(2) Eigenanalyse

Ziel dieses Seminar ist es, die Begriffe der Eigenanalyse, Orthonormalzerlegung und Singulärwertzerlegung in **R** anhand der in der Vorlesung diskutierten Beispiele nachzuvollziehen.

Eigenwerte und Eigenvektoren

Wir betrachten das in der Vorlesung betrachtete Beispiel zur Berechnung von Eigenwerten und Eigenvektoren. In **R** erlaubt der Befehl `eigen()` die Eigenanalyse einer quadratischen Matrix.

```
# Matrixdefinition
A = matrix(c(2,1,
             1,2),
           nrow  = 2,
           byrow = TRUE)

# Eigenanalyse
EA = eigen(A)

# Eigenwerte von A
print(EA$values)
```

```
[1] 3 1
```

```
# Eigenvektoren von A
print(EA$vectors)
```

```
          [,1]       [,2]
[1,] 0.7071068 -0.7071068
[2,] 0.7071068  0.7071068
```

Für ein tieferes intuitives Verständnis können die Eigenvektoren einer 2×2 Matrix mithilfe folgenden **R** Codes visualisiert werden.

```
# PDF Speicherung der Abbildung
library(latex2exp)
pdf(
file         = "./2_Abbildungen_S/mv_2_eigenvektoren.pdf",
width        = 4,
height       = 4)

# Abbildungsparameter
par(
family       = "sans",
mfcol        = c(1,1),
pty          = "s",
bty          = "l",
lwd          = 1,
```

```
las         = 1,
mgp         = c(2,1,0),
xaxs        = "i",
yaxs        = "i",
font.main   = 1,
cex         = .95,
cex.main    = 1.2)

# Plotobjekt
plot(
NULL,
xlab        = TeX("$x_1$"),
ylab        = TeX("$x_2$"),
xlim        = c(-2.6,2.6),
ylim        = c(-2.6,2.6))
grid()

# Pfeildarstellung der Eigenvektoren von A
arrows(
x0          = c(0,0),
y0          = c(0,0),
x1          = c(EA$vectors[1,1], EA$vectors[2,1]),
y1          = c(EA$vectors[1,2], EA$vectors[2,2]),
angle       = 20,
length      = .1,
lwd         = 2,
col         = c("black", "black"),
xpd         = TRUE)

# Pfeildarstellung der Matrixprodukte Av_1, Av_2
Av_1        = A %*% EA$vectors[,1]
Av_2        = A %*% EA$vectors[,2]
print(Av_2)
print(Av_1)
arrows(
x0          = c(0,0),
y0          = c(0,0),
x1          = c(Av_1[2,1], Av_2[2,1]),
y1          = c(Av_1[1,1], Av_2[1,1]),
angle       = 20,
length      = .1,
lwd         = 1,
col         = c("gray", "gray"),
xpd         = TRUE)

# Annotation
text( 1/sqrt(2),  0.9, TeX("$v_1$"))
text( 1/sqrt(2), -0.9, TeX("$v_2$"))
text( 2/sqrt(2),  2.4, TeX("$Av_1 = \\lambda_1 v_1$"))
text(-1/sqrt(2),  0.9, TeX("$Av_2 = \\lambda_2 v_2$"))
dev.off()
```

Die resultierende Abbildungs ist als Abbildung 1 dargestellt.

Orthonormalzerlegung

Wie in der Vorlesung diskutiert, können die Ergebnisse der Eigenanalyse der Matrix

$$A := \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix} \tag{1}$$

aufgrund ihrer Symmetrie und bei Wahl von normierten Eigenvektoren für eine Orthonormalzerlegung von A genutzt werden. Diese Einsicht wird in folgendem **R** Code reproduziert.

```
# Eigenanalyse der Beispielmatrix
A   = matrix(c(2,1,
               1,2),
             nrow  = 2,
```

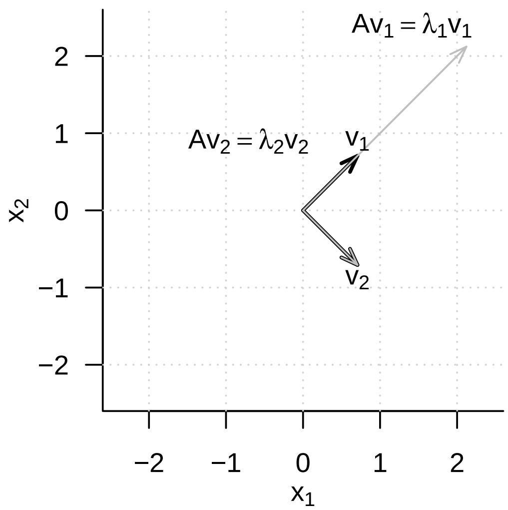

Abbildung 1. Visualisierung von Eigenvektoren einer 2×2 Matrix

```
             byrow = TRUE)
EA   = eigen(A)

# Überprüfung der Länge der von R ermittelten Eigenvektoren
cat("Laenge von v_1:", sqrt(t(EA$vectors[,1]) %*% EA$vectors[,1]))
```

```
Laenge von v_1: 1
```

```
cat("Laenge von v_2:", sqrt(t(EA$vectors[,2]) %*% EA$vectors[,2]))
```

```
Laenge von v_2: 1
```

```
# Definition der Orthonormalzerlegungsmatrizen
Q       = EA$vectors
Lambda  = diag(EA$values)

# Validierung der Orthonormalzerlegung von A
A       = Q %*% Lambda %*% t(Q)
print(A)
```

```
     [,1] [,2]
[1,]    2    1
[2,]    1    2
```

Singulärwertzerlegung

Zuletzt wollen wir den in der Vorlesung hergestellten Zusammenhang zwischen der Singulärwertzerlegung und der Eigenanalyse einer Matrix mithilfe des Singulärwertzerlegungsbefehls `svd()` an einem Beispiel verdeutlichen.

```
# Definition einer 2 x 3 Matrix Y
Y     = matrix(c(1,2,3,
                 4,5,6),
                 byrow = T,
                 nrow  = 2)
print(Y)
```

```
     [,1] [,2] [,3]
[1,]    1    2    3
[2,]    4    5    6
```

```
# Singulärwertzerlegung von Y  mithilfe von svd()
USV   = svd(Y)
U     = USV$u
S     = diag(USV$d)
V     = USV$v

# Eigenanalyse von YY^T mithilfe von eigen()
EAYYT  = eigen(Y %*% t(Y))
print(EAYYT$vectors)
```

```
          [,1]       [,2]
[1,] 0.3863177 -0.9223658
[2,] 0.9223658  0.3863177
```

```
print(EAYYT$values)
```

```
[1] 90.4026725  0.5973275
```

```
print(U)
```

```
           [,1]       [,2]
[1,] -0.3863177 -0.9223658
[2,] -0.9223658  0.3863177
```

```r
print(diag(S) ** 2)
```

```
[1] 90.4026725  0.5973275
```

```r
# Eigenanalyse von Y^TY mithilfe von eigen() und Fokus auf den von Null verschiedenen Eigenwerten
EAYTY  = eigen(t(Y) %*% Y)
print(EAYTY$vectors[,1:2])
```

```
           [,1]       [,2]
[1,] -0.4286671  0.8059639
[2,] -0.5663069  0.1123824
[3,] -0.7039467 -0.5811991
```

```r
print(EAYTY$values[1:2])
```

```
[1] 90.4026725  0.5973275
```

```r
print(V)
```

```
           [,1]       [,2]
[1,] -0.4286671  0.8059639
[2,] -0.5663069  0.1123824
[3,] -0.7039467 -0.5811991
```

```r
print(diag(S) ** 2)
```

```
[1] 90.4026725  0.5973275
```