



WERKZEUGE MUSTERERKENNUNG & MASCHINELLES LERNEN

Aufgabenblatt 2

(Ausgabe am Fr 27.4.2018 — Abgabe bis So 6.5.2018)

Aufgabe 1

10 P

Sie benötigen für diese Aufgabe zwei Grauwertbilder, die unter den Objektnamen `algae` und `tonga` zusammen mit der Ausgabefunktion `plot.array` in der 'R'-Objektedatei `binarize.rda` abgespeichert sind.

- (a) Lesen Sie die Kommandohilfe zur `hist`-Funktion und erzeugen Sie zum Bild `algae` das Grauwerthistogramm in 25 Stufen; Bild und Histogramm zeichnen Sie bitte in eine (2×2) -Grafik.
- (b) In das dritte und vierte Feld der (2×2) -Grafik fügen Sie bitte noch Grauwertbild und Histogramm für einen „einzelligen“ Bildausschnitt `alga` (Zeilen 210–260, Spalten 60–110) von `algae` ein.
- (c) Schreiben Sie eine Funktion `binarize(x, method='fixed', threshold=0.5)`, die das zur Schwelle `threshold` binarisierte Grauwertbild `x` erzeugt. Füllen Sie nun eine (3×3) -Leinwand mit den Binärbildern der Einzelzelle `alga` zu neun Schwellen zwischen 0.2 und 0.5.
- (d) Kompletтировать Sie nun Ihre Funktion `binarize()`. Bei Aufruf mit `method='mean'` bzw. `method='median'` soll als Schwelle Mittelwert oder Median aller Bildgrauwerte verwendet werden, bei `method='inter'` jedoch die Schwelle des Intermeans-Algorithmus (ME-Skript II.6, Blatt 12). Verwenden Sie dazu einfach die 'R'-Funktionen `mean()`, `median()` bzw. `kmeans()`.
- (e) Erzeugen Sie nun für jedes der drei Bilder `alga`, `algae`, `tonga` eine (2×2) -Grafik mit den Binärbildern zur Schwelle 0.5 und den drei gerade implementierten Varianten.

TIPP: Versehen Sie die Funktion `binarize` mit einem vierten Parameter `plot=FALSE`, der bei Belegung `plot=TRUE` die Grafikausgabe des produzierten Binärbildes bewirkt.

Abzuliefern ist Ihre 'R'-Programmdatei `binarize.R`.

Aufgabe 2

10 P

Wir betrachten einen ganz einfachen linearen B -bit-Quantisierer $q : \mathbb{R} \rightarrow \mathbb{R}$ (ME-Skript II.5, Blatt 9), der das Einheitsintervall $[0, 1]$ in 2^B gleich breite Zellen unterteilt und jeden Eingabewert auf seinen Zellenmittelpunkt abbildet. Übersteuernde Eingaben $x < 0$ und $x > 1$ werden wie die niedrigste bzw. höchste Zelle behandelt. Eine Beispielgrafik für das beschriebene Verhalten findet sich auf der Aufgabenwebseite.

- (a) Realisieren Sie $q(\cdot)$ als vektorisierte 'R'-Funktion `quantize(x, bits=8)`.
HINWEIS: Rufen Sie z.B. `?floor` oder `?round` und `?ifelse` auf!
- (b) Testen Sie die Funktion mit den Beispieldaten `sin(seq(0,pi,length=1500))`.
- (c) Schreiben Sie eine Funktion `SNR(x,y)`, die zum Datenvektor \mathbf{x} und seiner Quantisierung \mathbf{y} den Signal-Rausch-Abstand (Quotient Signalenergie/Störenergie) in Dezibel errechnet. Realisieren Sie zu diesem Zweck eine Hilfsfunktion `decibel(x)`, welche die Signalintensität einer Abtastfolge \mathbf{x} in Dezibel errechnet.
- (d) Wie groß ist die SNR der Beispieldaten bei Quantisierung mit $B = 8$ bzw. $B = 12$ Bit?
- (e) Wiederholen Sie beide Berechnungen (8/12 Bit) mit uniform verteilten Zufallsdaten (1500 Werte, Intervall $[0, 1]$); 'R'-Dokumentation unter `?runif`.
- (f) Wiederholen Sie die Berechnungen mit normalverteilten Zufallsdaten; 'R'-Dokumentation unter `?rnorm`. Wieder sind 1500 Werte zu erzeugen, die im Sinne vierfacher Standardabweichung (also $\mu = \frac{1}{2}$ und $\sigma = \frac{1}{8}$) im Intervall $[0, 1]$ liegen.
- (g) Wiederholen Sie die Berechnungen mit den äquidistanten Punkten `seq(from=0, to=1, length=1500)`.
- (h) Und nun wiederholen Sie die Berechnungen ein letztes Mal, wieder mit uniform verteilten Zufallsdaten, aber diesmal im Intervall $[0, 1.05]$.
- (i) Führen Sie abschließend eine kleine Versuchsreihe ($B = 8$ Bit) mit normalverteilten Zufallszahlen durch; erzeugen Sie jeweils 1500 Werte mit Verteilung $\mathcal{N}(\mu = \frac{1}{2}, \sigma = \frac{1}{2C})$, aber (außer $C = 4$ wie vorhin) mit 50 Einstellungen für C zwischen $C = 1$ und $C = 7$. Plotten Sie die 50 SNR-Werte in Abhängigkeit von C und deuten Sie die Kurvenform.

Abzuliefern sind die Tabelle der zehn SNR-Werte, Ihre 'R'-Funktionen (s.o.) zur Berechnung (`quantize.R`), die SNR-Graphik (`quantize.pdf`) sowie eine kurze Kommentierung der Ergebnisse — vielen Dank!

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

WWW: <http://www.minet.uni-jena.de/www/fakultaet/schukat/WMM/SS18>
e-Mail: EG.Schukat-Talamazzini@uni-jena.de