Arbeitsblatt 2

Aufgabe 1: Distanzen

Wir betrachten die synthetischen Daten mit den beiden Variablen x_1 und x_2.

Tabelle 1: Datensatz

	A	B	C	D	E	F	G	H	I	J	K	L
x1	10.0	8.0	11.0	9	12.0	45.0	55.0	59.0	52.0	56	58.0	57.0
x2	2.8	5.2	4.1	10	6.5	5.2	5.6	1.2	9.5	11	8.5	4.9

a) Lesen Sie die Beschreibung der R-Funktionen `dist(...)` und `daisy(...)` aus dem R-Package `cluster` durch. Bestimmen Sie die euklidischen Distanzen zwischen dem Objekt A und den Objekten D, F und G mit Hilfe von `dist(...)` oder `daisy(...)`.

Sie müssen dazu zuerst das Paket `cluster` installieren und laden (`library(cluster)`). Den Datensatz können Sie wie folgt erzeugen.

```
D.S <- data.frame(x1=c(10, 8, 11, 9, 12, 45, 55, 59, 52, 56, 58, 57),
                  x2=c(2.8, 5.2, 4.1, 10, 6.5,5.2,5.6,1.2, 9.5,11, 8.5, 4.9))
rownames(D.S) <- LETTERS[1:nrow(D.S)]
```

b) Bestimmen Sie die mit der Standardabweichung skalierten euklidischen Distanzen zwischen dem Objekt A und den Objekten D, F und G mit Hilfe von `daisy(...)`.

c) Ergänzen Sie den ursprünglichen Datensatz um eine Faktorvariable und berechnen Sie danach erneut die Distanzen zwischen dem Objekt A und den Objekten D, F und G mit Hilfe von `daisy(...)`.

```
D.S2 <- data.frame(x1=c(10, 8, 11, 9, 12, 45, 55, 59, 52, 56, 58, 57),
                   x2=c(2.8, 5.2, 4.1, 10, 6.5,5.2,5.6,1.2, 9.5,11, 8.5, 4.9),
                   x3=c("a", "a", "a", "b", "b", "b", "c", "c", "c", "d", "d", "d"))
rownames(D.S2) <- LETTERS[1:nrow(D.S2)]
```

d) Behandeln Sie die Variable x1 wie eine ordinale Variable und berechnen Sie danach noch einmal die Distanzen zwischen dem Objekt A und den Objekten D, F und G (`daisy(D.S2, metric="gower", type=list(ordratio=1))`). Was geschieht hier?

Aufgabe 2: Metrische MDS

Laden Sie den Datensatz `voting_NR.rdata`. Es enthält zwei Datensätze: `NR_voting` ist eine "Distanz"-Matrix, welche das Abstimmungsverhalten des Schweizer Parlaments (Nationalrats) beschreibt. Die Elemente in der Distanzmatrix zeigen, wie ähnlich bzw. unähnlich (Distanz) die Parlamentarierinnen und Parlamentarier abstimmen. Der Datensatz enthält Informationen zu 199 Leuten. Die Distanzmatrix wurde basierend auf 921 Abstimmungen im Jahr 2019 erstellt (vor den Erneuerungswahlen). Die Distanz zwischen zwei Nationalräten entspricht jeweils der mittleren absoluten Differenz in allen Abstimmungen bei welchen beide anwesend waren. Ja-Stimmen wurden dabei mit 1 codiert, Nein-Stimmen mit 0, Enthaltungen mit 0.5. Der zweite Datensatz `NR_meta` enthält Metainformationen (Fraktion und Kanton) aller Nationalräte. Die Daten sind gleich sortiert.

a) Checken Sie mit einer 2-dimensionalen Visualisierung, ob Mitglieder der gleichen Fraktion ähnlich abstimmen. Verwenden Sie dazu die metrische MDS-Methode (`cmdscale`). Um die Fraktionszugehörigkeit farblich zu visualisieren, können Sie in der Plot-Funktion das Argument `col` z.B. wie folgt parametrisieren:

```
col=c("yellow", "orange", "green", "black", "blue", "red", "darkgreen")[NR_meta$Fraktion]`
```

b) Wiederholen Sie Ihre Analyse mit der nicht-metrischen MDS (Funktion `isoMDS` aus dem Paket `MASS`). Gibt es grosse Unterschied in der Visualisierung?

Aufgabe 3: MNIST

Wir betrachten noch einmal den MNIST-Datensatz. Dieses Mal mit 2000 zufällig ausgewählten handgeschriebenen Nummern von 0 bis 9. Nach dem Laden des Datensatzes `mnist_2k.Rdata` steht das Dataframe `x` zur Verfügung. Jede Zeile des Dataframes enthält die linearisierten Pixel-Grauwerte eines Bildes `pixel0` bis `pixel783`. Zudem gibt es eine Spalte `label` mit den Labeln.

Machen Sie sich mit dem Datensatz vertraut. Folgender Code hilft Ihnen dabei:

```r
dim(x)
x$label <- as.factor(x$label)
table(x$label)
image_nummer <- 7 # hier können Sie ein Bild von 1 bis 2000 auswählen
bild <- matrix(as.numeric(x[image_nummer, 2:785]), ncol=28, byrow = TRUE)
image(t(bild[28:1,1:28]))
x$label[image_nummer]
```

a) Führen Sie eine Hauptkomponentenanalyse durch und visualisieren Sie das Ergebnis in einem 2D Score-Plot (beachten Sie, dass Sie die Spalte mit den Labels nicht für die PCA verwenden). Verwenden Sie Farben, um die verschiedenen Nummern zu visualisieren. Sehr einfach geht das mit der Funktion `autoplot(res_pca, data=x, colour= 'label')` aus dem Paket `ggfortify`.

b) Visualisieren Sie die Daten auch mittels metrischer MDS. Achtung die Distanzberechnung braucht etwas Zeit. Vergleichen Sie das Ergebnis mit den Erkenntnissen aus a). Verwenden Sie zur Visualisierung folgenden Code:

```r
d_cmd <- as.data.frame(res.cmd$Y) #res_cmd -> Ergebnis MDS
names(d_tsne) <- c("x1","x2")
d_cmd$label <- x$label
library(ggplot2)
ggplot(data = d_cmd, aes(x = x1, y = x2, col = label)) +
  geom_point() +
  ggtitle("2D MDS Visualisierung")
```

c) Visualisieren Sie die Daten auch noch mit t-SNE (Funktion `Rtsne` aus Paket `Rtsne`). Vergleichen Sie das Ergebnis mit den Erkenntnissen aus a) und b). Verwenden Sie zur Visualisierung hier folgenden Code:

```r
library(ggplot2)
d_tsne <- as.data.frame(res_tSNE$Y) #res_tSNE -> Ergebnis Rtsne
names(d_tsne) <- c("tsne1","tsne2")
d_tsne$label <- x$label
ggplot(data = , aes(x = tsne1, y = tsne2, col = label)) +
  geom_point()
```

Aufgabe 4: Leistungsnachweis

- Visualisieren Sie Ihren Datensatz (oder Teile davon) mittels eines geeigneten Dimensionsreduktionsansatzes Ihrer Wahl.
- Interpretieren Sie Ihr Ergebnis.