

Übung 01

Kennenlernen von R

INFI-IS

5xHWII

Albert Greinöcker

26.09.2021



CCA - COMPETENCE CENTRE

HTL Anichstraße

1 Einfache R-Anweisungen

1.1 Erzeugen von einfachen Zahlenfolgen und Zufallszahlen

Alle hier erzeugten Daten sollen in einer Variable gespeichert und ausgegeben werden

1.1.1 Wie können auf einfachem Weg die Zahlen von 100 bis 200 erzeugt werden?

Ergebnis:

```
a <- 100:200  
a
```

1.1.2 Wie können die Zahlen 100 bis 200 mit Abstand 2 (also 100, 102, 104,..) erzeugt werden?

```
b <- seq(100, 200, 2)
b
```

1.1.3 Erzeuge 100 Normalverteilte und 100 Gleichverteilte Zahlen

```
c <- rnorm(100)
c
d <- runif(100)
d
```

2 Einfache Berechnungen / Einfache Auswahl von Zahlen

Bitte für diesen Teil die normalverteilten Zahlen von 1.1.3 verwenden und die Ergebnisse wieder in einer **Variable** speichern.

- `x <- c`
- `y <- d`

2.1 Folgende Kennzahlen berechnen: Mittelwert, Median, Minimum, Maximum, Standardabweichung

```
xAverage <- mean(x)
yAverage <- mean(y)
xAverage
yAverage
#median
xMedian <- mode(x)
yMedian <- mode(y)
xMedian
yMedian
#minimum
xMin <- min(x)
yMin <- min(y)
xMin
yMin
#maximum
xMax <- max(x)
yMax <- max(y)
```

```
xMax  
yMAx  
#standard deviation  
xStandDeviat <- sd(x)  
yStandDeviat <- sd(y)  
xStandDeviat  
yStandDeviat
```

2.2 Alle Zahlen mit 100 multiplizieren

```
x100 <- x*100  
y100 <- y*100  
x100  
y100
```

2.3 Nur die ersten 10 Zahlen auswählen

```
xFirst10 <- x100[0:10]  
yFirst10 <- y100[0:10]  
xFirst10  
yFirst10
```

2.4 Nur die Werte auswählen, die größer 0 sind

```
xFirst10Higher0 <- xFirst10[xFirst10 > 0]  
yFirst10Higher0 <- yFirst10[yFirst10 > 0]  
xFirst10Higher0  
yFirst10Higher0
```

3 Einfache grafische Darstellungen

3.1 Erstellen eines einfachen Liniendiagramms aus den normalverteilten Zahlen von 1 . 1 . 3

```
plot(c, type="l")
```

So wird ein Bild eingefügt:

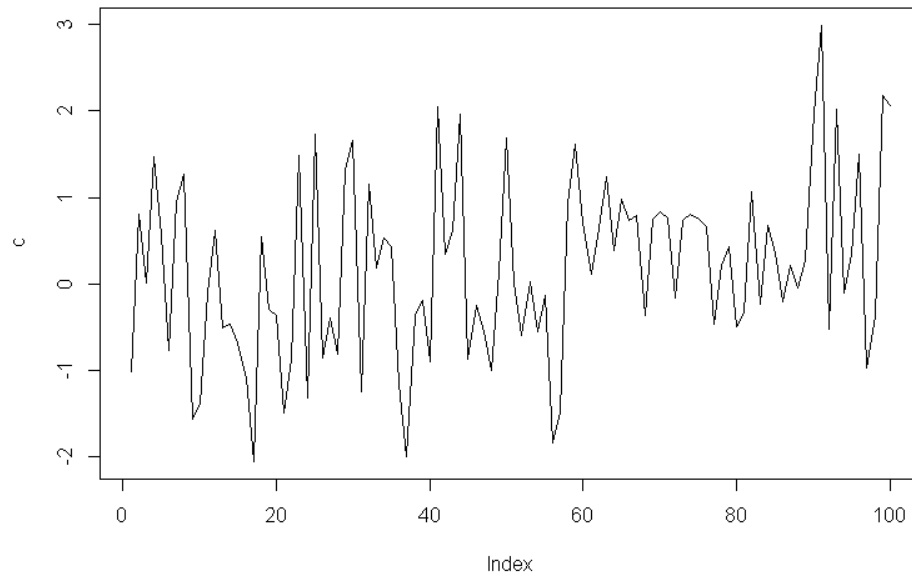
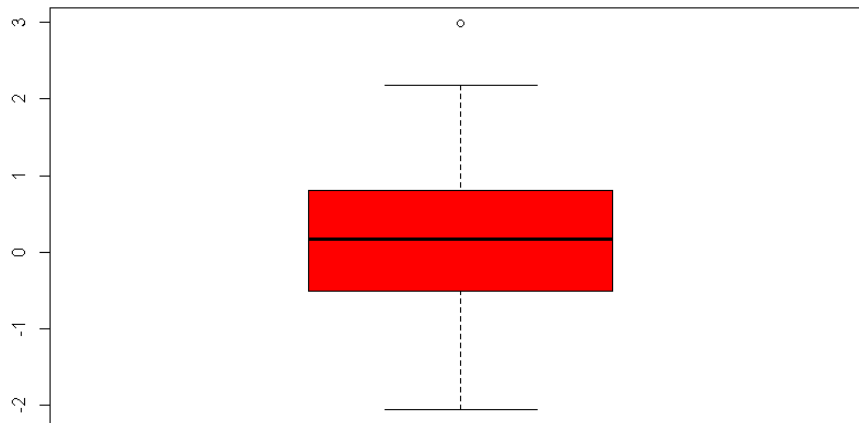


Figure 1: Figure caption.

3.2 Erstellen eines einfachen Boxplots dieser Zahlen

Der Boxplot soll rot eingefärbt werden

```
boxplot(c, col="red")
```



4 Komplexe Datentypen

4.1 Bitte eine Variable erzeugen, die die Wochentage als Factor speichert

```
dayOfWeek <- c("MON","TUE", "WED","THU", "FRI","SAT","SAN")
dayOfWeek
```

4.2 Diese soll verwendet werden, um einen Data Frame zu erzeugen, der zu jedem Wochentag die Anzahl der Stunden im regulären Stundenplan der Klasse abbildet, gemeinsam mit einer Variable, die die Position des Tages, also MO=1, DI=2, beinhaltet

```
hoursADay <- c(9,9,9,9,5,0,0)
numberOfWeekDay <- c(1,2,3,4,5,6,7)
dataFrameWeek <- data.frame(dayOfWeek,numberOfWeekDay,hoursADay)
hoursADay
numberOfWeekDay
dataFrameWeek
```

5 Kombination aus den oberen Bereichen

5.1 Erzeuge ein Data Frame, dass 2 Spalten mit jeweils 1000 Normalverteilten Werten beinhaltet

```
values1 <- rnorm(1000)
values2 <- rnorm(1000)
```

5.2 als 3. Spalte soll eine Variable Geschlecht mit den Ausprägungen 'm' und 'w' vergeben werden

Diese Werte sollen als Faktoren gespeichert werden

Hinweise:

- Mit dem Befehl **rep** kann man eine Liste erzeugen, mit dem ein Wert eine bestimmte Anzahl oft wiederholt wird
- Der Befehl **c** fügt mehrere Listen zu einer zusammen
- mit dem Befehl **cbind** kann man mehrere Spalten zusammenfügen, besser ist allerdings **data.frame**, da hier die Spalten unterschiedliche Datentypen haben können und Namen für die Spalten vergeben werden können.

```
sex <- c(rep('m',500),rep('w',500))
dataFrameCombi <- data.frame(values1, values2,sex)
dataFrameCombi
sex
```

5.3 Für die einzelnen Spalten sollen sprechende Bezeichnungen vergeben werden

```
names(dataFrameCombi)[1] <- "First norm"
names(dataFrameCombi)[2] <- "Second norm"
names(dataFrameCombi)[3] <- "Sex (male, female)"
names(dataFrameCombi)
```

5.4 Erzeuge eine Grafik, die einen Boxplot für 'm' und einen für 'w' enthält, den einen rot, den anderen grün

```
boxplot(dataFrameCombi$`First norm`~dataFrameCombi$`Sex (male, female)`,
col=c("blue","red"))
```

