Bayesian Filter, um Spam-Mails zu erkennen

Das *Theorem von Bayes* stellt einen Zusammenhang her zwischen der bedingten Wahrscheinlichkeit $P(A|B)$ und der umgedrehten bedingten Wahrscheinlichkeit $P(B|A)$. Dies ist nützlich, da man häufig bedingte Wahrscheinlichkeit in die eine Richtung gut schätzen/messen kann, sich aber eigentlich für die andere Richtung interessiert (wie wir schon in einigen Übungsaufgaben gesehen haben).

Für zwei Ereignisse A und B mit $P(A) \neq 0$ und $P(B) \neq 0$ folgt aus der Definition der bedingten Wahrscheinlichkeit

$$P(A|B) = \frac{P(A \cap B)}{P(B)} \qquad \text{und} \qquad P(B|A) = \frac{P(B \cap A)}{P(A)}$$

und der Formel der totalen Wahrscheinlichkeit

$$P(A) = P(B) \cdot P(A|B) + P(B^c) \cdot P(A|B^c)$$

nämlich:

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

Diese Formel geht auf den Mathematiker Thomas Bayes (1702-1761) zurück und wird z.B. in der Medizin sehr häufig verwendet. Weiss man beispielsweise, wie häufig eine Krankheit generell auftritt, wie viele Menschen in der Bevölkerung rauchen und wie viele der Erkrankten geraucht haben, kann man daraus einfach errechnen, wie hoch die Wahrscheinlichkeit ist, dass jemand mit der Erkrankung auch geraucht hat.
Das Theorem von Bayes kann auch benutzt werden, um Filter zu bauen, die Spam-E-Mails erkennen. Bei der Erfindung von E-Mails dachte niemand an irgendwelche Leute, die gefälschte Luxusuhren, Medikamente gegen Essstörungen oder Aktien-Scams an die Menschheit bringen wollen... Spam zu verschicken ist günstig und einfach, und so geht man heute davon aus, dass ca. 90 Prozent aller verschickten E-Mails in diese Kategorie fallen.

Die Idee hinter einem sogenannten Bayesschen Spamfilter ist erstaunlich einfach: Der Filter benützt Häufigkeitsinformationen aus vergangenen E-Mails, um abzuschätzen, ob eine neu eintreffende E-Mail als Spam oder als Non-Spam klassifiziert werden soll. Dabei berücksichtig der Filter insbesondere, ob gewisse Wörter in einer E-Mail auftreten. Intuitiv ist nämlich klar, dass bestimmte Wörter (wie z.B. „Viagra“ oder „Rolex“) deutlich häufiger in einer Spam-Mail als in einer Non-Spam-Mail auftreten werden.

Um unsere Überlegungen in mathematischer Sprache zu formulieren, definieren wir die folgenden zufälligen Ereignisse:

- S sei das Ereignis, dass eine eintreffende E-Mail eine Spam-Mail ist;
- E sei das Ereignis, dass eine eintreffende E-Mail ein bestimmtes Wort w enthält.

Mit dem Theorem von Bayes erhalten wir nun folgenden Ausdruck für die Wahrscheinlichkeit, dass eine E-Mail, die das Wort w enthält, eine Spam-Mail ist:

$$P(S|E) = \frac{P(E|S) \cdot P(S)}{P(E|S) \cdot P(S) + P(E|S^c) \cdot P(S^c)}$$

Das Gegenereignis S^c bezeichnet hier natürlich das Ereignis, dass eine eintreffende E-Mail eine normale E-Mail, d.h. Non-Spam oder ein sogenanntes *Ham* ist.
Für die Wahrscheinlichkeiten $P(S)$ und $P(S^c)$ setzen wir für den Moment die oben erwähnten Erfahrungswerte 0.9 bzw. $1 - 0.9 = 0.1$ ein, die diese Wahrscheinlichkeiten zumindest annähern. (Wir werden später nochmals zurückkommen auf diese Werte.) Natürlich können wir auch die Wahrscheinlichkeiten $P(E|S)$ und $P(E|S^c)$ nicht exakt bestimmen. (Dafür müssten wir die vollständige Information über alle je verschickten E-Mails der Vergangenheit und Zukunft haben.) Wir können aber diese Wahrscheinlichkeiten *abschätzen*, indem wir eine grosse Anzahl an E-Mails untersuchen. Je grösser diese Anzahl, desto genauer werden unsere Abschätzungen dieser Wahrscheinlichkeiten, und desto besser wird unser Filter funktionieren.

Konkret geht man wie folgt vor:

1. Zuerst wird eine Vielzahl an E-Mails gesammelt und (von Hand) in die Kategorien „Spam“ und „Non-Spam“ sortiert.
2. Die E-Mails beider Kategorien werden auf das Auftreten bestimmter Wörter untersucht. Die entsprechenden Auftretenshäufigkeiten liefern eine *Schätzung* für die tatsächlichen Wahrscheinlichkeiten, dass ein bestimmtes Wort in einer Spam- oder Non-Spam-Mail auftritt, also für die bedingen Wahrscheinlichkeiten $P(E|S)$ und $P(E|S^c)$.

   Die so erhaltenen Schätzungen für die unbekannten tatsächlichen Wahrscheinlichkeiten $P(x|y)$ bezeichnen wir als $\hat{p}(x|y)$, um diese zwei Dinge klar zu unterscheiden.
3. Mit Hilfe dieser Schätzungen und dem Satz von Bayes kann nun (näherungsweise) berechnet werden, mit welcher Wahrscheinlichkeit eine neu eintreffende E-Mail, die ein bestimmtes Wort enthält, eine Spam-Mail ist. Durch Setzen eines Schwellenwertes für diese Wahrscheinlichkeit kann der Spam-Filter kalibriert werden.

Ein Beispiel
Wir haben 2000 Spam-Mails und 1000 Non-Spam-Mails betrachtet und herausgefunden, dass das Wort „Rolex“ in 250 der Spam-Mails und in 5 der Non-Spam-Mails erscheint. Sollen wir eine E-Mail, dass das Wort „Rolex“ enthält, nun als Spam-Mail verwerfen oder nicht?
Aus den oben angegebenen Häufigkeiten ergeben sich die Schätzungen $\hat{p}(E|S) = 250/2000 = 0.125$ und $\hat{p}(E|S^c) = 5/1000 = 0.005$. Mit der bereits erwähnten Erfahrungswerten $P(S) = 0.9$ und $P(S^c) = 0.1$ ergibt sich damit

$$\hat{p}(S|E) = \frac{0.125 \cdot 0.9}{0.125 \cdot 0.9 + 0.005 \cdot 0.1} \approx 0.996$$

Nach unser Berechnung handelt es sich bei dieser E-Mail also mit fast 100% Wahrscheinlichkeit um eine Spam-Mail und wir werden diese E-Mail als Spam einstufen. Ob wir eine E-Mail im Allgemeinen als Spam verwerfen oder nicht, hängt davon ab, wo wir den Schwellenwert unseres Spamfilter setzen. Falls $\hat{p}(S|E)$ grösser ist als der gewählte Schwellenwert, kategorisieren wir die E-Mail als Spam. Je höher der gewählte Schwellenwert, desto mehr E-Mails wird der Filter durchlassen.
Die folgende Tabelle heisst Wahrheitsmatrix (*engl. confusion matrix*) und zeigt die vier Fälle auf, die beim Klassifizieren von E-Mails auftreten können (binäre Klassifikation, da nur zwei Ausgänge möglich sind).

		Tatsächliche Kategorie **(Observed)**		
		Spam	*Non-Spam*	Summe
Ergebnis des Spamfilters **(Predicted)**	*Spam (Positive)*	**TP** richtig identifiziert (true positive)	**FP** falsch identifiziert (false positive)	**TP + FP**
	Non-Spam (Negativ)	**FN** falsch identifiziert (false negative)	**TN** richtig identifiziert (true negative)	**FN + TN**
	Summe	**TP + FN**	**FP + TN**	Total

Tabelle 1: Die vier möglichen Ausgänge unser binären Klassifikation als Wahrheitsmatrix (*engl. confusion matrix*) dargestellt.

In der Literatur werden häufig die englischen Bezeichnungen verwendet:

1. true positive (TP) – correct positive prediction
2. false positive (FP) – incorrect positive prediction
3. true negative (TN) – correct negative prediction
4. false negative (FN) – incorrect negative prediction

Grundlegende Masszahlen

Aus der Wahrheitsmatrix können quantitative Masszahlen abgeleitet werden, die uns helfen, die Güte der Ergebnise unser Klassifikation zu beurteilen. Zur Vereinfachung der Diskussion betrachten wir hier ein konkretes Zahlenbeispiel und berechnen anhand dessen die entsprechenden Schätzungen für die Masszahlen.

		Tatsächliche Kategorie **(Observed)**		
		Spam	*Non-Spam*	Summe
Ergebnis des Spamfilters **(Predicted)**	*Spam (Positive)*	**TP** 400	**FP** 20	420
	Non-Spam (Negativ)	**FN** 10	**TN** 1000	1010
	Summe	410	1020	1430

Tabelle 2: Ein Zahlenbeispiel

Bezeichnung *english Term*	Masszahl als Wahrscheinlichkeit und die Formel für die empirische Berechnung	Schätzwerte Zahlenbeispiel
Fehlerrate *error rate*	$Err = P(\text{„Vorhersage falsch“}) = \frac{FP+FN}{TP+FP+TN+FN}$	$\frac{20+10}{1430} \approx 0.02$
Genauigkeit *accuracy*	$Acc = P(\text{„Vorhersage korrekt“}) = \frac{TP+FP}{TP+FP+TN+FN}$	$\frac{1000+400}{1430} \approx 0.98$
Senstivität *senstivity*	$Sens = P(\text{„Vorhersage Spam“} \mid \text{„tatsächlich Spam“}) = \frac{TP}{TP+FN}$	$\frac{400}{410} \approx 0.98$
Spezifität *specificity*	$Spez = P(\text{„Vorhersage Ham“} \mid \text{„tatsächlich Ham“}) = \frac{TN}{TN+FP}$	$\frac{1000}{1020} \approx 0.98$
Positiver Vorhersagewert *positive predictive value*	$PV^+ = P(\text{„tatsächlich Spam“} \mid \text{„Vorhersage Spam“}) = \frac{TP}{TP+FP}$	$\frac{400}{420} \approx 0.95$
Negativer Vorhersagewert *negative predictive value*	$PV^- = P(\text{„tatsächlich Ham“} \mid \text{„Vorhersage Ham“}) = \frac{TN}{TN+FN}$	$\frac{1000}{1010} \approx 0.99$

1. Die Fehlerrate (Err) und die Genauigkeit (Acc) sind die am häufigsten verwendeten Masszahlen, die intuitiv ebenfalls am einfachsten zu verstehen sind. Die Fehlerrate (Err) wird als die Anzahl aller fehlerhaften Vorhersagen geteilt

durch die Gesamtzahl Fälle berechnet. Der beste Wert für die Fehlerrate beträgt 0, während der schlechteste Wert 1 beträgt. Die Genauigkeit (*Acc*) wird als die Anzahl aller korrekten Vorhersagen geteilt durch die Gesamtzahl Fälle berechnet. Die beste Genauigkeit ist 1, während die schlechteste 0.0 ist. Die Genauigkeit ist gleich $1 - Err$.

2. Beim Aussortieren von E-Mails will man unbedingt vermeiden, dass Non-Spam-Mails als Spam-Mails identifiziert und aussortiert werden. Ein Spam-Mail als Non-Spam zu identifizieren ist deutlich weniger schlimm als der umgekehrte Fall. In vielen Situationen sind Fehlerkosten von positiven und negativen Vorhersagen unterschiedlich. Daher sind andere grundlegende Masszahlen wie Senstivität und Spezifität in solchen Fällen informativer als die Genauigkeit oder die Fehlerrate.

   Sensitivität (Rate der richtig-positiven Klassifikationen, auch Trefferquote genannt; *englisch: sensitivity, true positive rate, hit rate*)

   Die Sensitivität (*Sens*) gibt den Anteil der korrekt als positiv klassifizierten Objekte an der Gesamtheit der tatsächlich positiven Objekte an. Die beste Sensitivität beträgt 1, während die schlechteste 0 ist.

   Spezifität (Rate der richtig-negativen Klassifikationen, *englisch: specificity, true negative rate, correct rejection rate)*

   Die Spezifität (*Spez*) gibt den Anteil der korrekt als negativ klassifizierten Objekte an der Gesamtheit der in Wirklichkeit negativen Objekte an. Die beste Spezifität beträgt 1, während die schlechteste 0 beträgt.

3. Sensitivität und Spezifität charakterisieren die Güte der Ergebnis einer Klassifikation. Sie geben die Wahrscheinlichkeit für ein positives oder negatives Testergebnis an, wenn das E-mail ein Spam-Mail respektive Ham-Mail ist. Wir interessieren uns aber unter Umständen für die Güte der umgekehrten Wahrscheinlichkeit. D.h. was ist die Wahrscheinlichkeit, dass ein E-Mail tatsächlich Spam oder Ham ist, wenn das Ergebnis der Klassifizierung positiv respektive negativ ist.

   Positiver Vorhersagewert (auch Relevanz, Wirksamkeit, positiver prädiktiver Wert genannt; *englisch: precision, positive predictive value*)

   Der positive Vorhersagewert (PV^+) gibt den Anteil der korrekt als positiv klassifizierten Ergebnisse an der Gesamtheit der als positiv klassifizierten Ergebnisse an (erste Zeile der Wahrheitsmatrix). Der beste Wert ist 1, während der schlechteste 0 ist.

   Negativer Vorhersagewert (auch Segreganz, Trennfähigkeit genannt; *englisch: negative predictive value)*

   Entsprechend gibt der negative Vorhersagewert (PV^-) den Anteil der korrekt als negativ klassifizierten Ergebnisse an der Gesamtheit der als negativ klassifizierten Ergebnisse an (zweite Zeile der Wahrheitsmatrix). Der beste Wert ist 1, während der schlechteste 0 ist.

Sie werden nun mit Hilfe eines vorbereiteten Datensatzes einen Bayesschen Spamfilter programmieren und testen. Dieser Datensatz wurde an den Hewlett-Packard Labs vorbereitet und enthält Daten zu 2900 E-Mails, die von Usern als Spam oder Non-Spam klassifiziert wurden.
Sie finden die Daten auf Moodle in der Datei `Spam.xlsx`. Darin enthalten ist eine 2900×58-Tabelle, deren Zeilen den einzelnen E-Mails entsprechen und deren Spalten wie folgt zu interpretieren sind:

- Die ersten 48 Spalten enthalten die Auftretenshäufigkeiten des Variablennamens (z.B. make) in der E-Mail (in Prozent). Der entsprechende Eintrag ist

$$100 \cdot \frac{\text{Anzahl Vorkommen des Worts}}{\text{Anzahl Wörter der gesamten E-Mail}}.$$

  Beginnt der Variablenname mit num (z.B. num650), so gibt er die Häufigkeit der entsprechenden Zahl (in diesem Bsp. also 650) an.

- Die Variablen 49-54 enthalten analog die Auftretenshäufigkeiten der folgenden Zeichen in der E-Mail: Semikolon „;“, runde Klammer „(“, eckige Klammer „[“, Ausrufezeichen „!“, Dollarzeichen „$“ und Hash „#“. Der entsprechende Eintrag ist

$$100 \cdot \frac{\text{Anzahl Vorkommen des Zeichens}}{\text{Anzahl Zeichen der gesamten E-Mail}}.$$

- Die Variablen 55-57 enthalten die durchschnittliche, längste und gesamte Lauflänge von Grossbuchstaben in der E-Mail.

- Die letzte Spalte, nämlich „type“ enthält die Klassifikation in Spam und Non-Spam.

Ebenfalls auf Moodle finden Sie ein Jupyter Notebook `Spamfilter_starthilfe.ipyb`, das diesen Datensatz für Sie einliest und dann *zufällig* in zwei Mengen aufteilt: einen Trainingsdatensatz (80% der Daten) und einen Testdatensatz (20% der Daten). Wie die Namen schon andeuten, soll der Trainingsdatensatz verwendet werden, um unseren Spamfilter zu entwickeln/trainieren, und der Testdatensatz, um den Spamfilter danach zu testen. Diese beiden Datensätze dürfen nicht vermischt werden!
Der Testdatensatz wird also erst einmal nicht benötigt. Für die erste Spalte des Trainingsdatensatzes werden beispielhaft Histogramme für die tabellierten Häufigkeiten erstellt, jeweils separat für die Spam- und die Non-Spam-Mails. Solche Histogramme vermitteln einen ersten Eindruck von den charakteristischen Eigenschaften von Spam-Mails.

Nun sind Sie dran. Zum Aufwärmen versuchen Sie die folgenden Übungen zu lösen. Die Lösungen `Spamfilter_sol.ipyb` stehen Ihnen ebenfalls auf Moodle zur Verfügung.

1. Bestimmen Sie für das Ereignis E =„enthält mindestens 3% Ausrufezeichen“ analog zum Beispiel oben die Schätzwerte $\hat{p}(E|S)$, $\hat{p}(E|S^c)$ und $\hat{p}(S|E)$. Sollten wir E-Mails mit der Eigenschaft E also als Spam klassifizieren, wenn wir für das Aussortieren einen Schwellenwert von 98% verwenden?

Bemerkung: Das Ereignis E hat nicht exakt die oben diskutierte Form „enthält ein bestimmtes Wort (mindestens einmal)“; das ändert aber nichts am allgemeinen Vorgehen.

2. Erstellen Sie nun für den Trainingsdatensatz die Konfusionsmatrix bezüglich des Kriteriums E. Berechnen Sie die Masszahlen. Wie beurteilen Sie das Ergebnis?

3. Variieren sie nun die prozentuale Grenze (also die 3%) im Ereignis E, und versuchen Sie, Ihre Ergebnisse aus a) und b) so zu verbessern. (Arbeiten Sie weiterhin nur auf dem Trainingsdatensatz.)

4. Wenn Sie zufrieden mit dem Ergebnis sind, testen Sie Ihr so gefundenes Kriterium nun auf dem Testdatensatz. Entspricht das Resultat Ihren Erwartungen?

5. Ersetzen Sie nun die Schätzungen 0.9 und 0.1 für $P(S)$ und $P(S^c)$ jeweils durch 0.5 und spielen Sie die bisherigen Schritte nochmals durch. Was stellen Sie fest? Erhalten Sie immer noch sinnvolle Resultate?

Projektarbeit Spamfilter

Versuchen Sie nun selbstständig, den Spamfilter weiter zu verbessern, indem Sie andere Kriterien E definieren und testen. Diese können und sollen nun mehrere Spalten des Datensatzes miteinbeziehen, also z.B. die Form haben „enthält mindestens 3% Ausrufezeichen und mindestens 1% Dollarzeichen“

Grundsätzlich haben Sie zwei Optionen zur Verfügung:

1. Sie nehmen an, dass verschiedene Ereignisse (z.B. „enthält mindestens 3% Ausrufezeichen und mindestens 1% Dollarzeichen“) jeweils abhängig voneinander sind und berechnen mit den richtigen Formeln weiter.

2. Sie nehmen an, dass verschiedene Ereignisse (z.B. „enthält mindestens 3% Ausrufezeichen und mindestens 1% Dollarzeichen“) unabhängig voneinander sind und programmieren einen sogenannten *naive Bayes Classifier*. Wie der Name bereits sagt, ist es *naiv* zu glauben, dass die Ereignisse unabhängig voneinander sind. Allerdings sind Filter, die auf dieser naiven Annahme beruhen ertaunlich erfolgreich und werden häufig in Software eingesetzt.

   Die folgende Seite gibt Ihnen eine Einführung `https://towardsdatascience.com/how-to-build-and-apply-naive-bayes-classification-for-spam-filtering-2b8d3308501`

Ziel Ihrer Projektarbeit ist es, den bestmöglichen Spamfilter zu programmieren. Die Person mit dem besten bzw. zweit besten Ergebnis erhält 5 respektive 4 Bonuspunkte (bei Gleichstand gibt es 4 bzw. 3 Punkte). Falls Sie einen funktionsfähigen Spam-Filter liefern (und nicht einfach eine Kopie meines Codes), erhalten Sie mindestens 3 Bonuspunkte. Ihr Code inkl. Dokumentation geben Sie auf Moodle ab.