(4) Deskription und Inferenz

Ziel dieses Seminar ist es, die Implementation der Berechnung multivariater Deskriptivstatistiken zu verdeutlichen und sich der Grundlagen Frequentistischer Inferenz zu besinnen.

Multivariate Deskriptivstatistiken

Wir simulieren zunächst einen Datensatz von $n = 30$ vierdimensionalen Messwerten.

```
# R Pakete
library(matrixcalc)                          # Matrix Paket (is.positive.definite())
library(MASS)                                # Multivariate Normalverteilung (mvrnorm())

# Simulationsparameter
set.seed(1)                                  # Reproduzierbare Randomisierung
m     = 4                                    # Datenpunktdimension
n     = 30                                   # Anzahl Realisierung
mu    = rep(0,m)                             # Erwartungswertparameter
Sigma = matrix(runif(m^2), nrow = m)         # zufällige Matrix
Sigma = 0.5*(Sigma+t(Sigma))                 # symmetrische Matrix
Sigma = Sigma + m*diag(m)                    # positiv definite Matrix

# Datensatzgeneration
Y     = t(mvrnorm(n,mu,Sigma))               # Y = (\ups_1,...,\ups_n) \sim N(\mu,\Sigma)
fname = "4_Deskription_und_Inferenz.csv"     # Dateiname
write.csv(Y, file = fname, row.names = FALSE)  # Datenspeichern
```

Wir laden nun die Datenmatrix und berechnen auf Grundlage des in der Vorlesung diskutierten Theorems zu Datenmatrix und multivariate Deskriptivstatistiken das Stichprobenmittel, die Stichprobenkovarianzmatrix und die Stichprobenkorrelationsmatrix

```
# Laden einer m x n Datenmatrix
fname   = "4_Deskription_und_Inferenz.csv"
Y       = as.matrix(read.table(fname, sep = ",", header = TRUE))

# Deskriptivstatisken
n       = ncol(Y)                                  # Anzahl Datenvektorealisierungen
I_n     = diag(n)                                  # Einheitsmatrix I_n
J_n     = matrix(rep(1,n^2), nrow = n)             # 1_{nn}
y_bar   = (1/n)* Y %*% J_n[,1]                     # Stichprobenmittel
C       = (1/(n-1))*(Y %*% (I_n-(1/n)*J_n) %*% t(Y))  # Stichprobenkovarianzmatrix
D       = diag(1/sqrt(diag(C)))                    # Kov-Korr-Transformationsmatrix
R       = D %*% C %*% D                            # Stichprobenkorrelationsmatrix

# Ausgabe
print(y_bar)
```

```
           [,1]
[1,] -0.20454286
[2,]  0.40088096
[3,] -0.31740484
[4,] -0.08424694
```

```
print(C)
```

```
          [,1]       [,2]       [,3]      [,4]
[1,]  3.9134127 -0.2422984  1.2068574 1.5554252
[2,] -0.2422984  2.9762937 -0.3353690 0.3849835
[3,]  1.2068574 -0.3353690  3.6424652 0.6982073
[4,]  1.5554252  0.3849835  0.6982073 3.4603723
```

```
print(R)
```

```
           [,1]        [,2]       [,3]      [,4]
[1,]  1.00000000 -0.07099616  0.3196542 0.4226782
[2,] -0.07099616  1.00000000 -0.1018562 0.1199617
[3,]  0.31965422 -0.10185616  1.0000000 0.1966642
[4,]  0.42267824  0.11996174  0.1966642 1.0000000
```

Schließlich berechnen wir die quadrierter Euklidische und die Mahalanobis Distanz des berechneteten Stichprobenmittels vom Nullpunkt basierend auf der Stichprobenkovarianzmatrix.

```
# Laden einer m x n Datenmatrix
fname   = "4_Deskription_und_Inferenz.csv"
Y       = as.matrix(read.table(fname, sep = ",", header = TRUE))

# Deskriptivstatisken
n       = ncol(Y)                                   # Anzahl Datenvektorealisierungen
I_n     = diag(n)                                   # Einheitsmatrix I_n
J_n     = matrix(rep(1,n^2), nrow = n)              # 1_{nn}
y_bar   = (1/n)* Y %*% J_n[,1]                      # Stichprobenmittel
C       = (1/(n-1))*(Y %*% (I_n-(1/n)*J_n) %*% t(Y))  # Stichprobenkovarianzmatrix

# Ausgabe
cat("Quadrierte Euklidische Distanz von 0_4", t(y_bar) %*% y_bar)
```

```
Quadrierte Euklidische Distanz von 0_4 0.3103867
```

```
cat("Mahalanobis Distanz von 0_4", t(y_bar) %*% solve(C) %*% y_bar)
```

```
Mahalanobis Distanz von 0_4 0.07736309
```

Grundlagen Frequentistischer Inferenz

Anhand untenstehender Simulation der Dualität von Konfidenzintervallen und Hypothesentests im univariaten Einstichproben-T-Test Szenario besinnen wir uns der intuitiven Grundlagen der Frequentistischen Inferenz.

```
# Modellformulierung
n       = 12                                    # Stichprobengröße
mu      = 2                                     # wahrer, aber unbekannter, Erwartungswertparameter
sigsqr  = 1                                     # wahrer, aber unbekannter, Varianzparameter

# Konfidenzintervallparameter und Testparameter
delta   = 0.95                                  # Konfidenzbedingung
t_delta = qt((1+delta)/2, n-1)                  # \Psi^{-1}((\delta + 1)/2, n-1)
mu_0    = mu                                    # Nullhypothesenparameter

# Simulationen
set.seed(1)                                     # random number generator seed
ns      = 1e5                                   # Anzahl Simulationen
y_bar   = rep(NaN,ns)                           # Stichprobenmittelarray
s       = rep(NaN,ns)                           # Stichprobenstandardabweichungarray
kappa   = matrix(rep(NaN,2*ns), ncol = 2)       # Konfidenzintervallarray
kfn     = rep(NaN,ns)                           # Überdeckungsindikatorarray
phi     = rep(NaN,ns)                           # Testarray
for(i in 1:ns){                                 # Simulationsiterationen

  # Stichprobenrealisation und Konfidezintervallevaluation
  y          = rnorm(n,mu_0,sqrt(sigsqr))       # Stichprobenrealisierung
  y_bar[i]   = mean(y)                          # Stichprobenmittel
  s[i]       = sd(y)                            # Stichprobenstandardabweichung
  kappa[i,1] = y_bar[i] - (s[i]/sqrt(n))*t_delta  # untere KI Grenze
  kappa[i,2] = y_bar[i] + (s[i]/sqrt(n))*t_delta  # obere KI Grenze

  # Überdeckungs- und Testevaluation
  if(kappa[i,1] <= mu_0 & mu_0 <= kappa[i,2]){
      kfn[i] = 1} else{kfn[i] = 0}              # Überdeckungsindikatorevaluation
  if(kappa[i,1] <= mu_0 & mu_0 <= kappa[i,2]){
      phi[i] = 0} else{phi[i] = 1}}             # Testevaluation

# Ausgabe
cat(   "Geschätztes Konfidenzniveau =", mean(kfn),
     "\nGeschätzter Testumfang      =", mean(phi))
```

```
Geschätztes Konfidenzniveau = 0.9507
Geschätzter Testumfang      = 0.0493
```

Wir visualieren die ersten 100 Simulationen als Abbildung 1.

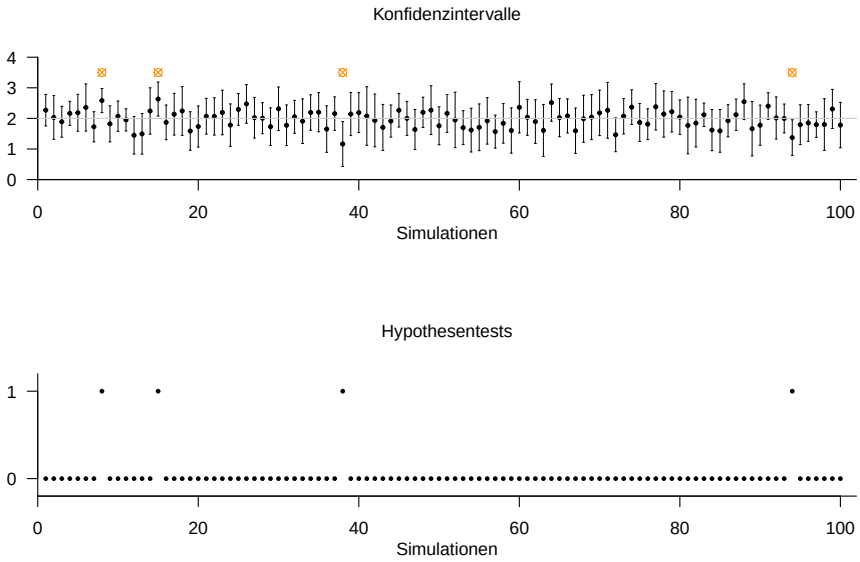

Abbildung 1. Grundlagen Frequentisticher Inferenz am Beispiel von Konfidenzintervallen und Hypothesentests im Einstichproben-T-Test Szenario