Projektpraktikum II (Stochastik)
Wintersemester term 2018/19
Humboldt-Universität zu Berlin

Sebastian Holtz

Projektaufgaben Block 2

1. Lade den Datensatz *Advertising.csv* von der Kursseite. Ziel der Aufgabe ist es, den Einfluss von Werbung im Fernsehen, im Radio und in Zeitungen auf die Verkaufszahlen mittels linearen Modellen zu analysieren. Mache dich hierfür mit den Begriffen p-Wert, Bestimmtheitsmaß R^2 sowie $t-$ und $F-$Statistik vertraut und nutze diese zur Bearbeitung folgender Aufgaben.

   (a) Analysiere für jeweils jeden Regressor ein einfaches lineares Modell. Erzeuge jeweils einen Plot, welcher die Beobachtungen und die Regressionsgerade basierend auf dem kleinsten Quadrate-Schätzer darstellt.

   (b) Analysiere nun eine multiple lineare Regression basierend auf allen drei Regressoren. Entscheide, ob auf Regressoren verzichtet werden kann und falls ja, analysiere das resultierende reduzierte lineare Modell und zeichne die Beobachtungen sowie $X\hat{\beta}$ in einen gemeinsamen Plot.

   (c) Diskutiere die Diskrepanz zwischen den in (b) als vernachlässigbar ermittelten Regressoren und deren Rolle im einfachen linearen Modell in (a). Nutze hierfür eine Korrelationsmatrix der Regressoren.

   (d) Nutze nun für das (nach (b)) reduzierte lineare Modell nur 75% der Stichprobe zur Parameterschätzung und analysiere den Vorhersagefehler mittels der verbliebenen 25%. Visualisiere das Resultat mittels eines aussagekräftigen Plots.

2. (a) Extrahiere die Datei *SpamOrHam.zip* von der Webseite. Lies die Trainingsdateien (*trainfeatures.txt, train-labels.txt*) als Matrizen ein. Verwende dafür den Befehl `read.table`. Die Spalten der Features sind Dokument-ID, Wort-ID und Worthäufigkeit des Wortes im angegebenen Dokument. In der Datei *dictionary.txt* findet man die Zuordnung zwischen einem Wort und seiner Wort-ID. Insgesamt verwenden wir nur die 2500 häufigsten Wörter in allen Emails (Trainings- und Testdaten). Die Labels 1 und 0 entsprechen den Klassen SPAM und HAM (= kein SPAM).

   (b) Erzeuge eine Featurematrix, wobei jede Zeile einem Dokument entspricht und die Worthäufigkeiten aller möglichen Worte im Dokument enthält. Beachte, dass die meisten Einträge in einer Zeile gleich 0 sein werden, da jede Email nur einen kleinen Teil aller Wörter im Wörterbuch enthält. Benutze dafür die Funktion `sparseMatrix` aus dem Package Matrix, um eine dünnbesetzte Matrix zu erzeugen (und um damit Speicherplatz zu sparen).

(c) Erzeuge die Wahrscheinlichkeiten für den Naive-Bayes-Classifier aus der Vorlesung. Beachte dabei Folgendes. Ein Dokument entspricht einem Vektor $x \in \{1, \dots 2500\}^p$, wobei $1 \leq p \leq p_{\max}$ und $p_{\max}$ ist die maximale Länge aller Emails, die wir betrachten. Die Variablen (X_j, Y_j) aus der Vorlesung enthalten dann für das j-te Dokument die Klasse $Y_j \in \{0, 1\}$ und den Vektor $X_j \in \{1, \dots, 2500\}$, wobei $1 \leq p_j \leq p_{\max}$ die Länge des j-ten Dokuments ist, und es gilt $X_j^{(k)} = i$, wenn das k-te Wort im Dokument dem i-ten Wort im Wörterbuch entspricht. Insbesondere ist die Verteilung P^X von X auf der Menge aller möglichen Dokumente definiert. Unsere Features enthalten allerdings keine Informationen über die Wortpositionen innerhalb einer Email. Denn wir machen die naive Annahme, dass alle Wortpositionen in einer Email, gegeben die Klasse Y , unabhängig voneinander sind. Daher ignorieren wir die Position und schätzen für $x \in \{1, \dots, 2500\}^p$

$$P(X = x|Y = 1) = \Phi_{x|1} = \prod_{k=1}^{p} \Phi_{x_k|1}$$

und für alle $i \in \{1, \dots, 2500\}$ unabhängig von k (!)

$$\Phi_{i|1} \approx \frac{\sum_{j=1}^{n} \sum_{m=1}^{p_j} \mathbb{1}(X_j^{(m)} = i, Y_j = 1) + 1}{\sum_{j=1}^{n} p_j \mathbb{1}(Y_j = 1) + |V|}.$$

Hierbei entspricht i dem Wort x_k an der k-ten Stelle. Allerdings ist $\Phi_{i|1}$ für alle k gleich. $|V|$ ist die Größe des Wörterbuchs. $\Phi_{x|0}$ und die $\Phi_{i|0}$ erhält man ähnlich. Beachte, dass Zähler und Nenner leicht angepasst wurden im Vergleich zur Vorlesung. Gib eine passende Erklärung für diese Anpassung an.

(d) Lies die Testdateien (*testFeatures.txt, test-label.txt*) als Matrizen ein (wie oben). Ermittle mit den geschätzten Parametern aus (c) für jedes Dokument eine der Klassen 0 oder 1 (beachte die Verwendung von Logarithmen, wie in der Vorlesung angegeben). Vergleiche mit den angegebenen Labels in *test-labels.txt*. Diskutiere das Ergebnis. Welche Wörter sind besonders gute Indikatoren für SPAM oder HAM?
Hinweis: Es sollten 6 falsche Klassifikationen erfolgen.

3. Aufgaben zur Hauptkomponentenanalyse

(a) Weise nach, dass das Minimierungsproblem (mit Notation der Vorlesung)

$$\min_{A,(v_i)_{1\leq i\leq n}} \sum_{i=1}^{n} \|X_i - Av_i\|^2, \tag{0.1}$$

die Lösungen $\hat{v}_i = \hat{A}^\top (X_i - \bar{X})$, $i = 1, \dots, n$, und $\hat{A} = (w_1, \dots, w_q)$ besitzt, sofern $\bar{X} = 0$. Zeige ferner, dass

$$\min_{\mu,A,(v_i)_{1\leq i\leq n}} \sum_{i=1}^{n} \|X_i - \mu - Av_i\|^2,$$

von $\hat{\mu} = \bar{X}$ und $\hat{A}$, $(\hat{v}_i)_{1 \leq i \leq n}$ (wie oben) gelöst werden.
Hinweis: Begründe und nutze im zweiten Minimierungsproblem, dass o.B.d.A. $\bar{v} = 0$ angenommen werden kann.

(b) Entpacke den Datensatz *Faces.zip* von der Homepage. Ziel ist es die Hauptkomponentenanalyse anhand dieses Datensatzes zu veranschaulichen. Diskutiere dafür zuerst deine Eindrücke (rein optisch) von Projektionen einzelner Bilder mittels q Hauptkomponenten, für verschiedene q. Experimentiere danach mittels einer Trainingsmenge des Datensatzes, inwiefern Bilder eines Testdatensatzes richtig klassifiziert werden.

Lösungen in einer zip-Datei per Mail bis zum 13. Dezember, 23:59 Uhr, abgeben.