Hauptkomponentenanalyse

Ron Wehrens, Christoph Steinbeck, et al.

21/06/2018

Vorbereitung

Diese Übung wird mit den Daten aus Forina et alii 1986 : M Forina, C Armanino, M Castino, M Ubigli, « Multivariate Data Analysis as a Discriminating Method of the Origin of Wines », Vitis. (http://www.vitis-vea.de/admin/volltext/e025968.pdf) arbeiten. Laden Sie sich den Artikel herunter und studieren Sie ihn bis einschliesslich der Sektion “Correlation analysis - Typical Correlations”. Beachten Sie das Argument für die Variablen-Selektion.

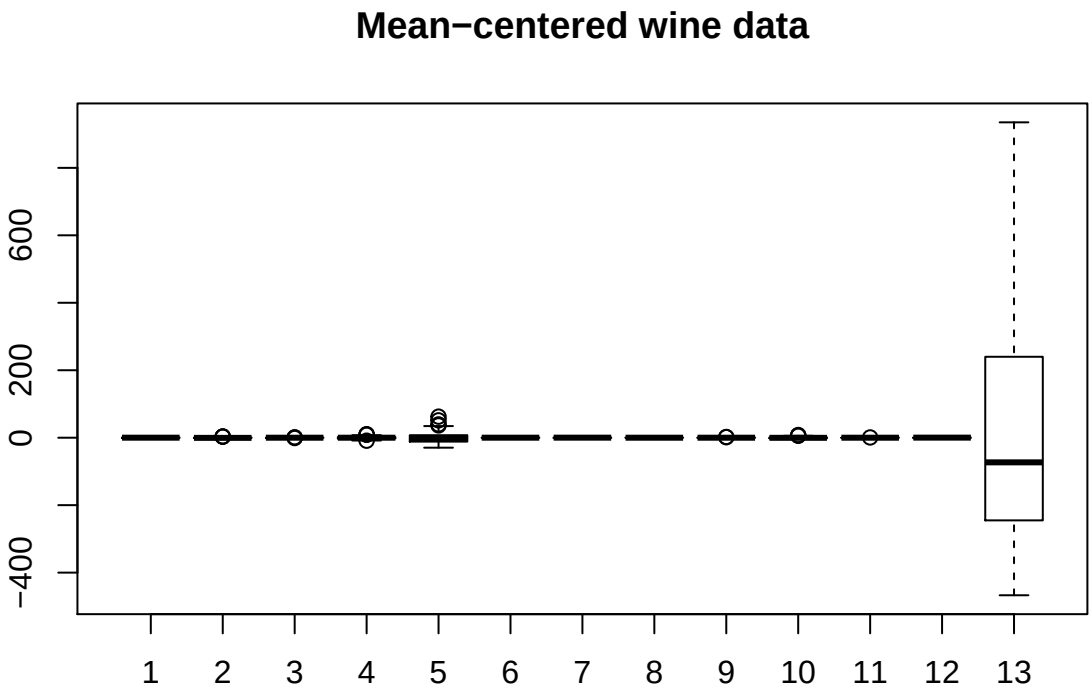
Datensichtung

Laden Sie mit dem Befehl den Weindatensatz und inspizieren Sie ihn. Welche Einheiten besitzen die einzelnen Variablen? Welche Wertebereiche werden überstrichen.

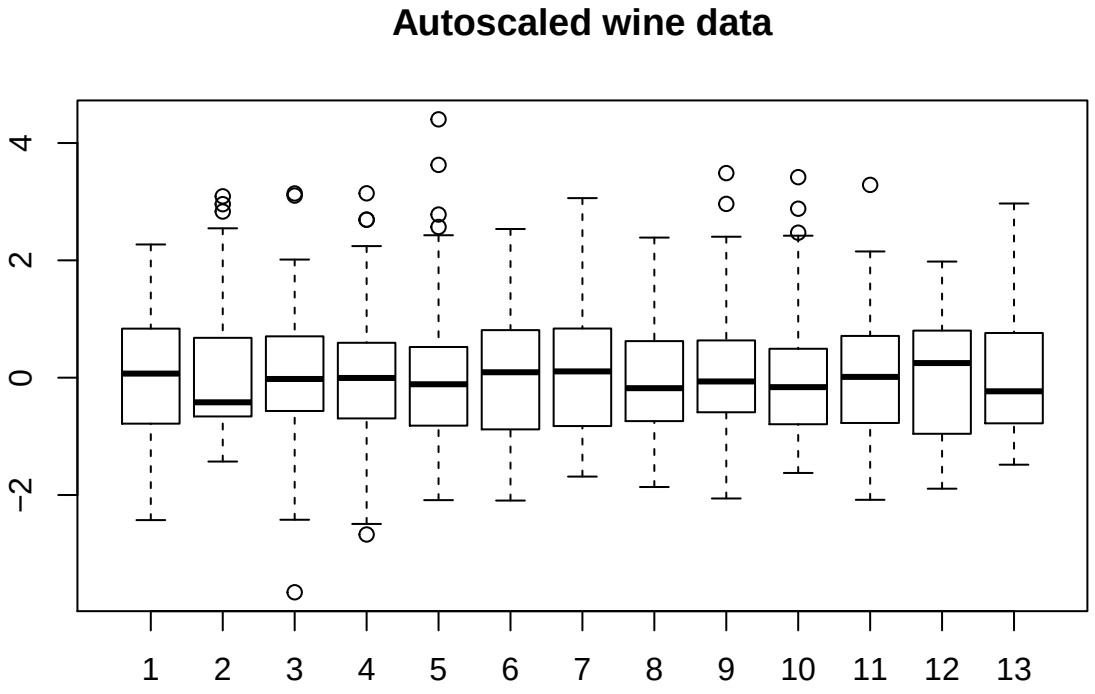
```
data(wines, package = "kohonen")
```

Erstellen Sie nach der Inspektion der Datenmatrix Boxplots, um einen Eindruck von der Antwort auf die o. g. Fragen zu bekommen. Beachten Sie den ‘Mißbrauch’ der Scale-Funktion zum Zentrieren der Daten (Siehe Hilfe zur Scale-Funktion)

```
wines.mc <- scale(wines, scale=FALSE)
wines.sc <- scale(wines)
boxplot(wines.mc ~ col(wines.mc),main = "Mean-centered wine data")
```

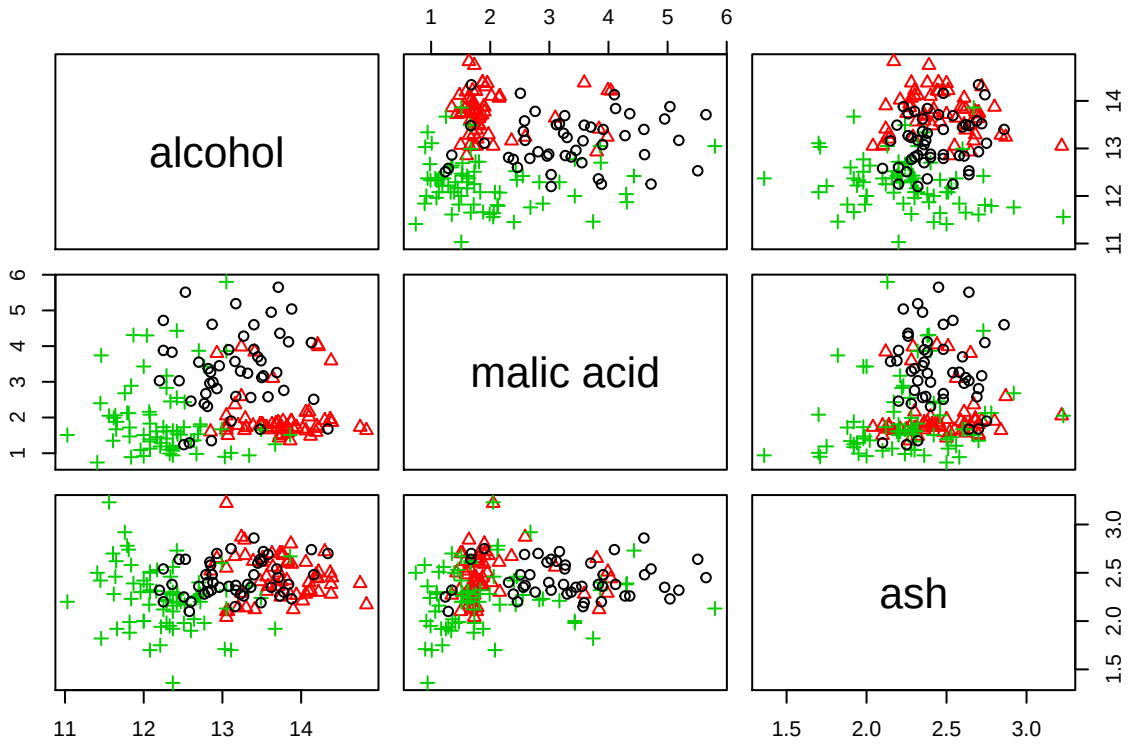



```
boxplot(wines.sc ~ col(wines.sc), main = "Autoscaled wine data")
```



Um einen Eindruck von der Struktur der Daten zu bekommen, verwenden Sie den pairs()-Befehl, um einige paarweise Auftragungen der Variablen gegeneinander zu erzeugen. Um inspizierbare Graphiken zu erhalten, empfiehlt es sich, eine kleine Zahl von Variablen-Paaren zu wählen, z. B. die ersten drei.

```
data(wines, package = "kohonen")
wine.classes <- as.numeric(factor(vintages, labels =c(1,2,3)))
pairs(wines[,1:3], pch = wine.classes, col = wine.classes)
```



Als nächstes wüßten wir gerne, ob es im vorliegenden Datensatz Variablen gibt, die stark miteinander

korrelieren. Hierzu bilden wir die Korrelationmatrix. Lesen Sie die Hilfe zum cor()-Kommando und wenden Sie das Verfahren auf die Weindaten an. Runden Sie das Ergebnis auf zwei Nachkommastellen zur besseren Lesbarkeit.

```
wines.cor <- cor(wines)
round(wines.cor, digits=2)
```

```
##                   alcohol malic acid   ash ash alkalinity magnesium
## alcohol              1.00       0.10  0.21          -0.30      0.26
## malic acid           0.10       1.00  0.16           0.29     -0.05
## ash                  0.21       0.16  1.00           0.45      0.29
## ash alkalinity      -0.30       0.29  0.45           1.00     -0.07
## magnesium            0.26      -0.05  0.29          -0.07      1.00
## tot. phenols         0.28      -0.33  0.13          -0.32      0.21
## flavonoids           0.23      -0.41  0.11          -0.35      0.19
## non-flav. phenols   -0.15       0.29  0.19           0.36     -0.25
## proanth              0.13      -0.22  0.01          -0.19      0.23
## col. int.            0.55       0.25  0.26           0.02      0.20
## col. hue            -0.08      -0.56 -0.08          -0.27      0.05
## OD ratio             0.06      -0.37  0.00          -0.27      0.05
## proline              0.64      -0.19  0.22          -0.44      0.39
##                   tot. phenols flavonoids non-flav. phenols proanth
## alcohol                   0.28       0.23             -0.15    0.13
## malic acid               -0.33      -0.41              0.29   -0.22
## ash                       0.13       0.11              0.19    0.01
## ash alkalinity           -0.32      -0.35              0.36   -0.19
## magnesium                 0.21       0.19             -0.25    0.23
## tot. phenols              1.00       0.86             -0.45    0.61
## flavonoids                0.86       1.00             -0.54    0.65
## non-flav. phenols        -0.45      -0.54              1.00   -0.36
## proanth                   0.61       0.65             -0.36    1.00
## col. int.                -0.06      -0.17              0.14   -0.03
## col. hue                  0.43       0.54             -0.26    0.29
## OD ratio                  0.70       0.79             -0.50    0.51
## proline                   0.50       0.49             -0.31    0.33
##                   col. int. col. hue OD ratio proline
## alcohol                0.55    -0.08     0.06    0.64
## malic acid             0.25    -0.56    -0.37   -0.19
## ash                    0.26    -0.08     0.00    0.22
## ash alkalinity         0.02    -0.27    -0.27   -0.44
## magnesium              0.20     0.05     0.05    0.39
## tot. phenols          -0.06     0.43     0.70    0.50
## flavonoids            -0.17     0.54     0.79    0.49
## non-flav. phenols      0.14    -0.26    -0.50   -0.31
## proanth               -0.03     0.29     0.51    0.33
## col. int.              1.00    -0.52    -0.44    0.32
## col. hue              -0.52     1.00     0.57    0.23
## OD ratio              -0.44     0.57     1.00    0.31
## proline                0.32     0.23     0.31    1.00
```

PCA mit Singulärwertzerlegung

Wenden Sie die Singulärwertzerlegung auf die *skalierten* Weindaten an und inspizieren Sie im 'Environment'-Fenster die Struktur des Ergebnisses wines.svd. U ist eine n x a orthonormale Matrix, die die linken

Singulärvektoren enthält. a ist die Zahl der berücksichtigten Faktoren, die hier der Zahl der ursprünglichen Variablen entspricht. D enthält die Singulärwerte. V enthält p x a rechten Singulärvektoren, mit p gleich der Zahl der ursprünglichen Variablen.

```
wines.svd <- svd(wines.sc)
```

Die svg Funktion hat die ursprüngliche Datenmatrix in drei Komponenten zerlegt:

$X = (UD)V^T = TP^T$

Die Interpretation der Matrizen T, P, U, D und V ist einfach. Die Ladungen, Spalten in Matrix P (oder gleichwertig die rechten Einzelvektoren, Spalten in Matrix V) geben die Gewichte der ursprünglichen Variablen in den PCs an. Variablen, die sehr niedrige Werte in einer bestimmten Spalte von V haben, tragen nur sehr wenig zu dieser bestimmten latenten Variable bei. Die Scores, Spalten in T, bilden die Koordinaten im Raum der latenten Variablen. Anders ausgedrückt: Das sind die Koordinaten der Proben, wie wir sie aus unserer neuen PCA-Sicht sehen. Die Spalten in U geben die gleichen Koordinaten in normierter Form an. Sie haben Einheitsvarianzen, während die Spalten in T Werte aufweisen, die den Varianzen der einzelnen PC entsprechen.

Wie oben zu sehen, müssen wir zur Erstellung der Matrix T das Kreuzprodukt der Matrizen U und D bilden, was in R mit dem Operator %*% geschieht. Da uns die svd-Funktion d als Vektor der Diagonalelemente liefert, müssen wir vor der Operation noch die vollständige Matrix mit dem diag()-Commando bilden.

Schauen Sie sich D und diag(D) an.

Die Ladungen werden einfach durch den Vector v erhalten:

```
wines.scores <- wines.svd$u %*% diag(wines.svd$d)
wines.loadings <- wines.svd$v
```

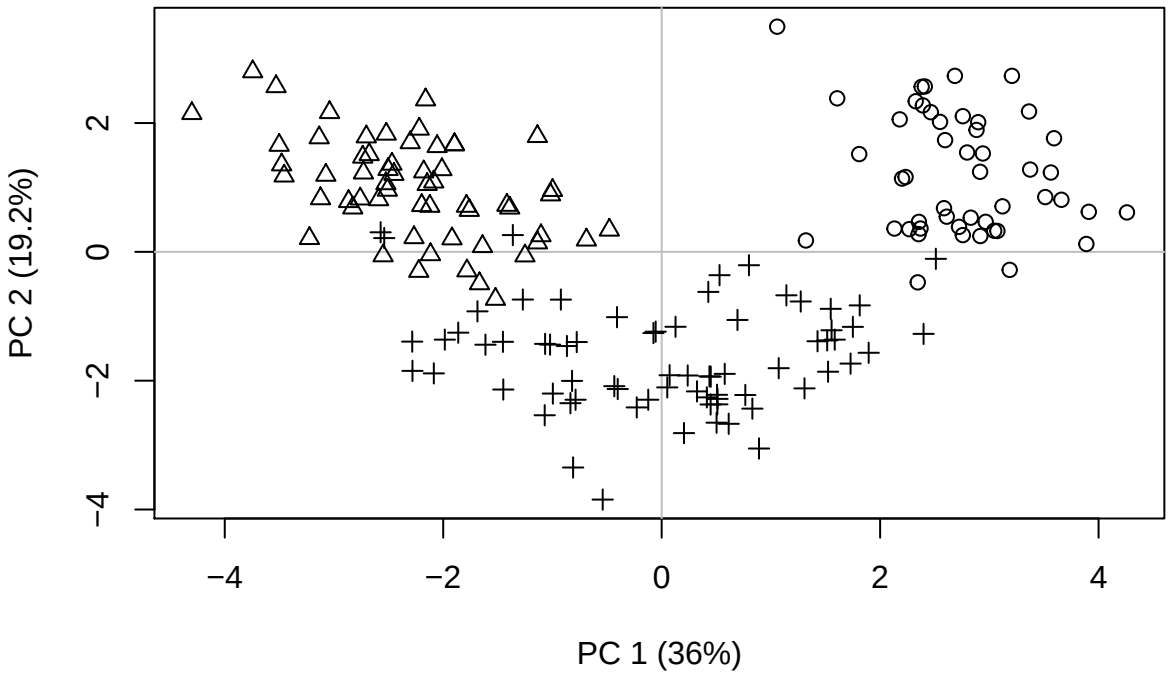
Die ersten beiden PCs stellen die Ebene dar, die den größten Teil der Varianz enthält; wie viel genau, ist durch die Quadrate der Werte auf der Diagonalen von D gegeben. Die Bedeutung der einzelnen PCs wird normalerweise durch den Prozentsatz der Gesamtvarianz angegeben, der erklärt wird.

```
wines.vars <- wines.svd$d^2 / (nrow(wines) - 1)
wines.totalvar <- sum(wines.vars)
wines.relvars <- wines.vars / wines.totalvar
variances <- 100 * round(wines.relvars, digits = 3)
variances[1:5]
```

```
## [1] 36.0 19.2 11.2  7.1  6.6
```

Nun haben wir alle Voraussetzungen, um die ersten beiden Hauptkomponenten PC1 und PC2 in einem 2D-Plot anzuzeigen:

```
plot(wines.scores[,1:2], type = "n", xlab = paste("PC 1 (", variances[1], "%)", sep = ""), ylab = paste
abline(h = 0, v = 0, col = "gray")
points(wines.scores[,1:2], pch = wine.classes)
```

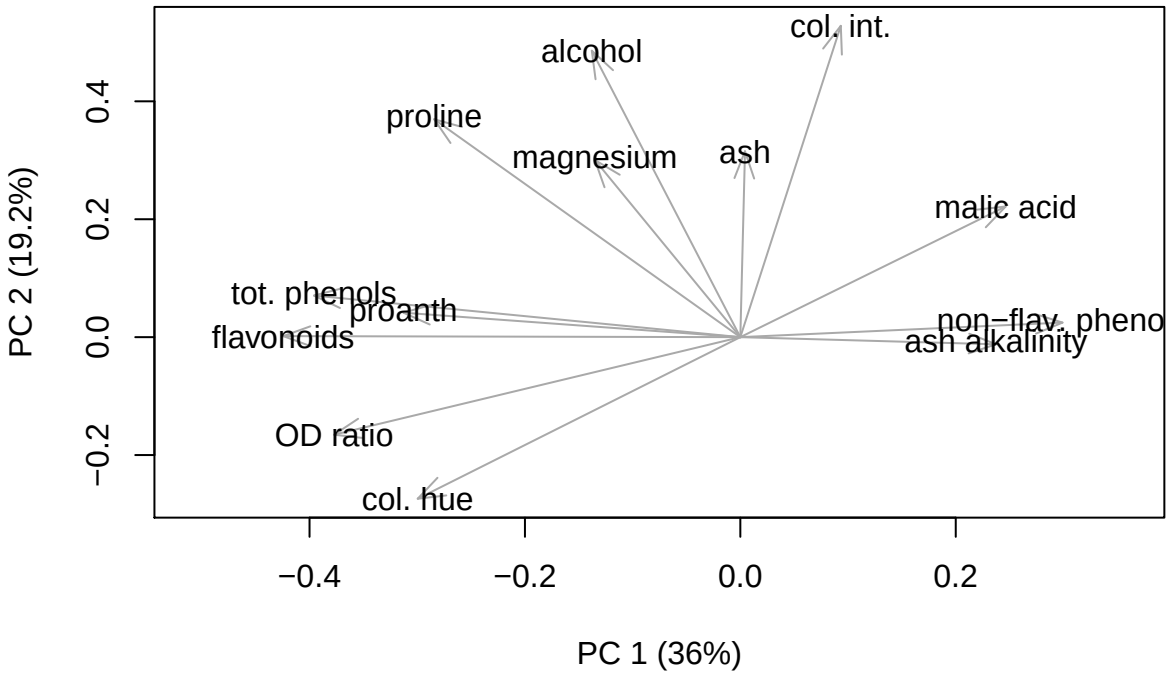


Studieren das obige Beispiel zum Plotten dieser PCA.

Sorgen Sie dafür, dass sich die drei Weinklassen auch noch farblich unterscheiden.

Wie schon mehrfach erwähnt, sind die Hauptkomponenten Linearkombinationen der ursprünglichen Variablen. Wie die ursprünglichen Variablen zu den neuen PC beitragen, können wir uns durch den Loadings-Plot verdeutlichen:

```
plot(wines.loadings[,1] * 1.2, wines.loadings[,2], type = "n", xlab = paste("PC 1 (", variances[1], "%)
arrows(0, 0, wines.loadings[,1], wines.loadings[,2], col = "darkgray", length = .15, angle = 20)
text(wines.loadings[,1:2], labels = colnames(wines))
```



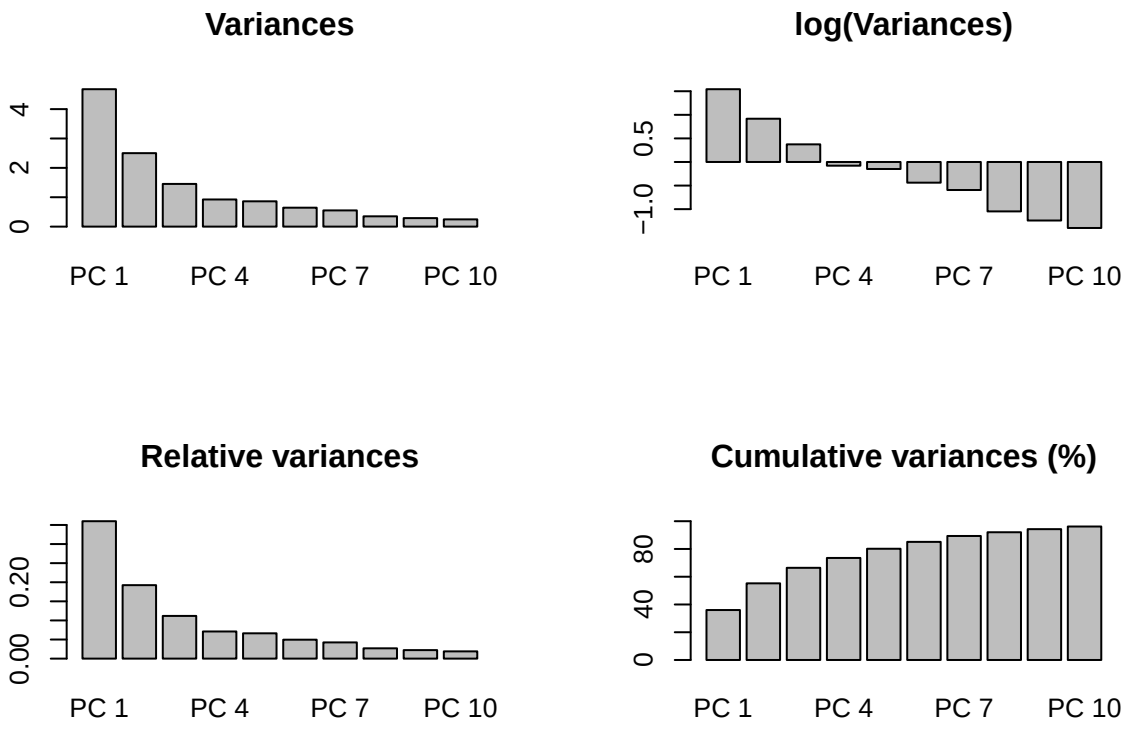
Der Faktor 1,2 im Plotbefehl wird verwendet, um Platz für die Textbeschriftungen zu schaffen. Die Weine der Klasse 3 zeichnen sich eindeutig durch niedrigere Alkoholwerte und eine geringere Farbintensität aus. Weine der Klasse 1 haben einen hohen Flavonoid- und Phenolgehalt und sind arm an nicht-flavonoiden Phenolen; das Gegenteil gilt für Weine der Klasse 2. Alle diese Schlussfolgerungen hätten wahrscheinlich auch durch die Betrachtung klassenspezifischer Boxplots für alle Variablen gezogen werden können. Die Kombination aus einem Principal Component Analysis Score Plot und einem Loading Plot zeigt dies jedoch wesentlich einfacher und stellt sogar direkte Informationen über Korrelationen zwischen Variablen und Objekten dar.

Wieviele Hauptkomponenten sollten wir berücksichtigen?

Die Frage, wie viele PCs man in Betracht ziehen oder anders ausgedrückt: Wo die Informationen aufhören und der Lärm beginnt, ist schwer zu beantworten. Viele Methoden berücksichtigen die Höhe der Varianz erklärt, und verwenden statistische Tests oder grafische Methoden, um zu definieren, welche PCs einbezogen werden sollen. Die Höhe der Abweichung pro PC wird in der Regel in einem Scree-Diagramm dargestellt: entweder die Abweichungen selbst oder die Logarithmen der Abweichungen werden als Balken dargestellt. Oftmals wird auch der Anteil der Gesamtvarianz berücksichtigt, der von jedem einzelnen PC erklärt wird. Die letzten PCs enthalten in der Regel keine Informationen und neigen vor allem auf einer Log-Skala dazu, die Scree-Plot weniger interpretierbar zu machen, so dass sie in der Regel nicht im Plot berücksichtigt werden.

Studieren Sie den folgenden Code:

```
par(mfrow = c(2,2))
barplot(wines.vars[1:10], main = "Variances",
names.arg = paste("PC", 1:10))
barplot(log(wines.vars[1:10]), main = "log(Variances)",
names.arg = paste("PC", 1:10))
barplot(wines.relvars[1:10], main = "Relative variances",
names.arg = paste("PC", 1:10))
barplot(cumsum(100 * wines.relvars[1:10]),
main = "Cumulative variances (%)",
names.arg = paste("PC", 1:10), ylim = c(0, 100))
```



Die PCs 1 und 2 erklären deutlich mehr Varianz als die anderen: zusammen decken sie 55% der Varianz ab. Die Scree-Plots zeigen keinen klaren Schnitt, was im wirklichen Leben eher die Regel als die Ausnahme ist. Je nach Ziel der Untersuchung könnte man für diese Daten drei oder fünf PCs in Betracht ziehen. Die Wahl von vier PCs wäre in diesem Fall wenig sinnvoll, da die fünfte PC fast die gleiche Varianz erklären würde: Wenn die vierte einbezogen wird, sollte auch die fünfte sein.