

## Übungsblatt 16

**Übung 16.1** Machen Sie dasselbe wie in Übung 15.1, aber mit einer LDA statt PCA auf zwei Features Ihrer Wahl. Berechnen Sie die Diskriminantenrichtung  $\vec{w}$  sowohl mit der Formel

$$\vec{w} = \frac{S_W^{-1}(\vec{m}_1 - \vec{m}_2)}{\|S_W^{-1}(\vec{m}_1 - \vec{m}_2)\|}$$

als auch mit der Funktion *lda* aus dem Package *MASS*. Es sollte sich eigentlich dasselbe Ergebnis ergeben; warum ist das Ergebnis trotzdem scheinbar verschieden?

Hinweise:

- Die *within Scatter Matrix*  $S_W$  können Sie wieder mit dem bereitgestellten Script *scatter.matrices.r* berechnen.
- Die Inverse einer Matrix berechnet die Funktion *solve*.
- Die euklidische Norm eines Vektors  $x$  können Sie berechnen mittels `sqrt(sum(x^2))`
- Die Funktion *lda* gibt eine benannte Liste mit verschiedenen Informationen zurück (die *names* Funktion gibt die Namen der Listeneinträge aus). Die Diskriminantenrichtungen (für  $C$  Klassen gibt es davon  $C - 1$ ) stehen im Listeneintrag `..$scaling`.

**Übung 16.2** Bestimmen Sie für *alle drei* Klassen des Iris-Datenstazes und *alle vier Features* die beiden LDA-Richtungen mit der Funktion *lda* aus dem R-Paket *MASS*. Beachten Sie, dass *lda* die Richtungsvektoren *nicht normiert* zurückgibt, so dass diese noch normiert werden müssen mittels:

```
res <- lda(features, classes)
lda.normiert <- scale(res$scaling, center=FALSE,
                     scale=sqrt(colSums(res$scaling^2)))
```

Die Projektion der originalen Features auf die LDA-Richtungen können Sie berechnen mittels (machen Sie Sich klar, wie die beiden Matrizen aussehen):

```
as.matrix(iris[,-5]) %*% as.matrix(lda.normiert)
```