Naiver Bayes-Klassifikator für kontinuierliche Merkmale

Prof. Dr. Jörg Frochte

Maschinelles Lernen

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

Wiederholung Naiver Bayes-Klassifikator – unser erster Klassifikator

Wiederholung

- Grundlegende Formel unseres Bayes-Klassifikators:

$$P(i \mid x) = \frac{\prod_{k=1}^{m} P(x^{(k)} \mid i) \cdot P(i)}{\sum_{j=1}^{N} P(j) \cdot \prod_{k=1}^{m} P(x^{(k)} \mid j)} = \frac{P(i) \cdot \prod_{k=1}^{m} P(x^{(k)} \mid i)}{\sum_{j=1}^{N} P(j) \cdot \prod_{k=1}^{m} P(x^{(k)} \mid j)}$$

- x ist der Vektor der Merkmale und i die Klassenzugehörigkeit, die wir vorhersagen wollen. m die Anzahl der Merkmale und N die der Klassen.
- Die Annahme ist, dass Merkmale unabhängig sind. Verletzung der Annahme führt zu schlechteren Ergebnissen.

Acute Inflammations Data Set

- Wir werden uns weitere Begriffe wie u.a.
  - Trainings- und Testmenge
  - Median und Mittelwert
  - Wahrscheinlichkeitsdichtefunktion

  und eine Erweiterung für kontinuierliche Merkmale an einem etwas realistischeren Beispiel klarmachen.
- Es geht um das **Acute Inflammations Data Set** vom Machine Learning Repository.
- In dieser kleinen Datenbank mit 120 Eintragungen und 8 Spalten geht es um ein medizinisches Diagnosesystem.
- Die ersten 6 Spalten sind Merkmale und die letzten beiden Diagnosen, jeweils eine Spalte pro Krankheitsbild.

Acute Inflammations Data Set

Merkmale x

1. Temperatur des Patienten in Grad Celsius
2. Auftreten von Übelkeit als Boolean-Wert
3. Lendenschmerzen als Boolean-Wert
4. Urinschub (kontinuierlicher Bedarf Wasser zu lassen) als Boolean-Wert
5. Blasenschmerzen als Boolean-Wert
6. Beschwerden an der Harnröhre wie Juckreiz oder Schwellung des Harnröhrenaustritts als Boolean-Wert

Zielwerte y

7. Krankheitsbild *Harnblasenentzündung* als Boolean-Wert
8. Krankheitsbild *Nierenentzündung mit Ursprung im Nierenbecken* als Boolean-Wert

Harnblasenentzündung erkennen mittels Acute Inflammations Data Set

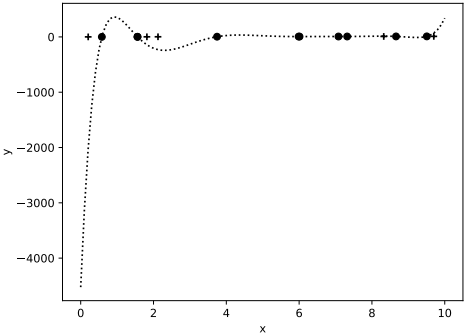
- Theoretisch sind bzgl. der Krankheitsbilder vier Kombinationen möglich:
  1. es liegt keine Diagnose vor
  2. es liegt nur die erste der beiden Diagnosen vor
  3. es liegt nur die zweite der beiden Diagnosen vor
  4. oder sogar beide
- Praktisch enthält die Datenbank 30 gesunde Personen, 19 Einträge, in denen beide Diagnosen vermerkt wurden, und 71 mit jeweils einer.
- Bzgl. der Harnblasenentzündung liegt ein recht ausgeglichenes Verhältnis von 61 False zu 59 True im Datensatz vor.
- Wenn wir nun einen Klassifikator programmieren, der zwischen der Gruppe Harnblasenentzündung *True* und *False* unterscheidet, so besteht die False-Gruppe sowohl aus Gesunden als auch aus Personen mit einer anderen Erkrankung.

Aufteilung der Daten in eine Trainings-, Validierungs- und Testmenge

- Wir verwenden für das Training/Fitting nie den ganzen Datenbestand, sondern nur einen Teil.
- Eine Menge werden wir zum Training des Klassifikators benutzen und eine, um die Qualität zu testen.
- Eine typische Aufteilung ist ca. 15% – 30% für die **Testmenge** und entsprechend 70% - 85% für die **Trainingsmenge**.
- Sollte es bei Verfahren noch die Notwendigkeit geben, Parameter im Laufe des Trainings zu wählen, wird von der Trainingsmenge oft noch eine Teilmenge zur **Validierung** abgezweigt.

Was bedeutete Overfitting?

- Der Grund für die Aufteilung liegt daran, dass Lernverfahren dazu tendieren, die Strukturen der Trainingsmenge auswendig zu lernen, falls genug Freiheitsgrade vorhanden sind.
- Nehmen wir als Beispiel an, wir hätten 15 Datensätze vorliegen, die jeweils aus einem x-Wert und einem y-Wert bestehen.
- Legen wir durch 10 dieser Werte ein Polynom und schauen, was mit den anderen 5 passiert.



- Würden wir die Qualität des Modells nur auf den Werten testen, die zum Training benutzt wurden, wären wir also ggf. begeistert.
- Sieht man sich jedoch die Testmenge ('+') an, so kann man sich vorstellen, wie fatal falsch die Aussagen des Modells für neue Daten gewesen wäre, besonders für kleine Werte.
- Die Qualität beurteilt man immer bzgl. einer Menge, die am Training nicht beteiligt war.

Konfusionsmatrix für Bayes mit nominalen Merkmalen

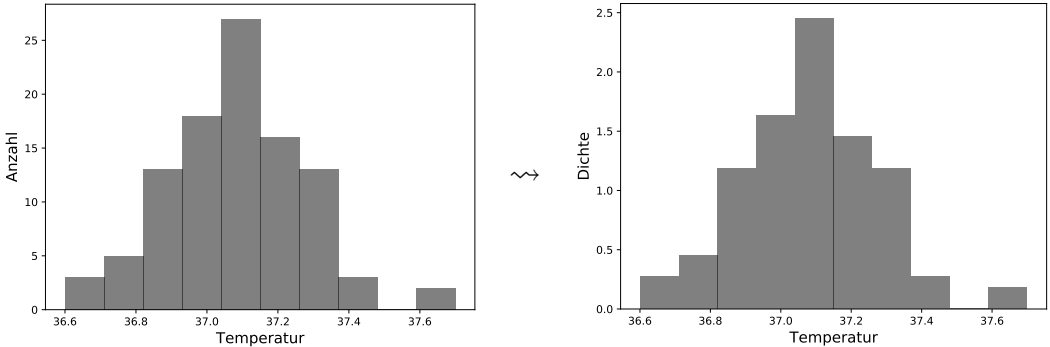
- Wir verwenden eine Testmenge von 24 Datensätzen.
- Setzen wir den Naiven Bayes-Klassifikator ausschließlich mit den nominalen Merkmalen um, also ohne Temperatur, haben wir 3 Fehler.
- Das alleine reicht nicht, um die Qualität zu beurteilen, wir müssen auch betrachten, wie die Fehler verteilt sind.
- Ein Mittel dazu ist die **Konfusionsmatrix**:

		Tatsächliche Klasse	
		nicht-erkrankt	erkrankt
Vorhergesagte Klasse	nicht-erkrankt	11	3
	erkrankt	0	10

- Im einfachsten Fall geht es um die Frage **false positives** und **false negatives**, aber wir können Konfusionsmatrizen für viel mehr Klassen später noch einsetzen.

Histogramme

- Die normale Körpertemperatur eines Menschen liegt zwischen 36.3 °C und 37.4 °C.
- Diese schwankt ein wenig im Laufe des Tages bedingt durch verschiedene Stoffwechselprozesse. Andere Abweichungen entstehen durch Sport oder Außentemperatur.
- Wenn wir mittags 100 Personen testen würden, läge der Mittelwert bei ca. 37 °C, aber mit einer Streuung z. B. so:



Wahrscheinlichkeitsdichtefunktion

- Würden wir bei immer mehr Personen messen und immer mehr Container verwenden, nähern wir uns immer mehr einer stetigen Funktion an.

Wahrscheinlichkeitsdichtefunktion

X sei eine stetige Zufallsvariable. Es existiert eine Funktion $f(x)$, für die gilt:

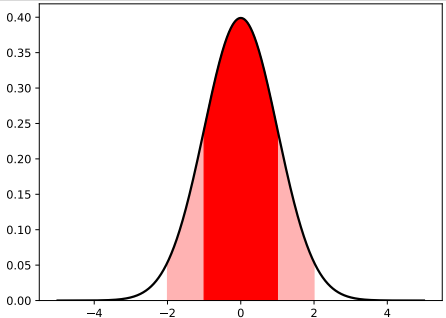
1. $P(a < X \leq b) = \int_a^b f(x)\,dx$ mit $a, b \in \mathbb{R},\ a \leq b$
2. $f(x) \geq 0$
3. $\int_{-\infty}^{+\infty} f(x)\,dx = 1$

Dann heißt die Funktion $f(x)$ Wahrscheinlichkeitsdichtefunktion der stetigen Zufallsvariable X.

- Wir schauen uns eine sehr prominente Wahrscheinlichkeitsdichtefunktion, nämlich die Gaußsche Normalverteilung, nun an.

- Die **Gaußsche Normalverteilung** ist eine Wahrscheinlichkeitsdichtefunktion mit

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$



- Bei einer Messung gibt es den **Erwartungswert** μ und wegen natürlichen Variationen oder Messfehlern kommt es zu einer Streuung um diesen Erwartungswert.
- Das Maximum der Kurve liegt entsprechend auch bei diesem Erwartungswert. Der Grad der Streuung wird durch die **Standardabweichung** σ beschrieben.
- Im Bereich von $\pm\sigma$ um μ liegen 68,27 % aller Messwerte
- Im Bereich von $\pm 2\sigma$ um μ liegen 95,45 % aller Messwerte
- Wir werden diese Gaußverteilung nun nutzen, um zu modellieren, wie wahrscheinlich in unserer Datenbank eine Körpertemperatur im Zusammenhang mit einer Diagnose ist.

- Unterstellen wir eine Gaußverteilung, nähern wir den tatsächlichen Erwartungswert durch den **Mittelwert der Stichprobe** an

$$\mu \approx \bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$$

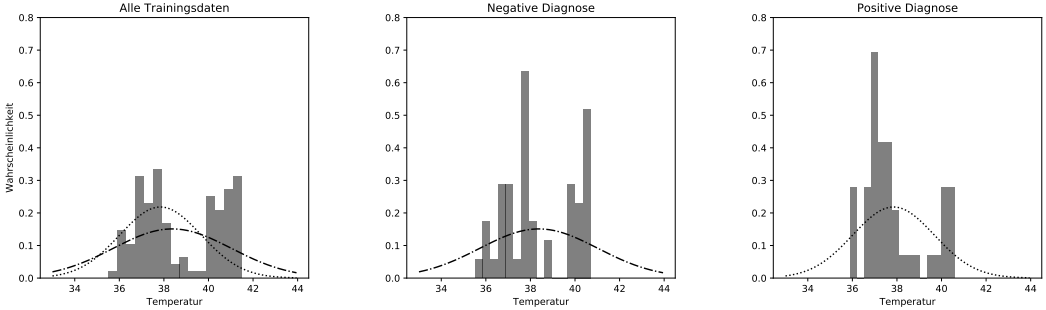
- Ebenso die **empirische Standardabweichung** mit der **Stichprobenvarianz** s^2:

$$\sigma^2 \approx s^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)^2 \Longleftrightarrow \sigma \approx s = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)^2}$$

Theorie und Praxis

In der Wahrscheinlichkeitstheorie unterscheidet man zwischen den Begriffen für die theoretischen Grenzfälle und denen für reale Stichproben. Wir haben es jedoch im praktischen Fall immer nur mit endlichen Datenmengen zu tun, also mit einer empirischen Standardabweichung. Wir werden daher im Folgenden einfach die Symbole σ statt s verwenden und knapper auch Standardabweichung schreiben, obwohl es sich nur um die empirische Standardabweichung unserer Datenbank handelt. Den Unterschied sollte man trotzdem im Kopf haben!

- Ein Punkt ist, dass die Verteilung in der Datenbank **nicht** einer **perfekten Gaußverteilung** entspricht.
- Gründe sind u.a., dass diese Gesunde & Kranke mit verschiedenen Krankheitsbildern enthält.



$$\mu_{\text{False}} = 38.45\ ,\ \sigma_{\text{False}} = 2.48 \quad \text{und} \quad \mu_{\text{True}} = 37.78\ ,\ \sigma_{\text{True}} = 1.93$$

$$P(T|\text{negative D.}) \rightsquigarrow \frac{1}{2.48\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{T-38.45}{2.48}\right)^2} \quad \text{und} \quad P(T|\text{positive D.}) \rightsquigarrow \frac{1}{1.93\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{T-37.78}{1.93}\right)^2}$$

- Wie ist das Verhältnis der Wahrscheinlichkeitsdichte f zur Wahrscheinlichkeit P, mit der wir bisher gerechnet haben?
- Das Verhältnis wird durch ein Integral bestimmt:

$$P(X \in [a, b]) = \int_a^b f(x)\,\mathrm{d}x$$

- Das bedeutet in unserem Beispiel: Wir können aus f die Wahrscheinlichkeit berechnen, dass die Temperatur zwischen 38 und 38.1 Grad liegt, indem wir das entsprechende Integral berechnen bzw. über einen Ansatz wie die Mittelpunktsregel annähern.

$$P(T \in [38, 38.1]) = \int_{38}^{38.1} f(x)\,\mathrm{d}x \approx f(38.05) \cdot (38.1 - 38) = f(38.05) \cdot \underbrace{0.1}_{\Delta x}$$

- Mit diesem Ansatz, bei dem man durch $\Delta x \to 0$ auch den Übergang zum exakten Integral bilden kann, wird plausibel, warum wir direkt mit der Dichtefunktion arbeiten können, obwohl es dann bzgl. der Einheiten keine Wahrscheinlichkeit mehr ist.

- Merkmale von $0 \cdots d$ sind diskrekt, von $d+1 \ldots m$ sind kontinuierlich
- Breite der Integrationsintervalle ist Δx

$$
\begin{aligned}
P(i \mid x) &= \frac{P(i) \cdot \prod_{k=1}^{d} P(x^{(k)} \mid i) \cdot \prod_{k=d+1}^{m} P(x^{(k)} \mid i)}{\sum_{j=1}^{N} P(j) \cdot \prod_{k=1}^{d} P(x^{(k)} \mid j) \cdot \prod_{k=d+1}^{m} P(x^{(k)} \mid j)} \\
&\approx \frac{P(i) \cdot \prod_{k=1}^{d} P(x^{(k)} \mid i) \cdot \prod_{k=d+1}^{m} f(x^{(k)} \mid i) \cdot \Delta x}{\sum_{j=1}^{N} P(j) \cdot \prod_{k=1}^{d} P(x^{(k)} \mid j) \cdot \prod_{k=d+1}^{m} f(x^{(k)} \mid j) \cdot \Delta x} \\
&= \frac{\cancel{\Delta x} \cdot P(i) \cdot \prod_{k=1}^{d} P(x^{(k)} \mid i) \cdot \prod_{k=d+1}^{m} f(x^{(k)} \mid i)}{\cancel{\Delta x} \cdot \sum_{j=1}^{N} P(j) \cdot \prod_{k=1}^{d} P(x^{(k)} \mid j) \cdot \prod_{k=d+1}^{m} f(x^{(k)} \mid j)}
\end{aligned}
$$

Verwendung im Naiver Bayes-Klassifikator

- Diese Wahrscheinlichkeiten fügen wir jetzt als weiteren Faktor in die Formel für den Naiven Bayes-Klassifikator ein.
- Da wir es hier nur mit einer binären Klassifikation (zwei Klassen) zu tun haben, ergeben sich Terme der folgenden Art:

$$P(\text{False} \mid x) = \frac{P(\text{False}) \cdot \prod_{k=1}^{m} P(x^{(k)} \mid \text{False}) \cdot f(T|\text{False})}{\left(P(\text{False}) \cdot \prod_{k=1}^{m} P(x^{(k)} \mid \text{False}) \cdot f(T|\text{False}) + P(\text{True}) \cdot \prod_{k=1}^{m} P(x^{(k)} \mid \text{True} \cdot f(T|\text{True})) \right)}$$

- Nutzen wir dieses zusätzliche Merkmal, so erhalten wir trotz aller Schwächen ein verbessertes Ergebnis und klassifizieren alle Elemente der Testmenge fehlerfrei.
- Das hängt bei einer so kleinen Datenbank auch ein wenig von der Wahl der Testmenge ab.
- Wie kann man die Aussage bzgl. der Genauigkeit unseres Modells bzw. der Methode und ihrer Parameter weiter verbessern?

Kreuzvalidierung

- Was wir bisher gemacht haben, nennt man auch **Holdout**-Ansatz als Spezialfall einer Kreuzvalidierung.
- Bei kleinen Datenmengen kann dieser Ansatz aber trügerisch sein. In einem solchen Fall kann eine **k-fache Kreuzvalidierung** sinnvoll sein.
- Hierbei werden die Daten in k zufällig ausgewählte Teilmengen ähnlicher Größe aufgeteilt.
- Dann muss die Abfolge aus Training und Test k-fach durchgeführt werden. Hierbei wird jeweils eine Teilmenge zum Testen verwendet, und mit den restlichen $k-1$ Teilmengen wird trainiert.
- Nehmen wir an, dass der Fehler in Durchlauf i mit E_i bezeichnet wird, dann bildet man einfach mit

$$E = \frac{1}{k} \sum_{i=1}^{k} E_i$$

das arithmetische Mittel. Ein typischer Wert für k liegt in der Region um 10.