinlichkeiten operieren, sind für bestimmte Anwendungen die effektivsten bekannten Verfahren. z.B. der naive Bayes Klassifikator als Text-Klasssifikator.

- Die Bayes’sche Analyse gibt an, unter welchen Umständen „nicht Wahrscheinlichkeits-Algorithmen” die wahrscheinlichste Hypothese für Trainingsdaten ausgeben.

Eigenschaften Bayes’scher Lernmethoden:

- Jedes Trainingsbeispiel kann die Wahrscheinlichkeit für die Korrektheit einer Hypothese schrittweise beeinflussen. Somit sind Bayes’sche Lernmethoden flexibler als Methoden, die Hypothesen eliminieren, die für ein Beispiel nicht stimmen.

- Hintergrundwissen kann mit beobachteten Daten kombiniert werden, um die Wahrscheinlichkeit für eine Hypothese festzulegen.
- Hintergrundwissen in Bayes’schem Lernen ist 1.die a priori Wahrscheinlichkeit für jede Hypothese 2. eine Wahrscheinlichkeitsverteilung für die beobachteten Daten für jede mögliche Hypothese.

- Bayes’sche Methoden können Hypothesen anpassen, die Wahrscheinlichkeitsvorhersagen angeben. (z.B.: “Der Patient hat ein 93% Genesungschance”)

- Neuklassifizierung geschieht durch die nach Zutreffwahrscheinlichkeit gewichtete kombinierte Voraussage mehrerer Hypothesen.

- Wenn Bayes’sche Algorithmen nicht praktisch anwendbar sind, können sie trotzdem als Standard für die optimale Entscheidungsfindung herangezogen werden, gegen den die verwendeten Algorithmen gemessen werden können.

Schwierigkeiten:

Für die Anwendung Bayes'scher Methoden müssen jedoch im Vorhinein viele Wahrscheinlichkeiten bekannt sein, oder durch Hintergrundwissen geschätzt werden. Und auf Grund der vielen zu berechnenden Wahrscheinlichkeiten, ergibt sich oft ein hoher Rechenaufwand.

Bayes' Theorem

Bedeutung für Machine Learning:

Abbildung 1: Thomas Bayes, engl. Pfarrer und Mathematiker. Vater des Bayes' Theorems [WIKIPEDIA]

Im Maschinenlernen geht es oft darum die Wahrscheinlichste Hypothese aus einer Menge gegebener Hypothesen heraus zu suchen. Durch das Bayes'sche Theorem wird das möglich, indem man damit die Wahrscheinlichkeit für jede einzelne Hypothese *h* aus einer Hypothesenmenge *H* berechnen kann.

Allgemeine Herleitung des Bayes Theorems:

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

Hier verwendete Notation:

Im Folgenden wird eine für die Fragestellungen passende Notation gewählt.

Die Wahrscheinlichkeit für das Zutreffen einer Hypothese *h*, vor betrachten der Trainingsdaten (Hintergrundwissen) heißt im Folgenden die „a priori“ Wahrscheinlichkeit P(*h*). Wenn kein Hintergrundwissen vorhanden ist, dann wird der gleiche Wert für alle $h \in H$ gesetzt. Des Weiteren ist P(*D*) die Wahrscheinlichkeit, für das Beobachten, der Trainingsdaten. Und P(*D* | *h*) lies: „*D* gegeben *h*”, ist die bedingte Wahrscheinlichkeit dass D beobachtet wird, wenn *h* zutrifft. P(*h* | *D*) lies: “*h* gegeben *D*” ist folglich die bedingte Wahrscheinlichkeit, dass *h* für die Trainingsdaten zutrifft.

Das Bayes’sche Theorem ermöglicht nun die bedingte Wahrscheinlichkeit P(*h* | *D*) (auch „a posteriori“ Wahrscheinlichkeit) mittels P(*h*), P(*D*) und P(*D* | *h*) zu berechnen.

Also ergibt sich mit der eingeführten Notation für das Bayes’ Theorem:

$$P(h \mid D) = \frac{P(D \mid h)\,P(h)}{P(D)} \quad (1)$$

Anwendung:

In der praktischen Anwendung stellen sich zwei grundsätzliche Fragestellungen, die man mit dem Bayes’ Theorem klären kann:

1. Was ist die Wahrscheinlichste Hypothese *h* aus einer Hypothesenmenge *H*, die eine Menge an Trainingsdaten erklärt?

2. Was ist die wahrscheinlichste Klassifikation einer neuen Instanz bei bekannten Trainingsdaten?

zur 1. Fragestellung:

Was ist die Wahrscheinlichste Hypothese *h* aus einer Hypothesenmenge *H*, die eine Menge an Trainingsdaten erklärt?

MAP-Hypothese:

Dazu werden über das Bayes Theorem die „a posteriori“ Wahrscheinlichkeiten für alle $h \in H$ berechnet. Die Hypothese, die dabei die größte a posteriori Wahrscheinlichkeit liefert, wird als maximum a posteriori Hypothese kurz MAP-Hypothese (h_{MAP}) bezeichnet.

$$h_{MAP} = \underset{h \in H}{argmax}\ P(h \mid D)$$

Mit dem Bayes Theorem:

$$= \underset{h \in H}{argmax}\ \frac{P(D \mid h)P(h)}{P(D)}$$

weglassen von P(*D*), da es eine konstante unabhängig von *h* ist:

$$= \underset{h \in H}{argmax}\ P(D \mid h)P(h) \qquad (2)$$

ML-Hypothese:

Wird $P(h_i) = P(h_j)$ für alle $h \in H$ gesetzt, z.B. wenn kein Vorwissen über die Wahrscheinlichkeitsverteilung der Hypothesen bekannt ist, muss nur noch $P(D \mid h)$ betrachtet werden, um die Wahrscheinlichste Hypothese zu finden. Die so, ohne Vorwissen, gefundene Hypothese ist die Maximum-Likelihood-Hypothese, oder kurz: ML-Hypothese, mit der Formel.

$$h_{ML} = \underset{h \in H}{argmax}\ P(D \mid h) \qquad (3)$$

Die Formel verdeutlicht noch einmal, dass die ML Hypothese im Gegensatz zur MAP Hypothese kein Vorwissen, über die Gesamtauftrittswahrscheinlichkeit der Hypothesen verarbeitet.

Bis jetzt bezeichnete *H* eine Hypothesenmenge und *D* eine Menge an Trainingsdaten.

Im Folgenden sei *H* sei eine Aussagenmenge, deren Gesammtwahrscheinlichkeit zu 1 summiert.

Beispielsweise: *H* = „Der Himmel ist Blau.", „Der Himmel ist nicht Blau."

Und *D* sei eine Menge anderer Daten.

Ein Beispiel: Der Krebspatient

Es haben 0,8 % der Bevölkerung Krebs. Des Weiteren ist der Test auf diese Krankheit nicht perfekt. Durch den Test wird Krebs wird nur mit 98%iger Sicherheit erkannt, und das Testergebnis „nicht Krebs" bedeutet nur einen 97%igen Ausschluss der Krankheit.

Der Hypothesenraum *H* = („Krebs" | „nicht Krebs") summiert offensichtlich mit seiner Gesammtwahrscheinlichkeit zu P(*H*) 1.

+ bezeichnet einen positives Testergebnis.

- bezeichnet einen negatives Testergebnis.

Die Wahrscheinlichkeiten zusammengefasst:

P(„Krebs") = 0,008 P(„nicht Krebs") = 0,992

P(+ | „Krebs") = 0,98 P(- | „Krebs") = 0,02

P(+ | „nicht Krebs") = 0,03 P(- | „nicht Krebs") = 0,97

Für einen Patienten mit positivem Testergebnis ergeben sich nach **Formel 2** folgende Wahrscheinlichkeiten für die jew. Hypothesen Krebs , „nicht Krebs"

P(+ | „Krebs")·P(„Krebs") = (0,98) · (0,008) = 0,0078

P(+ | „nicht Krebs")·P(„nicht Krebs") = (0,03) · (0,992) = 0,0298 -> **MAP-HYPOTHESE**

Normalisierung:

Um die korrekte bedingte Wahrscheinlichkeit zu erhalten wird, eine Normalisierung der Wahrscheinlichkeiten, über den Hypothesenraum vorgenommen. Da der Hypothesenraum zu P(*H*)=1 summiert ist diese Aktion legal und es gilt:

P („nicht Krebs“ | +) = 0,0298 / (0,0078 + 0,0298) = 0,792 = 79,2 %

P („Krebs“ | +) = 0,0078 / (0,0078 + 0,0298) = 0,208 = 20,8 %

Also ist die MAP-Hypothese trotz positivem Test „nicht Krebs“, mit einer Wahrscheinlichkeit von 79,2%.

Analyse:

- Die MAP Hypothese hängt stark von der a priori Wahrscheinlichkeit P(*h*) ab.
- Die Hypothesen werden nicht akzeptiert oder verworfen. Sie werden nur mit jedem Datensatz inkrementell mehr oder weniger wahrscheinlich.

Somit ergibt sich eine Sicherung der Ergebnisse durch mehr Daten. Im Beispiel sollten deswegen noch weitere Tests durchgeführt werden, um die MAP-Hypothese zu sichern, oder zu widerlegen.

Konzept-Lern-Algorithmen:

Hierbei beschreibt der Hypothesenraum H eine Menge an möglichen Konzepten, die auf Stimmigkeit mit Trainingsdaten geprüft werden, wodurch das richtige Konzept $c \in H$ herausgefiltert wird, welches angewendet auf die Eingabewerte der Trainingsdaten, die richtigen Zielwerte berechnet.

Brute-Force Bayes’ Konzeptlernen:

Es sei:

X = $((x_1,d_1), ...(x_n, d_n))$ eine Trainingsmenge, mit x_i = Eingabewert, d_i = zu berechnender Zielwert.

c: X $\rightarrow$ {0,1} ein zu lernendes Zielkonzept

also : c(x_i) = d_i

c könnte zum Beispiel eine Funktion auf den reellen Zahlen sein, die durch Daten bestimmt werden soll.

Ein Brute-Force Algorithmus:

Seien die $h \in H$ mögliche Konzepte, die die Trainingsdaten erklären sollen. Dann könnte ein Brute-Force Algorithmus wie folgt aussehen:

1. Berechne die a posteriori Wahrscheinlichkeiten für alle $h \in H$:

$$P(h \mid D) = \frac{P(D \mid h)\,P(h)}{P(D)}$$

2. Gebe h_{MAP} aus:

$$h_{MAP} = \underset{h \in H}{argmax}\ P(h \mid D)$$

Solch ein Algorithmus kann, auch wenn er selbst auf Grund des hohen Rechenaufwandes unpraktisch ist,, als Standard herangezogen werden, gegen den andere Algorithmen gemessen werden können.

Annahmen für den Algorithmus:

1. Fehlerlose Trainingsdaten D
2. $c \in H$, also das Zielkonzept im zu untersuchenden Raum H enthalten.
3. a priori sind alle $h \in H$ gleich wahrscheinlich

Wahl der Parameter:

P(h) = 1/|H| für alle $h \in H$, da eine a priori Gleichverteilung der $h \in H$ angenommen ist.

Da störungsfreie Daten vorausgesetzt werden, wird gesetzt:

$$P(D \mid h) = \begin{cases} 1 & \text{wenn } d_i = h(x_i) \text{ für alle } d_i \text{ in } d \\ 0 & \text{sonst} \end{cases}$$

Damit liefert der Algorithmus:

1. Für ein inkonsistentes Konzept:

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

2. Für ein konsistentes Konzept:

$$P(h \mid D) = \frac{1 \cdot \frac{1}{|H|}}{P(D)}$$

$$= \frac{1 \cdot \frac{1}{|H|}}{\frac{|VS_{H,D}|}{|H|}}$$

$$= \frac{1}{|VS_{H,D}|}$$

$VS_{H,D}$ sind hierbei die mit *D* konsistenten Hypothesen.

$P(D) = \frac{|VS_{H,D}|}{|H|}$ ergibt sich, da die Summe der Wahrscheinlichkeiten über alle $h \in H$ P(H)=1 sein muss und $|VS_{H,D}|$ per Definition die Anzahl mit *D* konsistenten Hypothesen ist.

Da alle konsistenten Hypothesen die gleiche a posteriori Wahrscheinlichkeit haben und alle inkonsistenten Hypothesen die a posteriori Wahrscheinlichkeit 0 haben, ist jede konsistente Hypothese eine MAP-Hypothese.

Zusammengefasst:

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

Konsistente Lerner:

Eine Hypothese, die keine Fehler auf den Trainingsdaten macht, heißt konsistent. Verfahren, die nur solche Hypothesen ausgeben, heißen konsistente Lerner. Sind $P(h_i) = P(h_j)$ für alle h_i, h_j und fehlerlose Trainingsdaten gegeben, so gibt ein konsistenter Lerner immer eine MAP-Hypothese aus.

Dies gilt auch für Algorithmen, die nicht auf Wahrscheinlichkeiten operieren.

Beispiel: FindS

Der Algorithmus FindS vergleicht die Zielwerte, die die Hypothesen für Trainingswerte ausgeben, mit den tatsächlichen Zielwerten und gibt die konsistenteste Hypothese aus, die also, die am längsten keine Fehler macht. Dabei operiert FindS nicht auf Wahrscheinlichkeiten.

Bayes'sche Analyse:

Da FindS eine konsistente Hypothese ausgibt, muss diese unter den gemachten Annahmen für $P(h)$ und $P(D \mid h)$ eine MAP-Hypothese sein. Durch weiteres Betrachten der Struktur von FindS kann die Aussage gemacht werden, dass FindS für jede Wahrscheinlichkeitsverteilung eine MAP-Hypothese ausgibt, die $P(h_1) \geq P(h_2)$ zuordnet, wenn h_1 konsistenter als h_2 gilt.

Die Bayes'sche Analyse basiert darauf, die induktive Folgerung des Algorithmus durch eine äquivalente Wahrscheinlichkeitsargumentation zu ersetzen, wobei die impliziten Annahmen über den Lerner durch $P(h)$ beschrieben werden und die Stärke, mit der eine Hypothese akzeptiert oder abgelehnt wird durch $P(D \mid h)$ ausgedrückt wird.

Kurz: Eine so auf dem Bayes' Theorem fußende Analyse bildet das Input/Output-Verhalten des Algorithmus nach. Das erlaubt es, Aussagen darüber zu machen, wie der Algorithmus unter bestimmten Bedingungen funktioniert.

Zur 2. Fragestellung:

Was ist die wahrscheinlichste Klassifikation einer neuen Instanz bei bekannten Trainingsdaten?

Allgemeines:

Eine neue Instanz/ein neuer Merkmalsvektor **x** soll klassifiziert werden.

Zuerst wird ihm durch die **Unterscheidungsfunktion d** eine Wahrscheinlichkeit für in Klassifikation in jede der möglichen Klassen zugeordnet.

Die **Entscheidungsregel e** gibt dann an, wie auf Grund dieser Wahrscheinlichkeit klassifiziert werden soll.
Die hier verwendeten Entscheidungsregeln funktionieren nach dem Prinzip, einer neuen Instanz die Klasse zuzuordnen, für die die nach der Unterscheidungsfunktion **d** berechnete Wahrscheinlichkeit für diese Klasse am größten ist.
Bei der Klassifikation wird die Entscheidungsregel **e** auf die Ergebnisse Unterscheidungsfunktion **d** der neuen Instanz angewendet.
Die Funktion **g =e(d(x))**, die sich dadurch ergibt, heißt Klassifikationsfunktion.
Mit ihr wird eine neue Instanz klassifiziert.

Funktionsweise der Bayes-Klassifikation:

Bayes'sche Klassifikationsverfahren gehören zu den überwachten Klassifikatoren, da sie erst durch Trainingsdaten mit bekannter Klassifikation trainiert werden und dann auf neue Instanzen angewendet werden können.

Der optimale Bayes' Klassifikator (BOC)/der optimale Bayes-Lerner

Der BOC klassifiziert nicht einfach auf Grund der MAP-Hypothese, sondern maximiert die Chance auf richtige Klassifikation, indem er eine „gewichtete Abstimmung" unter den Hypothesen $h \in H$ vornimmt.

Ein Beispiel zur intuitiven Beschreibung der Funktionsweise:

Unser Hypothesenraum bestehe aus 3 Hypothesen mit a posteriori Wahrscheinlichkeiten bezüglich der Trainingsdaten wie folgt:

$P(h_1 \mid D) = 0{,}4 \rightarrow$ MAP-Hypothese

$P(h_2 \mid D) = 0{,}3$

$P(h_3 \mid D) = 0{,}3$

Nun wird die neue Instanz von allen 3 Hypothesen klassifiziert:

h_1 klassifiziert positiv h_2 und h_3 negativ.

Diese Klassifizierungen werden mit den a posteriori Wahrscheinlichkeiten gewichtet summiert und es wird der überwiegenden Wahrscheinlichkeit nach klassifiziert.

V sei die Menge der möglichen Klassifizierungen.

Es gelten folgende Wahrscheinlichkeiten:

$P(h_1 \mid D) = 0{,}4$	$P(- \mid h_1) = 0$	$P(+ \mid h_1) = 1$
$P(h_2 \mid D) = 0{,}3$	$P(- \mid h_2) = 1$	$P(+ \mid h_2) = 0$
$P(h_3 \mid D) = 0{,}3$	$P(- \mid h_3) = 1$	$P(+ \mid h_3) = 0$

Nach dem eben formulierten Prinzip ergibt sich:

$$\sum_{h_i \in H} P(+ \mid h_i)\, P(h_i \mid D) = 0{,}4$$

bzw.:

$$\sum_{h_i \in H} P(- \mid h_i) \cdot P(h_i \mid D) = 0{,}6$$

folglich wird negativ klassifiziert.

Explizite Formel des Optimalen Bayes' Klassifikators:

$$\underset{v_j \in V}{\operatorname{argmax}} \sum_{h_i \in H} P(v_j \mid h_i)\, P(h_i \mid D) \qquad (4)$$

Bei gleicher a priori Kenntnis und gleicher Hypothesenmenge H ist dies im Durchschnitt der beste bekannte Klassifikator. Diese Methode maximiert die Chance auf korrekte Klassifikation bei gegebenem *H* und a priori Wahrscheinlichkeiten.

Aus dieser Vorgehensweise ergibt sich, dass der Bayes Klassifikator zu gleichen Ergebnissen kommen kann, wie eine Hypothese, die gar nicht in H enthalten ist.

Ein Erklärungsansatz dafür ist, dass der optimale Bayes' Klassifikator nicht den Hypothesenbereich *H* betrachtet, auf den das Bayes' Theorem angewendet wird, sondern einen davon verschiedenen Bereich *H'*, der Hypothesen einbezieht, die Vergleiche zwischen linearen Kombinationen von Hypothesen aus *H* anstellt.

Nachteile:

Obwohl der optimale Bayes Klassifikator die Besten Resultate bei der Klassifikation mit Trainingsdaten erreicht, ist er **sehr rechenintensiv.** Es müssen die a posteriori Wahrscheinlichkeit für jedes $h \in H$ berechnet und dann die Vorraussagen von jeder Hypothese kombiniert werden.

Praktische Abwandlungen des optimalen Bayes Klassifikators

Der Gibbs Algorithmus

1. Wähle zufällig eine Hypothese $h \in H$ aus, unter Berücksichtigung der Wahrscheinlichkeitsverteilung über h.

2. Benutze h, um die nächste Instanz x zu klassifizieren.

Dieses viel weniger rechenintensive Verfahren produziert unter bestimmten Bedingungen im schlechtesten Fall einen maximal doppelt so großen Fehler wie der optimale Bayes' Klassifikator.

Der naive Bayes Klassifikator/naiver Bayes Lerner

Diese Abwandlung des optimalen Bayes Klassifikators ist in manchen Gebieten vergleichbar gut wie Entscheidungsbäume, oder neuronale Netzwerke.

Anforderungen:

- Eine Instanz x ist durch eine Verbindung von Attributwerten gegeben und die Zielfunktion f(x) kann nur Werte aus einer endlichen Menge annehmen.
- Eine Menge von Trainingsdaten für die Zielfunktion wird bereitgestellt
- Eine neue Instanz wird durch ihre Attributwerte (a_1, ... , a_n) dargestellt.

Aufgaben:

Die Aufgaben des naiven Bayes' Klassifikators bestehen in der Vorhersage des Zielwertes und der Klassifikation einer neuen Instanz

Die Ausgabe des wahrscheinlichsten Zielwertes $\mathbf{v}_{\mathbf{MAP}}$ gegeben die Attribute, die die Klasse beschreiben, wird erreicht durch:

$$v_{MAP} = \underset{v_j \in V}{argmax}\ P(v_j \mid a_1, a_2, \ldots a_n)$$

mit V = Menge aller möglichen Klassifikationen v_j

durch das Bayes' Theorem:

$$v_{MAP} = \underset{v_j \in V}{argmax} \frac{P(a_1, a_2, \ldots a_n \mid v_j) P(v_j)}{P(a_1, a_2, \ldots a_n)}$$

$$= \underset{v_j \in V}{argmax} P(a_1, a_2, \ldots a_n \mid v_j) P(v_j) \qquad (6)$$

Eine Schätzung der $P(a_1, \ldots a_n \mid v_j)$ wäre nur für große Trainingsdatenmengen geeignet. Da die Anzahl dieser Terme gleich der Anzahl der möglichen Instanzen mal der Anzahl möglicher Zielwerte ist, müsste jede Instanz der Menge sehr oft ansehen um verlässliche Schätzwerte zu erhalten.

Vereinfachende Annahme:

- Alle Attribute sind stochastisch unabhängig.

Obwohl diese Annahme in der Praxis häufig verletzt wird, liefert der NBK trotzdem gute Ergebnisse für den Fall, dass nur wenige Korrelationen auftreten.

Damit ist die Wahrscheinlichkeit eine Reihe $a_1, a_2, \ldots a_n$ zu beobachten, für einen Zielwert der Instanz einfach das Produkt der Wahrscheinlichkeiten der einzelnen Attribute ist.

$$P(a_1, a_2, \ldots, a_n \mid v_j) = \Pi_i P(a_i \mid v_j)$$

Eingefügt in (6) ergibt dies:

$$v_{NB} = \underset{v_j \in V}{argmax} P(v_j) \prod_i P(a_i \mid v_j) \qquad (7)$$

➔ Die aus den Trainingsdaten abzuschätzenden $P(a_i \mid v_j)$ Terme sind nur die Menge der Attribute multipliziert mit der Menge der Zielwerte. Das ist eine viel kleinere Menge, als die $P(a_1, \ldots, a_n \mid v_j)$.

Zusammenfassung :

1. Lernschritt: Schätzen der $P(v_j)$ und $P(a_i \mid v_j)$ auf Grund ihrer Häufigkeit in den Trainingsdaten. Die Menge dieser gelernten Schätzungen korrespondiert zu der gelernten Hypothese.

2. Diese Hypothese wird dann verwendet, um die neuen Instanzen mittels Formel **(7)** zu klassifizieren.

3. Wenn die Annahme der Unabhängigkeit erfüllt wird, dann ist die Bayes Klassifikation $\mathbf{v}_{NB}$ identisch der MAP-Klassifikation.

4. Im Gegensatz zu anderen besprochenen Methoden gibt es keine explizite Suche durch den Hypothesenbereich. Das wäre hier der Bereich der möglichen Zuweisungen zu den beiden Termen.

5. Stattdessen wird die Hypothese durch zählen der Häufigkeiten der Datenkombinationen in den Trainingsdaten gebildet.

Beispiel: Tennisspieler

Als Beispiel ist eine Trainingsdatenmenge in Tabellenform gegeben und es soll mittels der in Tabelle1 gegebenen Daten ein neuer Tag d dahin gehend klassifiziert werden, ob eine Person wahrscheinlich Tennis spielen wird, oder nicht.

$$v_{NB} = \underset{v_j \in V}{argmax}\, P(v_j) \prod \prod P(a_i \mid v_j)$$

Mit V = (Ja/Nein)

$$v_{NB} = \underset{v_j \in \{Ja,\, Nein\}}{argmax}\, P(Sonnig \mid v_j) P(Kühl \mid v_j) P(Hoch \mid v_j) P(Stark \mid v_j)$$

Tag	*Ausblick*	*Temperatur*	*Luftfeuchtigkeit*	*Wind*	*Tennisspielen*
1	Sonnig	Heiß	Hoch	Schwach	Nein
2	Sonnig	Heiß	Hoch	Stark	Nein
3	Bewölkt	heiß	Hoch	Schwach	Ja
4	Regnerisch	Mild	Hoch	Schwach	Ja
5	Regnerisch	Kühl	Normal	Schwach	Ja
6	Regnerisch	Kühl	Normal	Stark	Nein
7	Bewölkt	Kühl	Normal	Strark	Ja
8	Sonnig	Mild	Hoch	Schwach	Nein
9	Sonnig	Kühl	Normal	Schwach	Ja
10	Regnerisch	Mild	Normal	Schwach	Ja
11	Sonnig	Mild	Normal	Stark	Ja
12	Bewölkt	Mild	Hoch	Stark	Ja
13	Bewölkt	Heiß	Normal	schwach	Ja
14	Regnerisch	Mild	Hoch	Stark	Nein

Tabelle 1: Bsp.. Tennisspieler Trainingsdaten [Mitchell]

Bestimmung der Wahrscheinlichkeiten für die Zielwerte, durch die Tabelle:

P(*Ja*) = 9/14 = 0,64

P(*Nein*) = 5/14 = 0,36

Bestimmung der Bedingten Wahrscheinlichkeiten:

P (*Stark* | Ja) = 3/9 = 0,33

P (*Stark* | *Nein*) = 3/5 = 0,60

analog für alle restlichen benötigten Wahrscheinlichkeiten.

Für die zwei möglichen Klassifikationen „Ja“/„Nein“ ergeben sich folgende Wahrscheinlichkeiten:

P (*Ja*)* P (*Sonnig* | *Ja*) * P (*Kühl*| *Ja*) * P (*Hoch* | *Ja*) *P (*Stark* | *Ja*) = 0,0053

P (*Nein*)* P (*Sonnig* | *Nein*) * P (*Kühl* | *Nein*) * P (*Hoch* | *Nein*) *P (*Stark*| *Nein*) = 0,0206

Also ergibt sich für die naive Bayes’ Klassifizierung:

$$v_{NB} = \underset{v_j \in \{Ja, Nein\}}{argmax}\, P(v_j) \prod \prod P(a_i \mid v_j) = Nein$$

Berechnung der a posteriori Wahrscheinlichkeit durch Normalisierung:

P(*Nein* | a_i) = 0,0206 / (0,0206 + 0,0053) = 0,795 = 79,5 %

Also ist die Klassifikation auf Grund der hohen a posteriori Wahrscheinlichkeit recht sicher.

Verbesserung: Estimating Probabilities:

Bisher wurden die Wahrscheinlichkeiten durch den Quotienten $\mathbf{n_c/n}$ beschrieben, wobei

n gleich der Anzahl aller möglichen Ereignisse ist und n_c gleich der Anzahl ist, wie oft ein bestimmtes Ereignis vorkommt.

Problem :

Ergibt sich für eine Wahrscheinlichkeit ein Wert nahe an der Null-Wahrscheinlichkeit, so muss nach **Formel (7)** mit allen anderen Wahrscheinlichkeiten multipliziert werden. Dies kann die Schätzung dominieren und so die Performance des Verfahrens verschlechtern.

Lösung:

Die *n* tatsächlichen Beobachtungen werden durch *m* “virtuelle Beobachtungen” entsprechend zu einer gewählten Wahrscheinlichkeitsverteilung *p* ergänzt.

m-estimate of propability:

Berechnung der Wahrscheinlichkeiten durch:

$$\frac{n_c + mp}{n + m} \quad (8)$$

mit p = a priori Schätzung der Wahrscheinlichkeit

Ist kein Hintergrundwissen über die Wahrscheinlichkeitsverteilung gegeben, so wird eine Gleichverteilung angenommen und p wird 1/k gewählt, wobei *k* gleich der Anzahl der Möglichen Werte für ein Attribut.

Im Bsp**.:** Zwei Möglichkeiten : Wind <Stark, Schwach> = > p = 0,5

m ist die sog. „equivalent sample size”, die Anzahl der „virtuellen“ Ziehungen, die angibt, wie stark die Schätzung *p* zu gewichten ist.

Die so erreichte Schätzung ist, aus den genannten Gründen, in der Praxis oft erfolgreicher als die Berechnung der tatsächlichen Wahrscheinlichkeiten.

Textklassifikation:

In der Praxis ist das die Hauptaufgabe für den NB-Klassifikator.

Bei einem Performancetest wurde durch sammeln von je 1000 Artikeln aus 20 Newsgroups ein Pool von 20 000 Artikeln erstellt.

Diese sollten nun der richtigen Newsgroup zugeordnet werden. Es wurden 2/3 als Trainingsbeispiele genutzt und dann die Performance über den Rest ermittelt.

Es wurden 89% der Artikel in die korrekte Newsgroup klassifiziert, was unter Klassifikatoren ein außergewöhnlich gutes Ergebnis ist.

Der Maximum Likelihood Klassifikator

Der ML-Klassifikator wird verwendet, wenn kein Hintergrundwissen über die Anteilswahrscheinlichkeiten der Klassen im Untersuchungsgebiet bekannt ist.

In diesem Fall wird davon ausgegangen, dass die Merkmalsvektoren in einer Klasse nach einer Gauß'schen Normalverteilung streuen, da die Gauß'sche Normalverteilung eine gute Approximation für zufällige stochastisch unabhängige Ereignisse darstellt. Es können auch bessere Annahmen über die Wahrscheinlichkeitsverteilung gemacht werden, wenn Kenntnisse über die zu Grunde liegenden Mechanismen der Streuung bekannt sind.

Klassifikation Mittels MLK:

Auch der MLK zählt zu den überwachten Klassifikatoren. Somit muss er auch in einem ersten Schritt mittels Daten mit bekannter Klassifikation trainiert werden.

Bei der eigentlichen Klassifikation werden die Wahrscheinlichkeiten berechnet, dass der zu klassifizierender Merkmalsvektor d in einer Klasse k_i. liegt. Der Merkmalsvektor d wird dann der Klasse k_i zugeordnet, für die die berechnete Wahrscheinlichkeit maximal wird.

Also ergibt sich für die ML-Klassifizierung:

$$Klassifizierung_{ML} = \underset{k_i \in K}{argmax}\, P(d \mid k_i) \quad (9)$$

Dieses Verfahren findet oft in der Bildbearbeitung seine Anwendung. Es wird dort als überwachtes, pixelbasiertes Klassifikationsverfahren bezeichnet, wobei die Daten als Merkmalsvektoren der Pixel gegeben sind. z.B.: Position und Grauwert des Pixels.

Dies ergibt einen Merkmalsraum, der wie in Abbildung3 aussehen könnte.

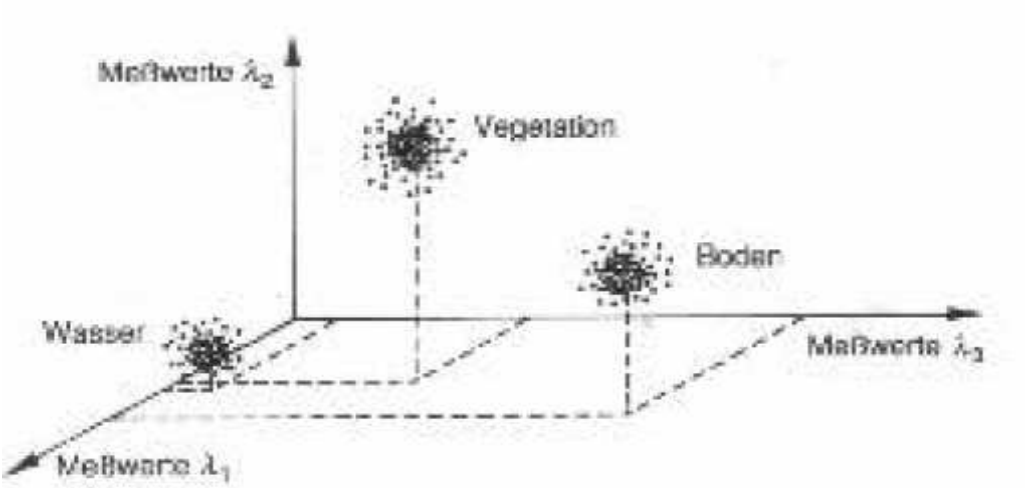

Abbildung 3: 3D Merkmalsraum mit Objekten Vegetation, Boden und Wasser [Albertz 1991]

Dann wird über die Zentren der Musterklasse im Normalfall eine, je nach den Messdaten geformte Gauß'sche Normalverteilung gelegt und die Klasse bestimmt, in der ein neuer Merkmalsvektor (auch als „neue Instanz" bezeichnet) mit größter Wahrscheinlichkeit liegt. Zu dieser Klasse wird der neue Merkmalsvektor dann hinzugefügt.

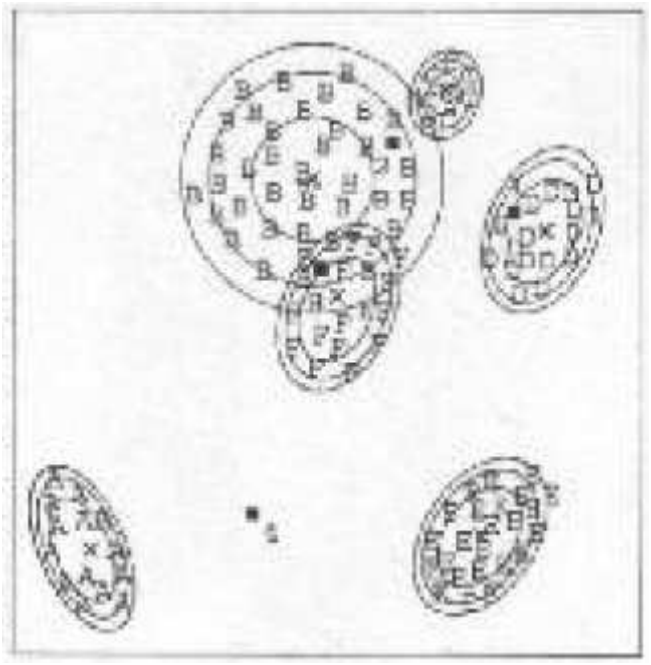

Abbildung 4: Klassifizierungsprinzip des MLK [Hildebrand 1991]

Abbildung4 zeigt die Merkmalsvektoren in der Ebene mit je einer Gauß'schen Normalverteilung mit Maximum im Zentrum der abgebildeten Ellipsen. Die Elemente werden der Klasse zugeordnet, deren Normalverteilung am Ort des Merkmals den größten Wert hat.

Direkter Vergleich zwischen NB-Klassifikator und ML-Klassifikator

Gemeinsamkeiten:

Beide zählen zu den sog. Überwachten Lernern. D.h. sie benötigen beide eine Menge an Trainingsdaten mit bekannter Klassenzugehörigkeit, bevor eine unbekannte Instanz klassifiziert werden kann.

Sie besitzen beide die gleiche Entscheidungsregel e, nämlich Klassifizieren beide eine Instanz in die Klasse, die das Maximum der, durch die Unterscheidungsregel d zugeordneten Wahrscheinlichkeit, beschreibt.

Unterschiede

Sie unterscheiden sich in der Unterscheidungsfunktion d. Während der NB-Klassifikator durch die Berechnung der bedingten Wahrscheinlichkeit $P(k_i \mid d)$ entscheidet, benutzt der ML-Klassifikator $P(d \mid k_i)$ als Entscheidungsregel, wenn keine Wahrscheinlichkeitsverteilung angenommen wird.

Dieser Unterschied ist darin begründet, dass die a priori Anteilswahrscheinlichkeiten für das auftreten einer Klasse beim NB-Klassifikator bekannt und beim ML-Klassifikator unbekannt sind.

Daraus ergeben sich auch unterschiedliche Einsatzgebiete.
Während der NB-Klassifikator ein beliebter Textklassifikator ist, wird der ML-Klassifikator vornehmlich in der Bildverarbeitung/Sprachverarbeitung eingesetzt.

Literaturangaben:

Tom M. Mitchell Machine Learning

Eine sehr gute Zusammenfassung über das gesamte Gebiet des Maschinen Lernens, das aber auf Grund der Größe des abgedeckten Themengebiets kaum mathematische Herleitungen bietet.

Alberts 1991

Als Quelle für Abbildungen benutzt.

Hildebrandt 1991

Als Quelle für Abbildungen benutzt.

Wikipedia

Als Bildquelle verwendet und zusätzlich Liefert es eine grundlegende Übersicht über den gesamten Themenbereich.