Datenanalyse mit R

Schummelzettel

Kürzel für Schummelzettel

d – Ein Datensatz, in R ein Data Frame
v – ein einzelner Vektor, z.B. **d$Spaltenname**

Arbeitsverzeichnis

getwd()
Zeigt das aktuelle Arbeitsverzeichnis an.
setwd("C://dateipfad/arbeitsverzeichnis")
Verändert das aktuelle Arbeitsverzeichnis.

Datensatzverarbeitung

Einlesen

xlsx-Paket (Excel-Dateien)
odbc-Paket (SQL-Datenbanken

d <- read.table("dateiname.txt")
Liest Textdatei und speichert diese in **d**.
d <- read.csv ("dateiname.csv")
Liest .**csv**-Datei mit **,** als Spaltentrenner und speichert diese in **d**.

! Mit dem Parameter **encoding** kann die Zeichencodierung verändert werden
Bsp.: **encoding="UTF-8"** oder **encoding="latin1"**

d <- read.csv2("dateiname.csv")
Liest .**csv**-Datei mit **;** als Spaltentrenner und speichert diese in **d**.
load(file = "dateiname.RData")
Liest .**RData**-Datei, lädt die darin gespeichert Variablen. Überschreibt gleichlautende Variablen im Arbeitsplatz.
d <- readRDS(file ="dateiname.rds")
Liest .**rds**-Datei, speichert das darin enthaltene R-Objekt in **d**.

Exportieren

write.table(d, "dateiname.txt")
Schreibt Textdatei aus **d** und speichert diese in die Datei **dateiname.txt**.
write.csv (d, file = "dateiname.csv")
Schreibt .**csv**-Datei aus d und speichert diese in die Datei **dateiname.csv**.

! Mit Parameter **row.names = FALSE** wird keine Spalte mit Durchnummerierung der Zeilen erzeugt

save(d, file = "dateiname.RData")
Speichert **d** in die Datei **dateiname.Rdata** zum späteren Laden mit **load**.
saveRDS(d, file = "dateiname.rds")
Speichert einzelnes R-Objekt d in die Datei **dateiname.rds** zum späteren Laden per **readRDS()**.

Datensatzsuntersuchung/ -vorbereitung

nrow(d)	Zeilenanzahl
ncol(d)	Spaltenanzahl
dim(d)	Zeilenanzahl und Spaltenanzahl
colnames(d)	Spaltennamen
head(d)	erste 6 Zeilen
tail(d)	letzte 6 Zeilen
View(d)	Übersicht gesamter Datensatz

! Bei einem großen Datensatz ist ein Absturz von RStudio möglich.

str(d)	Struktur des Datensatzes (Spaltennamen, Format der Elemente pro Spalte, erste Elemente jeder Spalte)
summary(d)	Gibt einen zusammenfassenden Überblick für jede Spalte. Output unterscheidet sich je nach Typ der Spalte.
quantile(v, probs=x)	Quantile von Spalte **v** ausgeben. Welche Quantile ausgegeben werden sollen wird mit dem Vektor **x** festgelegt.
factor(v)	Spaltenformat zu Faktoren umwandeln. Damit werden Kategorien als solche gekennzeichnet.
levels(v)	Ausgabe der Kategorien bei Faktor-Format
length(unique(v))	Anzahl der Kategorien
table(v)	Überblick wie viele Elemente pro Kategorie vorkommen
plot(v)	Diagramm für die zugehörige Spalte **v**. Diagrammart hängt von den Format der Spalte ab.
qqnorm(v) oder **EnvStats::qqPlot(v)**	Q-Q-Plot der Normalverteilung; **ggPlot()** stammt aus dem Paket **EnvStats**

Maße

Lagemaße

Frage: Wo liegen die Werte?

Modus	**which.max(table(v))**
	x <- v[complete.cases(v)] **which.max(table(x))** Ohne Berücksichtigung von NAs
Median	**median(v)**
	median(v, na.rm = TRUE) Ohne Berücksichtigung von NAs
Arithmetisches Mittel	**mean(v)**
	mean(v, na.rm = TRUE) Ohne Berücksichtigung von NAs

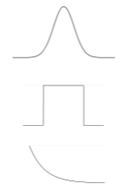
Streumaße

Frage: Wie unterschiedlich sind die Werte?

Varianz	**var(v)**
	var(v, na.rm = TRUE) Ohne Berücksichtigung von NAs
Standardabweichung	**sd(v)**
	sd(v, na.rm = TRUE) Ohne Berücksichtigung von NAs

Verteilungen

	Zufallsvariable	Dichtefunktion	Quantil
Normalverteilung	**rnorm**	**dnorm**	**qnorm**
Gleichverteilung	**runif**	**dunif**	**qunif**
Exponentialverteilung	**rexp**	**dexp**	**qexp**

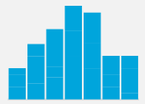


Histogramme

Ziel: Verteilung der Stichprobe grafisch darstellen

hist(v) Histogramm aus dem Vektor **v** erstellen

Sinnvolle Vorgänge:



- Anzahl der Balken verändern
  hist(v, breaks = zahl)
- Datensatz in Subgruppen aufteilen
  d_anteil_eins <- subset(d, bedingung)
  d_anteil_zwei <- subset(d, !bedingung)

! Achten Sie darauf, dass an der Grenze der Bedingung keine Datenzeilen verloren gehen.

Histogramme über alle Variablen:

Hmisc - Paket

hist(d) Alle Variablen aus dem Datensatz **d** gleichzeitig ansprechen und pro Spalte ein Histogramm erstellen.

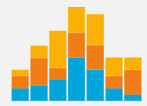
! Voraussetzung: alle Daten sind numerisch

Hmisc::hist.data.frame(d) Alle Variablen aus dem Datensatz **d** gleichzeitig ansprechen und pro Spalte ein Histogramm erstellen.

Segmentierungsanalyse mit Histogrammen

ggplot2 – Paket
viridis – Paket

Additives Histogramm



ggplot2::ggplot(d, aes(x=v1, fill = v2)) +
geom_histogram(binwidth = zahl_breite)

Additives Histogramm erstellen mit Datensatz **d** mit den Inhalten aus Spalte **v1**, wobei sich die Balken farblich nach Spalte **v2** aufeinander legen.

Separiertes Histogramm

ggplot2::ggplot(d, aes(x=v1, fill = v2)) +
geom_histogram(binwidth = zahl_breite,
position = "dodge")

Separiertes Histogramm erstellen mit Datensatz **d** mit den Inhalten aus Spalte **v1**, wobei sich die Balken farblich nach Spalte **v2** separieren.

LeverageData

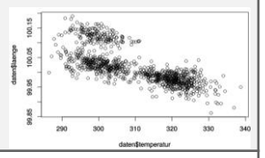
Erstellt von Annika Kies und Martin Kies von LeverageData GmbH • martin.kies@leveragedata.de • Version von: 2020-11

Streudiagramme und Matrixplot

Ziel: Variablen Abhängigkeiten bei numerischen Daten entdecken.

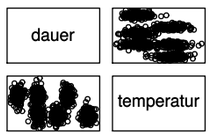
Streudiagramm (Scatter Plot / XY-Plot)

plot(x = v1, y = v2)
Stellt die Elemente der Spalte **v1** mit den Elementen der Spalte **v2** in Abhängigkeit dar.



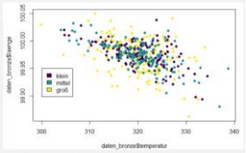
Matrixplot

spalten <- c(...)
plot(d[, spalten])
Stellt jede Kombination von Variablen getrennt dar. Hilfreich ist eine Vorauswahl von Spalten.



Segmentierung mit Streudiagramm

plot(x = v1, y = v2, pch = 19)
d1 <- subset(d, bedingung1)
d2 <- subset(d, bedingung2)
...



lines(d1$v1, d1$v2, type = "p", pch=19)
lines(d2$v1, d2$v2, type = "p", pch=19)
...
Stellt Streudiagramm mit den Elementen aus Spalte **v1** und Spalte **v2** in Abhängigkeit dar. Der Datensatz wird nach Bedingung aufgeteilt und jeder Datensatzanteil bekommt eine andere Farbe.

! Parameter **type = "p"** sorgt dafür, dass Punkte und keine Linien erzeugt werden.
Parameter **pch = 19** sorgt dafür, dass die Punkte ausgefüllt sind.

Korrelationsanalyse

Ziel: Objektive Kennzahl für die Abhängigkeit von Variablen ermitteln.

cor(v1, v2) ← Korrelationskoeffizient zwischen Spalte **v1** und Spalte **v2**.

! Setzen sie außerdem den Parameter **use** auf **use = "pairwise.complete.obs "**. Damit werden fehlende Werte in einer Spalte nicht berücksichtigt.

spalten <- c(...)
cor(d[,spalten]) ← Korrelationskoeffizienten-matrix von allen ausgewählten Spalten erstellen.

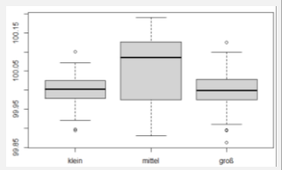
Grafische Aufbereitung:
corrplot-Paket oder ggplot2-Paket mit Ggally-Paket

Boxplots und Multi-Vari Chart

Ziel: Variablen Abhängigkeiten bei kategorialen und stetigen Daten entdecken.

Boxplot (Kastendiagramm)

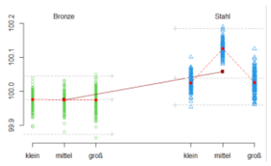
plot(x = v1, y = v2)
Stellt die Elemente der Spalte **v1** mit den Elementen der Spalte **v2** in Abhängigkeit dar. **x** muss dabei eine Faktorvariable erhalten.



Multi-Vari Chart

qualityTools - Paket

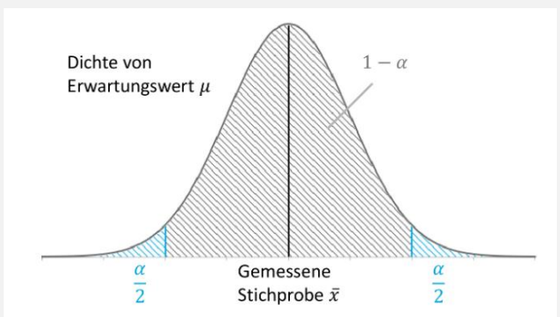
qualityTools::mvPlot(v1, v2, v3, FUN = ...)
Zwei kategoriale Daten aus den Spalten **v1** und **v2** in Abhängigkeit zu einer Spalte **v3** mit stetigen Variablen.



Konfidenzintervalle

Ziel: Bereich bestimmen, in dem der tatsächliche Wert mit hoher Wahrscheinlichkeit basierend auf der Stichprobe unter Modellannahmen liegt

Konfidenznivau	1 - α
Irrtumswahrscheinlichkeit	α Gibt an, wie viel Prozent aller Fälle der wahre Wert nicht im Konfidenzintervall liegt



Hypothesentest

Ziel: Beobachtungen von Eigenschaften der Grundgesamtheit statistisch verifizieren.

Vorgehensweise

1. Formulierung von Nullhypothese H_0 und Alternativhypothese H_1

H_0	H_1	Einordnung
$\theta \geq \theta_0$	$\theta < \theta_0$	Einseitiger Test (linksseitig)
$\theta \leq \theta_0$	$\theta > \theta_0$	Einseitiger Test (rechtsseitig)
$\theta = \theta_0$	$\theta \neq \theta_0$ ($\theta < \theta_0$ oder $\theta > \theta_0$)	Zweiseitiger Test

2. Auswahl eines geeigneten statistischen Tests
3. Festlegen der Irrtumswahrscheinlichkeit (meist 5%, 1%, oder 0.1%)
4. Durchführung des Tests: Vergleich von berechneter Testgröße mit dem kritischen Wert aus der Verteilung der Teststatistik
5. Treffen der Testentscheidung:
   → Falls berechneter Wert den kritischen Wert über/unterschreitet ist H_0 abzulehnen
   → Andernfalls ist H_0 nicht abzulehnen und H_1 ist plausibel

Fehler 1. und 2. Art

	H_0 ist richtig	H_0 ist falsch
H_0 wird nicht verworfen	Richtige Entscheidung Wahrscheinlichkeit: $1-\alpha$	Fehler 2. Art Wahrscheinlichkeit: β
H_0 wird verworfen	Fehler 1. Art Wahrscheinlichkeit: α	Richtige Entscheidung Wahrscheinlichkeit: $1-\beta$

ANOVA

Ziel: Mehrere Kategorien miteinander vergleichen im Hinblick der Mittelwerte.

a <- aov(y ~ k, data = d)
summary(a)
a – Ergebnis der ANOVA
y – Zu prüfende Variable
k – Kategorivariable
d – Datensatz

Interpretation:
Pr(>F) klein -> **y** ist über **k** hinweg im Mittel nicht gleich

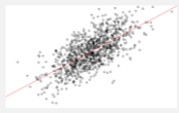
Lineare Regression

Ziel: Lineare Beziehung zwischen Variablen bestimmen.

Einfache lineare Regression:

Optimale Parameter a und b schätzen:
$$Y = a + bX + \varepsilon$$

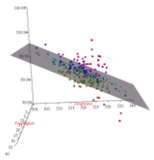
lm(y ~ x, data = d)
y / x – Die Variable y aus Datensatz **d** soll durch die Variable **x** aus Datensatz **d** erklärt werden.
Interpretation:
coefficients[1] bzw. **Intercept** → a
coefficients[2] → b



Multiple lineare Regression:

Optimale Parameter a,b_1,b_2... schätzen:
$$Y = a + b_1X_1 + b_2X_2 + \cdots + \varepsilon$$

lm(y ~ x1 + x2 + ..., data = d)
y / x1 / x2 – Die Variable y aus Datensatz **d** soll durch die Variablen **x1** und **x2** (...) aus Datensatz **d** erklärt werden.
Interpretation:
coefficients[1] bzw. **Intercept** → a
coefficients[2] → b1
coefficients[3] → b2
...



Statistische Verfikation

summary(lm(...))
Signifikanz der Regression anzeigen.

Bereich	Interpretation
Coefficients: ...	Beschreibung der Koeffizienten
	Estimate – Spalte gibt die geschätzten Parameter an **Pr(>|t|)** – Spalte berechnete p-Werte von Hypothesentest „Parameter = 0“ * - Sterne Geben Signifikanzniveau an
Residuals: ...	Einschätzung wie gut die Regression die Daten beschreibt
	Es gilt: umso < Residuen, desto besser beschreibt die Regressionsgerade die Datenpunkte
Multiple R-squared: ...	%-der Varianz der Daten, die durch die Einflussvariablen beschrieben werden kann = 1 → Daten perfekt beschrieben
F-statistic: ...	Hypothesentest: alle Einflussparameter zusammen 0

LeverageData

Erstellt von Annika Kies und Martin Kies von LeverageData GmbH • martin.kies@leveragedata.de • Version von: 2020-11

Hypothesentest

Kürzel

- MW – Mittelwert
- Wsh. – Wahrscheinlichkeit
- Standardabw. - Standardabweichung

Pakete: compositions, car, goftest, DescTools, EnvStats

Testziel: Anteilswerte/ Zusammenhänge

Testziel: Verteilung

Testziel: Mittelwert oder Median/mittlere Lage oder Varianz/Varianzvergleich

Testziel: Median/ mittlere Lage

Testziel: Varianzvergleich

Testziel: Varianz/ Varianzvergleich

Testziel: Varianzvergleich

Testziel: Varianz

Sollen kategoriale Variablen verglichen werden?
- Nein → Sollen verschiedene Stichproben verglichen werden?
  - Nein → Sollen zwei Merkmale miteinander verglichen werden?
    - Nein → Gibt es mehr als zwei Ausprägungen bei dem betrachteten Merkmal?
      - Nein → Stichprobe symmetrisch?
        - Nein → Vorzeichentest (Eine Stichprobe)
          - **DescTools::SignTest(d, mu = zahl)**
          - **d** – Stichprobe
          - **zahl** – Median auf den getestet wird
          - Interpretation: **p-value** klein → Median ist nicht **zahl**
        - Ja → Wilcoxon-Vorzeichen-Rang-Test (Eine Stichprobe)
          - **wilcox.test(d, mu = zahl)**
          - **d** – Stichprobe
          - **zahl** – mittlere Lage auf die getestet wird
          - Interpretation: **p-value** klein → mittlere Lage ist nicht **zahl**
      - Ja → Unabhängige Stichproben?
        - Nein → Differenz symmetrisch?
          - Nein → Vorzeichentest (Zwei Stichproben)
            - **DescTools::SignTest(d1, d2, mu = zahl)**
            - **d1/d2** – Stichprobe 1/2
            - **zahl** – Abstand zw. Median 1 & 2
            - Interpretation: **p-value** klein → Abstand Median **d1** & **d2** ist nicht **zahl**
          - Ja → Wilcoxon-Vorzeichen-Rang-Test (Zwei Stichproben)
            - **wilcox.test(d1, d2, mu = zahl, paired = TRUE)**
            - **d1/d2** – Stichprobe 1/2
            - **zahl** –mittlere Lage zw. Stichprobe 1 & 2
            - **paired = TRUE** – Stichproben sind abhängig voneinander
            - Interpretation: **p-value** klein → mittlere Lage zw. **d1** & **d2** ist nicht **zahl**
        - Ja → Mehr als zwei Stichproben?
          - Nein → Wilcoxon-Mann-Whitney-Test
            - **wilcox.test(d1, d2, mu = zahl, paired = FALSE)**
            - **d1/d2** – Stichprobe 1/2
            - **zahl** – mittlere Lage zw. 1 & 2
            - **paired = FALSE** – Stichproben sind unabhängig voneinander
            - Interpretation: **p-value** klein → mittlere Lage zw. **d1** & **d2** ist nicht **zahl**
          - Ja → Kruskal-Wallis-Test
            - **kruskal.test(d, g = gruppierung)**
            - **d** – Stichproben
            - **gruppierung** – Vektor mit den Gruppierungen
            - Interpretation: **p-value** klein → Abstand Median der Gruppierungen ist nicht gleich
    - Ja → Gibt es bei einem Merkmal mehr als zwei Ausprägungen?
      - Nein → Vierfeldertest
        - Spezialfall von χ^2-Unabhängigkeitstest
      - Ja → χ^2-Unabhängigkeitstest
        - **chisq.test(m, correct = FALSE)**
        - **m** – Matrix mit den absoluten Häufigkeiten auf die getestet wird
        - **correct = FALSE** – umstrittene Yates Korrektur ist ausgeschaltet
        - Interpretation: **p-value** klein → Spalten von **m** unterschiedliche Häufigkeiten
  - Ja → χ^2-Verteilungstest
    - **chisq.test(v_n, p = p0)**
    - **v_n** – Vektor **v** mit k Kategorien und absolute Vorkommnisse von n_k pro Kategorie k in einem Vektor
    - **p0** – Wahrscheinlichkeiten auf die getestet wird in einem Vektor
    - Interpretation: **p-value** klein → die Wahrscheinlichkeits-verteilung trifft nicht zu
    - Binomialtest
      - **binom.test(n1, n2, p = p0, alternative = a, conf.level = c)**
      - **n1** – Anzahl untersuchte Vorkommnisse
      - **n2** – Gesamtanzahl
      - **p0** – Wsh. auf die getestet wird
      - **a** – legt fest, ob ein- oder zweiseitig getestet wird (Std: beidseitig)
      - **c** – Konfidenzniveau, also 1 – Irrtumswsh. α
      - Interpretation: **p-value** klein → **p0** trifft nicht zu
- Ja → χ^2-Homogenitätstest
  - **chisq.test(m, correct = FALSE)**
  - **m** – Matrix mit den absoluten Häufigkeiten auf die getestet wird
  - **correct = FALSE** – umstrittene Yates Korrektur ist ausgeschaltet
  - Interpretation: **p-value** klein → Spalten von **m** unterschiedliche Häufigkeiten

Liegen 8 oder mehr Datenpunkte vor?
- Nein → Soll auf Normalverteilung getestet werden?
  - Nein → Test nicht sinnvoll
  - Ja → Liegen 3 bis 5000 Datenpunkte vor?
    - Nein → Test nicht sinnvoll
    - Ja → Shapiro-Wilk-Test
      - **shapiro.test(d)**
      - **d**– Stichprobe
      - Interpretation: **p-value** klein → **d** ist nicht normalverteilt
- Ja → Sind die Parameter der zu prüfenden Normalverteilung bekannt?
  - Nein → Anderson-Darling-Test
    - **goftest::ad.test(d, "pvert", mean=mean(d), sd=sd(d), estimated=TRUE)**
    - **d** – Stichprobe
    - ***pvert*** – Kürzel der Verteilung, die geprüft werden soll. Bsp.: **pnorm, punif, pexp…**
    - **mean(d)** – MW von d
    - **sd(d)** – Standardabw. von d
    - **estimated=TRUE** – Parameter der Stichprobe sind geschätzt
    - Interpretation: **p-value** klein → Verteilung ist nicht ***pvert***
  - Ja → Kolmogorov-Smirnov-Test
    - **ks.test(d, "pvert", mean=mean(d), sd=sd(d))**
    - **d** – Stichprobe, ***pvert*** – Kürzel der Verteilung, die geprüft werden soll. Bsp.: **pnorm, punif, pexp…, mean(d)** – MW von **d**, **sd(d)** – Standardabw. von **d**
    - Interpretation: **p-value** klein → Verteilung ist nicht ***pvert***
    - **ks.test(d1, d2)**
    - **d1/d2** – Stichprobe 1/2
    - Interpretation: **p-value** klein → Verteilungen von **d1/d2** sind nicht gleich

Normalverteilung oder Glockenform der Stichprobe(n)?
- Nein → Mehr als zwei Stichproben?
  - Nein → Levene-Test
    - **car::leveneTest(d1, d2, center="mean")**
    - **d1/d2** – Stichprobe 1/2
    - **center="mean"** – Standard Levene-Test
    - Interpretation: **Pr(>F)** klein → Varianz ist nicht gleich
  - Ja → Bartlett-Test
    - **bartlett.test(list(d1,d2,d3))**
    - **d1/d2/d3** – Stichprobe 1/2/3
    - Interpretation: **p-value** klein → Varianz ist nicht gleich
- Ja → Mehr als eine Stichprobe?
  - Ja → Mehr als zwei Stichproben?
    - Nein → F-Test
      - **var.test(d1, d2)**
      - **d1/d2** – Stichprobe 1/2
      - Interpretation: **p-value** klein → Varianz ist nicht gleich
    - Ja → Bartlett-Test
      - **bartlett.test(list(d1,d2,d3))**
      - **d1/d2/d3** – Stichprobe 1/2/3
      - Interpretation: **p-value** klein → Varianz ist nicht gleich
  - Nein → VarTest
    - **EnvStats::varTest(d, sigma.squared = zahl)**
    - **d** – Stichprobe
    - **zahl** – Varianz auf die getestet wird
    - Interpretation: **p-value** klein → Varianz ist nicht **zahl**

Ist ein Sichttest per Histogramm oder Q-Q-Plot ausreichend?
Achtung: Bei Stichprobenzahlen kleiner 10 und größer 100 sind Verteilungstests in der Praxis oft wenig aussagekräftig!
- Nein → Sollen zwei Stichproben miteinander verglichen werden?
  - Nein → Sichttest
    - Unabhängige Stichproben?
      - Nein → Abhängiger t-Test
        - **t.test(d1, d2, mu= zahl, paired=TRUE)**
        - **d1/d2** – Stichprobe 1/2
        - **zahl** – Abstand zw. MW 1 & 2
        - Interpretation: **p-value** klein → Abstand MW **d1** & **d2** ist nicht zahl
      - Ja → Sind die Standardabweichungen σ_1 und σ_2 unterschiedlich?
        - Nein → Zweistichproben-t-Test
          - **t.test(d1, d2, mu = zahl, var.equal=TRUE)**
          - **d1/d2** – Stichprobe 1/2
          - **zahl**– Abstand zw. MW 1 & 2
          - Interpretation: **p-value** klein → Abstand MW **d1** & **d2** ist nicht zahl
        - Ja → Welch-Test
          - **t.test(d1, d2, mu = zahl, var.equal=FALSE))**
          - **d1/d2** – Stichprobe 1/2
          - **zahl**– Abstand zw. MW 1 & 2
          - Interpretation: p-value klein → Abstand MW **d1** & **d2** ist nicht gleich
  - Ja → Eine oder zwei Stichproben?
    - Eine → Ist die Standardabweichung σ unbekannt?
      - Nein → Gauß-Test
        - **compositions::Gauss.test(d, mean = zahl1, sd = zahl2)**
        - **d** – Stichprobe
        - **zahl1** –MW auf den getestet wird
        - **zahl2** – bekannte Standardabw.
        - Interpretation: **p-value** klein → MW ist nicht **zahl1**
      - Ja → Einstichproben-t-Test
        - **t.test(d, mu = zahl)**
        - **d** – Stichprobe
        - **zahl** – MW auf den getestet wird
        - Interpretation: **p-value** klein → MW ist nicht **zahl**
    - Zwei → Unabhängige Stichproben?
      - Nein → Abhängiger t-Test
      - Ja → Sind die Standardabweichungen σ_1 und σ_2 unterschiedlich?
        - Nein → Zweistichproben-t-Test
        - Ja → Welch-Test

Erstellt von Annika Kies und Martin Kies von LeverageData GmbH • martin.kies@leveragedata.de • Version von: 2020-11

LeverageData