Einfache Varianzanalyse

Peter Büchel

HSLU I

Stoc: Block 09

Einfache Varianzanalyse

- Ungepaarter t-Test: Vergleich von 2 Mittelwerten
- Einfache Varianzanalyse: Vergleich von mehreren Mittelwerten
- Beispiel:
  - Man will 3 Diäten auf ihre Wirksamkeit testen
  - Man wählt je 10 Personen zufällig aus
  - Misst den Gewichtsverlust nach 2 Monaten
  - Vergleicht die durchschnittlichen Gewichtsverluste
  - Wie bei t-Test: Sind Unterschiede statistisch signifikant?
  - Wieder Hypothesentest für Entscheid

Beispiel: Reissfestigkeit von Papier

- Papierhersteller, der Einkaufs-Papiertragtaschen herstellt, interessiert sich für die Verbesserung der Reissfestigkeit seines Produkts
- Vermutung: Reissfestigkeit hängt von Hartholz-Konzentration im Papierbrei ab
- Üblicherweise: Konzentrationen liegen in Bereich von 5 % bis 20 %
- Die Produktionsingenieure beschlossen, die Reissfestigkeit bei vier Hartholzkonzentrationsstufen mit einem vollständig randomisierten Versuchsplan zu untersuchen: bei 5 %, 10 %, 15 % und 20 %
- Für jede Konzentrationsstufe werden sechs Versuchsproben in einer Pilotanlage erstellt
- Die resultierenden 24 Papierproben werden in zufälliger Reihenfolge im Labor auf ihre Reissfestigkeit getestet

- Die gemessenen Reissfestigkeiten (in psi) sind in Tabelle festgehalten

	Versuchsprobe					
Hartholz-Konzentration [%]	1	2	3	4	5	6
5	7	8	15	11	9	10
10	12	17	13	18	19	15
15	14	18	19	17	16	18
20	19	25	22	23	18	20

- Bei Durchführung von Messungen darauf achten, dass Messumfeld so homogen wie möglich gehalten wird (möglichst gleiche Versuchsbedingungen)

- Falls wichtige Grössen ändern können, müssen sie miterfasst werden

- Weil man nie sicher ist, ob das gelingt, werden Messungen in zufälliger Reihenfolge durchgeführt
- Laborproben werden zufällig aus den 24 Proben gewählt, ohne Rücksicht auf die Hartholzkonzentration oder Fertigstellung der Probe
- Frage nach den Einflüssen der unterschiedlichen Behandlungen kann man zunächst untersuchen, indem man jede Gruppe durch einen *Zwei-Stichproben-Test* (d.h. z. B. durch den Rangsummen-Test von Wilcoxon, den t-Test von Student oder den Vorzeichen-Test) mit jeder anderen vergleicht
- Resultate für einen bestimmten Test in einer symmetrischen Matrix von p-Werten zusammenfassen

- Beispiel: Reissfestigkeit
- Tabelle: P-Werte für den Zwei-Stichproben t-Test für die Reissfestigkeit von Papier

Hartholz-Konzentration [%]	5 %	10 %	15 %	20 %
5 %	–			
10 %	0.0010	–		
15 %	0.00076	0.38	–	
20 %	0.00	0.006	0.010	–

- Vergleichen wir z.B. die Werte für 10 % und 20 %, so erhalten wir einen p-Wert von 0.006

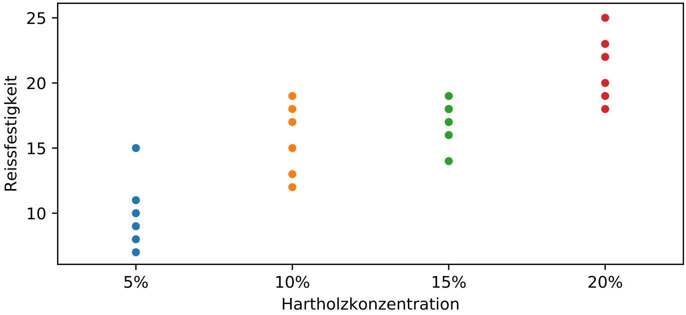
Python

- Code:

```
from pandas import DataFrame
import scipy.stats as st
import numpy as np
import seaborn as sns

rf = DataFrame({
"HC": np.repeat(["5%", "10%", "15%", "20%"], [6, 6, 6, 6]),
"Strength":[7, 8, 15, 11, 9, 10, 12, 17, 13, 18, 19, 15, 14, 18, 19, 17,
16, 18, 19, 25, 22, 23, 18, 20]
})

per5 = rf.loc[rf["HC"]=="5%", "Strength"]
per10 = rf.loc[rf["HC"]=="10%", "Strength"]
per15 = rf.loc[rf["HC"]=="15%", "Strength"]
per20 = rf.loc[rf["HC"]=="20%", "Strength"]

st.ttest_ind(per10,per20)
## Ttest_indResult(statistic=-3.4979930040209894, pvalue=0.00574574017074254)
```

- Unterschied zwischen 5 % und 10 % Hartholz-Konzentration mit einem p-Wert von 0.0010 signifikant
- Unterschied zwischen 10 % und 15 % Hartholz-Konzentration mit einem p-Wert von 0.35 *nicht* signifikant

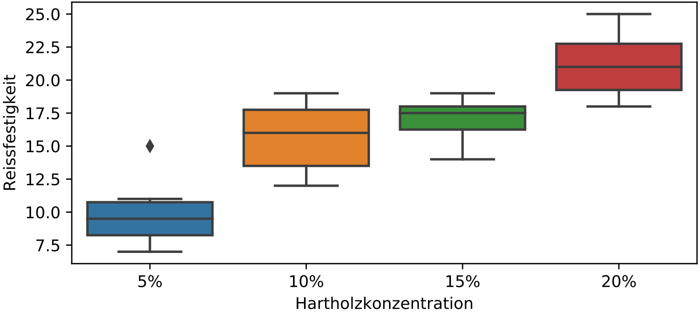
- Unterschiede können graphisch mit Hilfe von *Stripcharts*



- Code:

```
sns.stripplot(x="HC", y="Strength", data=rf)

plt.xlabel("Hartholzkonzentration")
plt.ylabel("Reissfestigkeit")

plt.show()
```

- Oder Boxplot:



- Code:

```
sns.boxplot(x="HC", y="Strength", data=rf)

plt.xlabel("Hartholzkonzentration")
plt.ylabel("Reissfestigkeit")

plt.show()
```

Vorsicht bei Paarvergleichen

- Vielzahl von Paar-Vergleichen problematisch von der Grundidee des statistischen Hypothesentests her

- Bsp: 7 Gruppen werden miteinander verglichen

- Anzahl Paarvergleich-Tests:

$$\frac{7 \cdot 6}{2} = 21$$

- Haben 7 Mittelwerte: $\mu_1, \ldots, \mu_7$

- Annahme: Es gibt *keinen* wahren Unterschied zwischen den Mittelwerten

- D. h.: Alle Nullhypothesen sollten beibehalten werden

$$\mu_1 = \mu_2, \quad \mu_1 = \mu_3, \quad \ldots \quad , \quad \mu_6 = \mu_7$$

- p-Werte von Paar-Vergleichen:

	1	2	3	4	5	6	7
1							
2	0.22						
3	0.42	0.53					
4	0.71	0.09	0.74				
5	0.31	0.55	0.21	0.89			
6	0.38	0.03	0.91	0.44	0.67		
7	0.23	0.43	0.10	0.15	0.27	0.39	

- Hier: Unterschied zwischen zwei Gruppen wird angezeigt:

$$\widehat{\mu}_2 \neq \widehat{\mu}_6$$

- $\widehat{\mu}_2, \widehat{\mu}_6$: Gemessene Mittelwerte

- Aber: Kein wahrer Unterschied vorhanden
- Problematik: Konstruktion Hypothesentest
- Zeigt Unterschied nur mit einer bestimmten Wahrscheinlichkeit an
- Werden sehr viele Hypothesentest gemacht, wird zu einer Wahrscheinlichkeit von 5 % Nullhypothese verworfen, obwohl keine Unterschied vorhanden ist

Theoretische Überlegung, wie Resultate dieser Tests aussehen können

- Aufgrund der Irrtums-W'keit von 5 % ist es anschaulich klar, dass ab und zu unter 21 Tests eine „Fehlentscheidung 1. Art", nämlich dass die Nullhypothese fälschlicherweise verworfen wird, auftritt

- Bei 21 Tests ist die erwartete Anzahl Fehlentscheide 1. Art:

$$21 \cdot 0.05 \sim 1$$

- D.h.: Im Mittel 1 Hypothesentest wird fälschlicherweise verworfen

- Die Nullhypothese, „alle Gruppen gehorchen dem gleichen Modell", wird also viel zu oft verworfen, wenn die Regel lautet:

- „Die Nullhypothese wird verworfen, wenn der extremste Unterschied auf dem Niveau $\alpha = 5\,\%$ signifikant ist"

- Wie kann man das vermeiden?
- Eine konsequente Antwort heisst: Wir dürfen nur *eine* Frage stellen, die wir mit einem Test beantworten
- Die sinnvolle Frage lautet: „Gibt es überhaupt Unterschiede zwischen den Gruppen?"
- Oder anders gesagt: „Unterscheidet sich wenigstens eine der Gruppen von einer andern?"
- Nullhypothese: „Alle Gruppen folgen dem gleichen Modell."

Gruppenmittel-Modell

- Beispiel zum Datensatz `Reissfestigkeit` von Papier, lässt sich durch ein lineares Modell (oder als Verallgemeinerung des Zwei-Stichproben-Modells) festhalten

- Wir wollen g Gruppen vergleichen, wobei in jeder Gruppe gerade m Beobachtungen gemacht werden

- Datensatz `Reissfestigkeit`:
  - 4 unterschiedliche Hartholzkonzentrationen verwendet, also ist $g = 4$
  - für jede Hartholzkonzentration $m = 6$ Messungen für Reissfestigkeit

- Ziel ist es, ein Modell zu entwickeln, dass die Reissfestigkeit in Abhängigkeit der Hartholzkonzentrationsstufen beschreibt

Allgemeines Modell

- Einfachstes Modell: Einzelne Beobachtungen innerhalb einer Gruppe streuen um einen gemeinsamen Wert:

$$Y_{ij} = \mu_i + \varepsilon_{ij} \qquad i = 1, 2, \ldots, g; \quad j = 1, 2, \ldots, m$$

  wobei Y_{ij} die j-te Beobachtung in der i-ten Gruppe ist

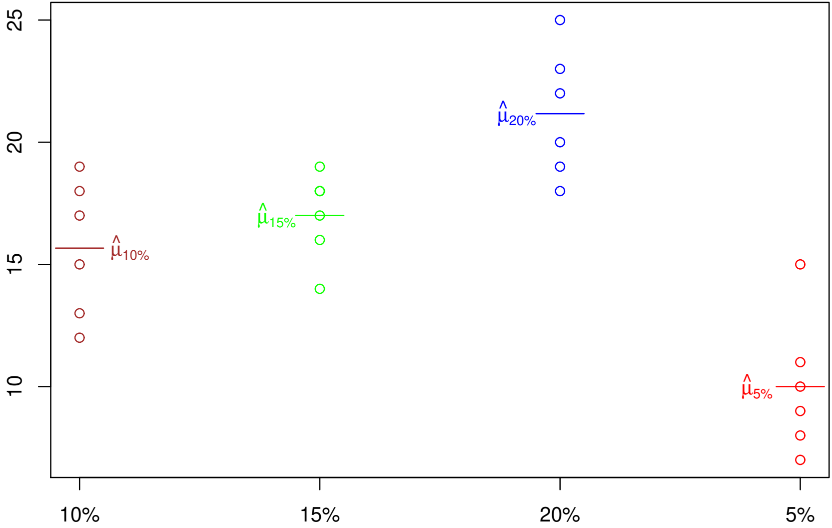
- Grösse μ_i: „Mittelwert" der i-ten Gruppe

- Annahme: Fehlerterme ε_{ij} unabhängig identisch normalverteilt sind

- Alle Gruppen dieselbe Standardabweichung des Fehlerterms in diesem Modell

Beispiel

- Datensatz `Reissfestigkeit`:



- Lineare Regression: Y_{ij} ist die Zielgrösse (die wir vorhersagen möchten), die Behandlungsart μ_i ist eine Faktorvariable (zu variierende Grösse)

- Äquivalente Modellformulierung:

$$Y_{ij} = \mu + \tau_i + \varepsilon_{ij} \qquad i = 1, 2, \ldots, g; \quad j = 1, 2, \ldots, m$$

  mit dem Fehler

$$\varepsilon_{ij} \sim \mathcal{N}(0, \sigma^2)$$

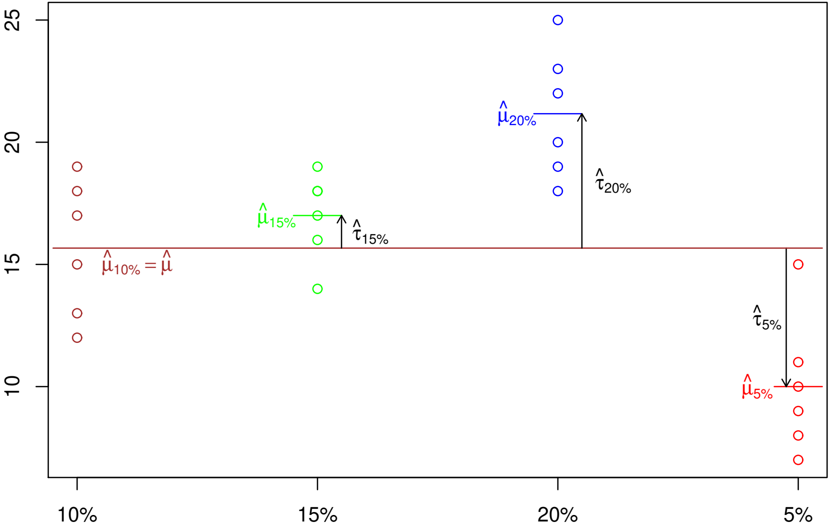
- Parameter μ haben also alle Beobachtungen gemeinsam („globaler Mittelwert")

- Parameter τ_i $(i = 1, \ldots, g)$ behandlungsspezifische Abweichungen von diesem globalen Mittelwert

- Beispiel: Spezifisch für jede Hartholz-Konzentration

Beispiel

- Datensatz `Reissfestigkeit`:



- Diese Parameter heissen auch *Behandlungseffekte* (eng. *treatment effects*)
- Parameter in diesem Modell nicht mehr eindeutig identifizierbar, da $g+1$ Parameter $\mu, \tau_1, \ldots, \tau_g$ für g unterschiedliche Gruppenmittelwerte vorhanden
- Benötigen Nebenbedingung, wobei es deren mehrere gibt
- Beispiel:

$$\mu = \mu_1$$

  und folglich

$$\tau_1 = 0, \quad \tau_2 = \mu_2 - \mu, \quad \tau_3 = \mu_3 - \mu$$

- Gruppe 1 bildet hier die Referenz, oder die sogenannte *Baseline*
- Nur $g-1$ der Behandlungseffekte τ_i frei variierbar

Parameterschätzung

- Wie schätzen wir nun die Parameter

$$\mu, \quad \tau_1, \quad \dots, \quad \tau_g$$

  so dass das Modell möglichst gut zu den Daten passt?

- Kriterium: Summe der quadrierten Residuen minimieren:

$$\sum_{i=1}^{g}\sum_{j=1}^{m}(Y_{ij} - \widehat{\mu} - \widehat{\tau}_i)^2$$

- Es kann gezeigt werden, dass

$$\widehat{\mu}_i = \frac{1}{m}\sum_{j=1}^{m} Y_{ij}$$

- Details (mühsam): siehe Skript

- Konkret mit

Beispiel: Reissfestigkeit Papier

- Koeffizienten des Gruppenmittel-Modells für `Reissfestigkeit`

- Code:

```
from pandas import DataFrame
import pandas as pd
import numpy as np
import scipy.stats as st

from statsmodels.formula.api import ols
from statsmodels.stats.anova import anova_lm

rf = DataFrame({
"HC": np.repeat(["5%","10%","15%","20%"], [6, 6, 6, 6]),
"Strength":[7, 8, 15, 11, 9, 10, 12, 17, 13, 18, 19, 15, 14, 18, 19, 17,
16, 18, 19, 25, 22, 23, 18, 20]
})

fit = ols("Strength~HC",data=rf).fit()
fit.summary()
```

- `ols`: ordinary least square

- Output:

```
##                            OLS Regression Results
## ==============================================================================
## Dep. Variable:               Strength   R-squared:                       0.746
## Model:                            OLS   Adj. R-squared:                  0.708
## Method:                 Least Squares   F-statistic:                     19.61
## Date:                Mon, 20 Apr 2020   Prob (F-statistic):           3.59e-06
## Time:                        11:19:09   Log-Likelihood:                -54.344
## No. Observations:                  24   AIC:                             116.7
## Df Residuals:                      20   BIC:                             121.4
## Df Model:                           3
## Covariance Type:            nonrobust
## ==============================================================================
##                  coef    std err          t      P>|t|      [0.025      0.975]
## ------------------------------------------------------------------------------
## Intercept     15.6667      1.041     15.042      0.000      13.494      17.839
## HC[T.15%]      1.3333      1.473      0.905      0.376      -1.739       4.406
## HC[T.20%]      5.5000      1.473      3.734      0.001       2.428       8.572
## HC[T.5%]      -5.6667      1.473     -3.847      0.001      -8.739      -2.594
## ==============================================================================
## Omnibus:                        0.929   Durbin-Watson:                   2.181
## Prob(Omnibus):                  0.628   Jarque-Bera (JB):                0.861
## Skew:                           0.248   Prob(JB):                        0.650
## Kurtosis:                       2.215   Cond. No.                         4.79
## ==============================================================================
##
## Warnings:
## [1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

- Kürzer:

```
fit.params
```

```
## Intercept    15.666667
## HC[T.15%]     1.333333
## HC[T.20%]     5.500000
## HC[T.5%]     -5.666667
## dtype: float64
```

- Beachte: Output der Parameter für HC10 % fehlt

- Der ist aber, da er zuerst auftritt, gleich 0 (Baseline)

- **Python**-Befehl **ols**: Globaler Mittelwert geschätzt durch $\widehat{\mu} = 15.66$
- Parametrisierung $\mu = \mu_1$
- Die geschätzten Gruppenmittelwerte lauten somit:

$$\begin{aligned}
\widehat{\mu}_{5\,\%} &= 15.7 - 5.7 = 10 \\
\widehat{\mu}_{10\,\%} &= 15.7 + 0 = 15.7 \\
\widehat{\mu}_{15\,\%} &= 15.7 + 1.3 = 17 \\
\widehat{\mu}_{20\,\%} &= 15.7 + 5.5 = 21.2
\end{aligned}$$

- 95 %-Vertrauensintervalle

- Code:

```
fit_pred = fit.get_prediction()
fit_pred.conf_int()
```

```
## [[ 7.8274691  12.1725309 ]
##  [ 7.8274691  12.1725309 ]
##  [ 7.8274691  12.1725309 ]
##  [ 7.8274691  12.1725309 ]
##  [ 7.8274691  12.1725309 ]
##  [ 7.8274691  12.1725309 ]
##  [13.49413576 17.83919757]
##  [13.49413576 17.83919757]
##  [13.49413576 17.83919757]
##  [13.49413576 17.83919757]
##  [13.49413576 17.83919757]
##  [13.49413576 17.83919757]
##  [14.8274691  19.1725309 ]
##  [14.8274691  19.1725309 ]
##  [14.8274691  19.1725309 ]
##  [14.8274691  19.1725309 ]
##  [14.8274691  19.1725309 ]
```

- Somit ist zum Beispiel das 95 %-Vertrauensintervall für $\mu_{5\,\%}$

$$[7.8, 12.2]$$

Beispiel: Fleischverpackung

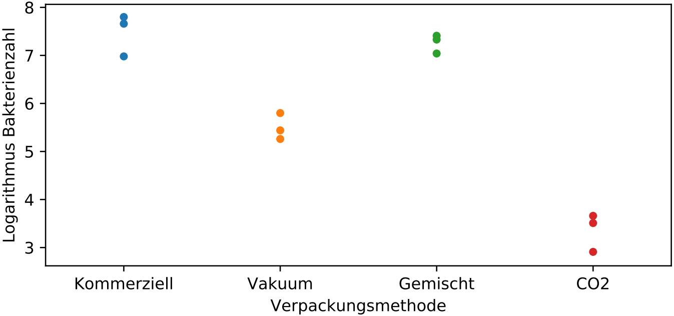
- Studie: Effekt der Verpackungsart auf das Bakterienwachstum von gelagertem Fleisch untersuchen
- Es wurden vier Verpackungsarten („Behandlungsarten”) untersucht:
  - Kommerzielle Plastikverpackung (mit Umgebungsluft)
  - Vakuumverpackung
  - 1 % CO, 40 % O_2, 59 % N
  - 100 % CO_2
- Versuchseinheiten besteht aus 12 Rindssteaks von rund 75 g
- Interessieren für die Wirksamkeit einer Verpackungsart, das Bakterienwachstum zu unterdrücken
- Gemessene Zielgrösse: (Logarithmus) Anzahl Bakterien pro Quadratzentimeter

Beispiel: Fleischverpackung

- Datensatz `Meat` graphisch darstellen
- Code:

```
meat = DataFrame({
"Treatment":
np.repeat(["Kommerziell","Vakuum","Gemischt","CO2"], [3, 3,
3, 3]),
"meat_id":[7.66, 6.98, 7.80, 5.26, 5.44, 5.80, 7.41, 7.33,
7.04, 3.51, 2.91, 3.66]
})

sns.stripplot(x="Treatment", y="meat_id", data=meat)
plt.xlabel("Verpackungsmethode")
plt.ylabel("Logarithmus Bakterienzahl")

plt.show()
```

- Plot:



Logarithmus Bakterienzahl
8
7
6
5
4
3
Kommerziell
Vakuum
Gemischt
CO2
Verpackungsmethode

- Koeffizienten des Gruppenmittel-Modells für den Datensatz `Meat`

- Code:

```
fit = ols("steak_id~Treatment",data=meat).fit()

fit.params
## Intercept                       3.36
## Treatment[T.Gemischt]           3.90
## Treatment[T.Kommerziell]        4.12
## Treatment[T.Vakuum]             2.14
## dtype: float64
```

- Somit lauten die geschätzten Gruppenmittelwerte

$$
\begin{aligned}
\widehat{\mu}_{\text{CO}_2} &= 3.36 - 0 = 3.36 \\
\widehat{\mu}_{\text{Kommerziell}} &= 3.36 + 4.12 = 7.48 \\
\widehat{\mu}_{\text{Gemischt}} &= 3.36 + 3.90 = 7.26 \\
\widehat{\mu}_{\text{Vakuum}} &= 3.36 + 2.14 = 5.50
\end{aligned}
$$

- 95 %-Vertrauensintervalle Python wie folgt:

- Code:

```
fit_pred = fit.get_prediction()

fit_pred.conf_int()
## [[7.02684427 7.93315573]
##  [7.02684427 7.93315573]
##  [7.02684427 7.93315573]
##  [5.04684427 5.95315573]
##  [5.04684427 5.95315573]
##  [5.04684427 5.95315573]
##  [6.80684427 7.71315573]
##  [6.80684427 7.71315573]
##  [6.80684427 7.71315573]
##  [2.90684427 3.81315573]
##  [2.90684427 3.81315573]
##  [2.90684427 3.81315573]]
```

- 95 %-Vertrauensintervall für $\mu_{\text{Kommerziell}}$

$$[7.03, 7.93]$$

Anova-Test

- *Anova*: *Analysis of Variance*
- Frage: Gibt es ob überhaupt Unterschiede zwischen den Gruppen?
- Nullhypothese:
$$H_0: \quad \mu_1 = \mu_2 = \ldots = \mu_g$$
- Alternativhypothese: Mindestens zwei Gruppen unterscheiden sich, also $\mu_i \neq \mu_j$ mit mit mindestens einem Paar $i \neq j$
- Bsp: Nullhypothese verwerfen, falls:
$$\mu_3 \neq \mu_5$$
- Gesucht Teststatistik, die extreme Werte annimmt, wenn sich die Gruppen in ihrer Lage unterscheiden

- Wenn sich Gruppenmittelwerte stark unterscheiden
  $\rightarrow$ Nullhypothese falsch

- Was „stark" heisst, hängt aber auch von der Streuung der Beobachtungen *innerhalb* der Gruppen ab

- Wie beim t-Test sogenannten F-Wert bilden

- Wenn F gross $\rightarrow$ Nullhypothese verwerfen

- Wenn F klein $\rightarrow$ Nullhypothese beibehalten

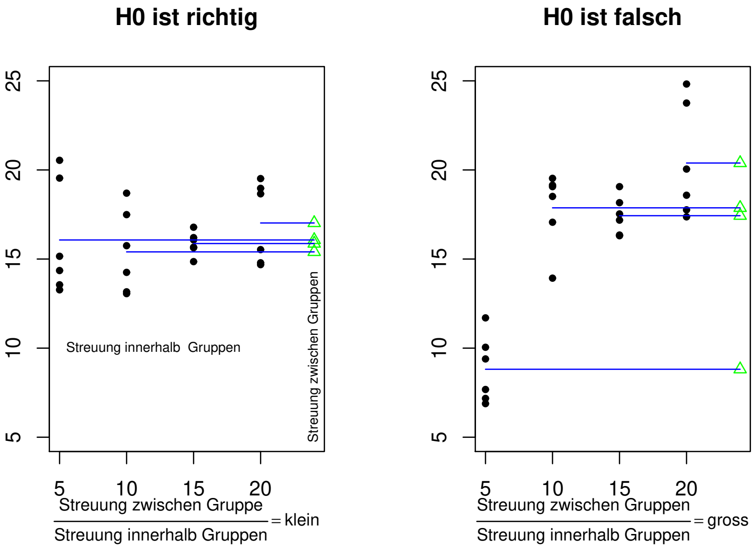
- Definition: F-Wert:

$$F = \frac{\text{Streuung der Gruppenmittelwerte}}{\text{Mittelwert der Streuungen der Gruppen}}$$

- Technische Details zur Berechnung dieses F-Wertes erheblich (machen das hier nicht)

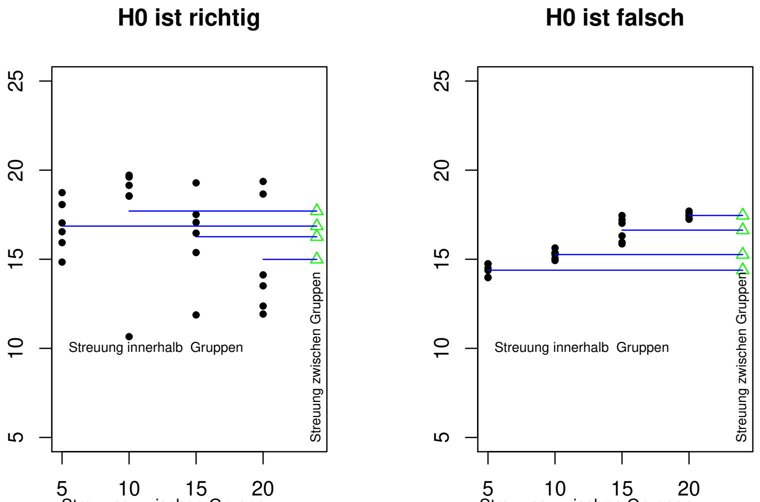
- Graphische Interpretation

- Abbildung:



- Abbildung: auf beiden Seiten Streuung *innerhalb* der Gruppen gleich
- Mittelwert der Streuungen gleich
- Nenner des F-Wertes ist auf beiden Seiten der Abbildung gleich
- Linke Seite Streuung der Mittelwerte kleiner als auf der rechten Seiten
- Zähler des F-Werte auf der linken Seite ist kleiner als der Zähler auf der rechten Seite
- Bei gleichbleibendem Nenner ist der F-Wert auf der linken Seite kleiner als der F-Wert auf der rechten Seiten
- Wenn der F-Wert klein genug $\quad\rightarrow\quad$ Nullhypothese *nicht* verwerfen

- Abbildung:



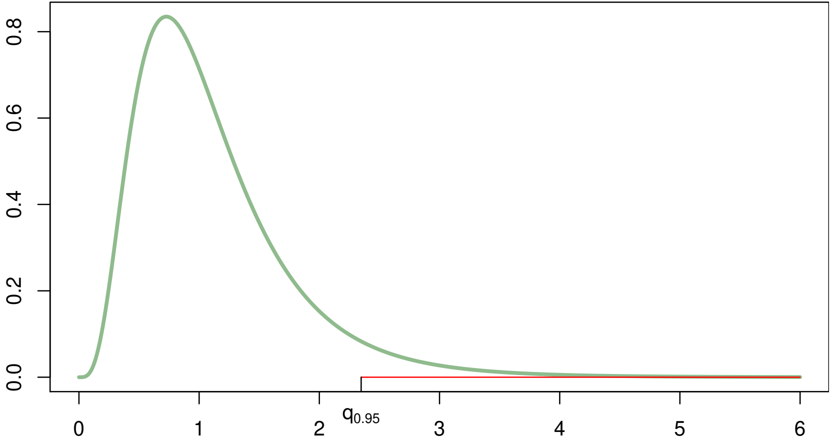
$$\frac{\text{Streuung zwischen Gruppe}}{\text{Streuung innerhalb Gruppen}} = \text{klein} \qquad \frac{\text{Streuung zwischen Gruppen}}{\text{Streuung innerhalb Gruppen}} = \text{gross}$$

- Abbildung: auf beiden Seiten Streuung der Mittelwerte gleich
- Zähler des F-Wertes auf beiden Seiten der Abbildung gleich
- Auf linker Seite Streuung in den Gruppen grösser als auf der rechten Seiten
- Nenner des F-Werte auf der linken Seite ist grösser als der Zähler auf der rechten Seite
- Bei gleichbleibendem Zähler ist der F-Wert auf der linken Seite kleiner als der F-Wert auf der rechten Seiten
- Wenn F-Wert klein genug $\quad\rightarrow\quad$ Nullhypothese *nicht* verwerfen

- Wie früher Teststatistik-Werte: F-Werte in Verteilung der Teststatistik unter der Null-Hypothese in p-Werte umrechnen
- p-Wert-Skala: Verwerfungsbereiche (unplausible Werte) einfach zu merken
- Bei p-Werten kleiner als das Niveau wird die Null-Hypothese verworfen, sonst beibehalten

F-Kurve

- Graph einer F-Kurve:



- Liegt F-Wert im roten Bereich (Verwerfungsbereich), dann wird H_0, verworfen

Bespiel: Papier Reissfestigkeit

- Varianzanalyse-Tabelle: **Python**

```
rf = DataFrame({
"HC": np.repeat(["5%","10%","15%","20%"], [6, 6, 6, 6]),
"Strength":[7, 8, 15, 11, 9, 10, 12, 17, 13, 18, 19, 15, 14, 18,
19, 17, 16, 18, 19, 25, 22, 23, 18, 20]
})

fit = ols("Strength~HC",data=rf).fit()

anova_lm(fit)
```

```
##              df      sum_sq     mean_sq          F    PR(>F)
## HC          3.0  382.791667  127.597222  19.605207  0.000004
## Residual   20.0  130.166667    6.508333        NaN       NaN
```

- 1. Spalte: `df` sind die Freiheitsgrade (degrees of freedom)
- 2. Spalte: `sum_sq` die Quadratsummen (Sum of Squares)
- 3. Spalte: `mean_sq` die mittlere Quadratsumme (Mean Squared)
- 4. Spalte; gefolgt von der Teststatistik `F` und zuletzt der *P*-Wert (`Pr(>F)`).
- Wert der Teststatistik und der entsprechende *P*-Wert werden auf der Zeile der Behandlung (entspricht hier der Zeile `HC`) aufgeführt
- *P*-Wert von $4 \cdot 10^{-6}$ besagt, dass ein Effekt von unterschiedlichen Hartholz-Konzentrationen signifikant auf dem 5 % Niveau nachgewiesen werden kann
- Die Gruppenmittelwerte unterscheiden sich also signifikant
- Schon aus Boxplots aus ersichtlich

Beispiel: Fleischverpackung

- Varianzanalyse-Tabelle für den Datensatz `Meat`

```
meat = DataFrame({
"Treatment": np.repeat(["Kommerziell","Vakuum","Gemischt","CO2"], [3, 3, 3,
3]),
"steak_id":[7.66, 6.98, 7.80, 5.26, 5.44, 5.80, 7.41, 7.33, 7.04, 3.51,
2.91, 3.66]
})

fit = ols("steak_id~Treatment",data=meat).fit()

anova_lm(fit)
```

```
##              df    sum_sq    mean_sq          F    PR(>F)
## Treatment   3.0   32.8728   10.95760  94.584376  0.000001
## Residual    8.0    0.9268    0.11585        NaN       NaN
```

- p-Wert von $1 \cdot 10^{-6}$ besagt, dass ein Effekt von unterschiedlichen Verpackungsmethoden signifikant auf dem 5 % Niveau nachgewiesen werden kann
- Die Gruppenmittelwerte unterscheiden sich also signifikant
- Diese Feststellung deckt sich mit der Beobachtung in Abbildung

Bemerkung

- Anova: Entscheidet nur, ob es einen Unterschied zwischen Mittelwerten gibt
- Macht *keine* Aussage, *welcher* abweicht
- Muss graphisch ermittelt werden