



## WERKZEUGE MUSTERERKENNUNG & MASCHINELLES LERNEN

# Aufgabenblatt 6

(Ausgabe am Fr 25.5.2018 — Abgabe bis So 3.6.2018)

### Aufgabe 1

8 P

Für die Binärbilder (Grauwertmatrizen mit Nullen und Einsen) in `box.rda` (Webseite zur Übung) soll die fraktale Dimension (ME-Skript V.4, Blatt 12) als Geradengefälle nach dem logarithmierten Potenzgesetz geschätzt werden.

- (a) Implementieren Sie eine Funktion `zoom(x,delta=1)`, welche die Binärbildmatrix `x` um den ganzzahligen Faktor `delta` ( $\delta \in \mathbb{N}$ ) verkleinert. Dazu wird `x` in Quadrate zu  $\delta \times \delta$  Originalpixel zerlegt, wobei etwaige Reste am rechten und unteren Bildrand eliminiert werden („clipping“). Im vergrößerten Bild erhält jedes  $\delta \times \delta$ -Quadrat den Wert Eins genau dann, wenn **alle** seine Originalpixel den Wert Eins hatten.

*TIPP:* Lösen Sie die Aufgabe ohne Laufschleifen, sondern mit einer der vektoriellen Funktionen `apply` oder `tapply` oder mit 3D-Arithmetik und `aperm`. Erledigen Sie das „clipping“ gleich zu Beginn.

- (b) Testen Sie die Funktion, indem Sie das Bild `Couple` auf einer  $3 \times 3$ -Leinwand in den Verkleinerungsstufen  $\delta \in \{1, 2, \dots, 9\}$  visualisieren.
- (c) Implementieren Sie eine Funktion `slope(x,y)`, welche die Steigung  $b$  der Ausgleichsgeraden  $y = a + b \cdot x$  durch die Punkte  $(x_1, y_1), \dots, (x_n, y_n)$  der Argumentvektoren `x`, `y` berechnet. Zur Lösung reicht ein einfacher Aufruf der 'R'-Funktion `lm()`. Testen Sie die Funktion mit den Daten `x=-5:+5` und `y=pi*x+sin(x)`.
- (d) Implementieren Sie eine Funktion `boxdim(x,d)` zur Schätzung der Boxdimension von `x`. Dazu wird für jedes  $\delta$  im Bereich `1:d` das Volumen (Objektgröße in Pixel) des  $\delta$ -gezoomten Bildes berechnet und daraus die Dimension als Geradengefälle der Raster-Volumen-Paare.
- (e) Erweitern Sie `boxdim()` um ein drittes Argument `plot=FALSE` zum Einschalten einer Grafikausgabe mit den oben genannten Punkten in doppelt-logarithmischen Koordinaten und der errechneten Boxdimension als `mtext()`-Beschriftung.
- (f) Abschließend erzeugen Sie eine Folge von  $2 \times 2$ -Leinwänden mit allen zwölf Binärrmatrizen aus `box.rda` je als S/W-Bild und als `boxdim()`-Darstellung mit `d=8`. Iterieren Sie über die Bildobjektnamen aus `load(box.rda)` und verwenden Sie die 'R'-Funktion `get()`.

Abzugeben ist lediglich Ihr 'R'-Programmcode `pca.R`.

## Aufgabe 2

12 P

In dieser Aufgabe geht es um *etikettierte* Merkmalsdaten, ihre graphische Darstellung und ihre Transformation nach Karhunen-Loève (PCA, ME-Skript V.5).

Wir stellen Merkmalsdaten in 'R' als `data.frame` mit  $N + 1$  Spalten dar; jede Zeile entspricht einem Muster; die Spalten  $1, 2, \dots, N$  enthalten die Merkmalwerte (Typ `numeric`) und die letzte Spalte zeigt die wahre Klassenzugehörigkeit (Typ `factor`) an.

- (a) Laden Sie den Irisdatensatz mit dem Kommando `data(iris)` und lesen Sie die sieben Datensätze aus `load('pca.rda')` ( $\leadsto$  Aufgabenwebseite) ein.
- (b) Schreiben Sie eine Grafikausgabefunktion `plot_lfd(x, ...)` zur Scatterplotdarstellung (siehe `?plot.data.frame`) der multivariaten Datensätze. Die Punkte der Zeichnung sind nach Klassenzugehörigkeit einzufärben. Es bezeichne `x` den Datensatz (mit Klassenfaktor in der letzten Spalte) und `...` die an `plot` zu delegierende Restparameterliste.
- (c) Testen Sie `plot_lfd()` mit den vier Teildatensätzen `iris[c(j:4,5)]` für  $j = 1, 2, 3, 4$ .
- (d) Schreiben Sie nun eine Konstruktorfunktion `PCA(x, n=?)`, die ein Objekt der Klasse `PCA` erzeugt mit den Listenelementen `mean`, `eigenval`, `eigenvec` für den Mittelwertvektor und die ersten `n` Eigenwerte bzw. Eigenvektoren des Datensatzes `x`. In der Voreinstellung für Argument `n` sollen *alle* Hauptachsen eingespeichert werden.
- (e) Dann schreiben Sie für die neue Klasse eine Funktion `predict.PCA(o, newdata)`, die auf den Eingabedatensatz `newdata` die (i.a. unvollständige) Hauptachsentransformation des `PCA`-Objekts `o` anwendet. Die Eingabevektoren `x` sind also gemäß der Skriptformel  $\mathbf{D}^{-1/2} \cdot \mathbf{U}^\top \cdot (\mathbf{x} - \boldsymbol{\mu})$  zu zentrieren, zu rotieren und dann zu skalieren. Achtung! Die Eingabe `newdata` und auch die Rückgabe sind `data.frame`-Objekte; der Klassenfaktor ist von Eingabe zu Ausgabe durchzuschleusen!  
  
Brauchbare 'R'-Funktionen für den Konstruktor sind `colSums`, `cov` und `eigen`, für den Prädiktor z.B. `apply`, `sweep` oder `scale`.
- (f) Erzeugen Sie nun zur Kontrolle die Grafikausgaben `plot_lfd(predict(PCA(iris,j), iris))` mit  $j = 4, 3, 2, 1$  für die vier möglichen (un)vollständigen Transformationen.
- (g) Starten Sie eine  $(2 \times 2)$ -Leinwand und visualisieren Sie nun die Transformatierten `predict(PCA(ldata, n=2), tdata)`. Für `ldata` und `tdata` setzen Sie wahlweise `iris` ein und den Teildatensatz `iris.part`, der lediglich die 50 *setosa*-Muster enthält.
- (h) Starten Sie nun eine Schleife über die acht Datensätze (inklusive `iris`) mit je einer  $(2 \times 2)$ -Leinwand und den vier Scatterplots für (1) die beiden ersten Originalmerkmale, (2) die beiden letzten Originalmerkmale, (3) die beiden ersten Hauptkomponentenmerkmale, (4) die beiden letzten Hauptkomponentenmerkmale.

Abzugeben ist die Datei `pca.R` mit Ihrem Programmcode.

## Hinweise zum Übungsablauf

---

- ➡ Die wöchentliche WMM-Vorlesung findet am Mittwoch um 12:15 Uhr statt.  
Das Aufgabenblatt gibt es immer am Freitag (PDF im Netz).  
Der späteste Abgabetermin ist Sonntag 23:59 Uhr.
- ➡ Die Übungsaufgaben dürfen natürlich (und sollten sogar) in Gruppenarbeit (2 Mitglieder) gelöst werden.
- ➡ Schriftliche Lösungen („Textantworten“) sind als PDF beizufügen oder direkt im e-Mail-Textkörper unterzubringen.
- ➡ Alle anderen Lösungen (Programmieraufgaben, Daten und Grafiken) sind als elektronischer Anhang der Lösungs-e-Mail abzuliefern.
- ➡ Programmcode (Dateien \*.R) muss auch wirklich in 'R' ausführbar sein.  
(Kommando `Rscript <name>.R` auf einem der Rechner des FRZ-Pools)
- ➡ Ganz wichtig:  
Schriftliche Antworten werden von mir gedruckt, gelesen, kommentiert und korrigiert.  
Deshalb diese Textteile bitte **niemals** im abgegebenen Programmcode verstecken!
- ➡ Je Gruppe und je Aufgabenblatt ist **genau eine** e-Mail zu senden:
  - Vermerk »WMM/*n*« und Gruppenname im **subject**-Feld  
(*n* ∈ ℕ ist die laufende Nummer des Übungsblattes)
  - die Namen der beteiligten Gruppenmitglieder im Textrumpf
  - Tabellen, Bilder, Programmcode, Sensordaten als Attachments  
(elektronische Anlagen)
  - etwaige schriftliche Antworten im Textrumpf der Post oder als Attachment  
(Text/PDF)
- ➡ Einige Aufgabentexte verweisen Sie zum Nachschlagen von Details auf das Folien-skript zur Vorlesung Mustererkennung; Sie finden es unter der URL  
<http://www.minet.uni-jena.de/fakultaet/schukat/ME/Scriptum/>.  
Die Angabe *ME-Skript II.6* bedeutet: Kapitel II, Abschnitt 6