Ökonometrie — 2009

Oliver Kirchkamp*

12. Juli 2009 — 13:12

Dies ist eine Zusammenfassung der Folien aus der Vorlesung. Ohne den Besuch der Vorlesung hilft Ihnen dieses Dokument vermutlich nicht wirklich weiter. Es stellt kein „Vorlesungsskript“ dar und soll auch kein Lehrbuch ersetzen. Um Ihnen das Leben leicht zu machen, orientiert sich die Vorlesung (und auch diese Übersicht) am Vorgehen von Stock and Watson. Alle Formeln aus der Vorlesung finden Sie genauer (und mit deutlich weniger Fehlern) dort. Da ich die Angewohnheit habe, vor der Vorlesung Folien noch anzupassen (und hoffentlich zu verbessern), wird sich dieses Dokument im Laufe des Semesters noch verändern. Drucken Sie deshalb besser nur die aktuellen Seiten aus.

*Universität Jena, D-07743 Jena, email oliver@kirchkamp.de

Homepage: http://www.kirchkamp.de/oekonometrie/

Termine: Vorlesung: Mi 10:15-11:45, SR 223; Übung: Di 12:15-13:45, SR 226

Literatur:

* Stock and Watson; Introduction to Econometrics, Pearson, 2006
- Studenmund; Using Econometrics, Pearson, 2006
- Barreto and Howland; Introductory Econometrics, Cambridge, 2006

Software:

- R
  - frei
  - großer Funktionsumfang
  - Tipps zu R, Links zu Dokumentationen finden Sie auf der Homepage
  - Einen ersten Einstieg in R gewinnen Sie mit dem R-commander
  - Probieren Sie die Beispiele mit R aus der Vorlesung und Übung möglichst mit Ihrem Computer aus. Benutzen Sie die Online Hilfe um neue Kommandos zu verstehen.
- SAS, STATA, EViews, TSP, SPSS,...
  - teuer
  - stärker spezialisiert
  - keine konsequente Syntax

Inhaltsverzeichnis

1 Einleitung

Nennen Sie eine interessante ökonomische Theorie
Behauptung:

- Für jede ökonomische Theorie gibt es eine alternative Theorie die das Gegenteil vorhersagt.
- Viele ökonomische Theorie suggerieren Zusammenhänge — oft auch mit Implikationen für die Politik — aber praktisch nie wird der Zusammenhang quantifiziert.
- Um wieviel steigt die Leistung der Studenten, wenn Kurse kleiner werden?
- Wieviel mehr verdienen Sie, wenn Sie ein Jahr mehr studieren?
- Was ist die Preiselastizität für Zigaretten?
- Um wieviel steigt das BSP wenn die EZB den Zinssatz um 1% senkt?

1.1 Was macht Ökonomie

- Theorien entwickeln
- Theorien testen
- Theorien zur Vorhersage verwenden

1.2 Ökonometrie verwendet Daten um kausale Zusammenhänge zu messen

- Ideales Vorgehen: kontrolliertes Experiment (Kontrollgruppe/Treatmentgruppe)

  - Um wieviel steigt die Leistung der Studenten, wenn Kurse kleiner werden?
  - Wieviel mehr verdienen Sie, wenn Sie ein Jahr mehr studieren?
  - Was ist die Preiselastizität für Zigaretten?
  - Um wieviel steigt das BSP wenn die EZB den Zinssatz um 1% senkt?

- $\uparrow$ schwierig
- Fast immer haben wir Daten aus unkontrolliertem Prozess
  - Testscores von Studenten
  - Einkünfte von Studienabgängern
  - Zeitreihen zur Geldpolitik
- Probleme die mit Daten aus unkontrolliertem Prozess entstehen:
  - nicht beobachtete Faktoren
  - simultane Kausalitäten
  - Koinzidenz $\leftrightarrow$ Kausalität

1.3 Lernziele

- Anwendung ökonometrische Methoden
  - Quantifizierung kausale Effekte mit Beobachtungsdaten aus unkontrolliertem Prozess
  - Zeitreihen extrapolieren
- Bewertung der ökonometrischen Arbeit von anderen

1.4 Beispiel

- Wie reagiert der Lernerfolg, wenn Schulklassen um einen Schüler kleiner werden? Was, wenn die Klassen um 8 Schüler kleiner werden?
- Kann man diese Frage ohne Daten beantworten?

- z.B. test scores für 420 Schuldistrikte in Kalifornien in 1998/99
  - `str` = Student/teacher ratio (Anzahl der Schüler im Distrikt / Vollzeitäquivalente Anzahl Lehrer)
  - `testscr` = 5^{th}-grade test score (Stanford-9 achievement test)

Für die Beispiele verwenden wir die Statistische Software R.

Die einzelnen Komponenten von R sind auf Bibliotheken aufgeteilt. Da der Funktionsumfang aller Bibliotheken sehr groß ist, werden zu Beginn nur wenige Bibliotheken geladen. Weitere Bibliotheken können jedoch jederzeit mit dem Kommando `library` nachgeladen werden. Das Kommando `RSiteSearch` und die R Site Search Extension für Firefox helfen uns herauszufinden, welche Bibliothek gerade eine bestimmte Funktionalität bereitstellt. Hier verwenden wir die Bibliothek. `Ecdat` die verschiedene ökonometrische Datensätze bereitstellt sowie die Bibliothek `car` die uns zu einigen praktischen ökonometrischen Funktionen hilft.

```
library(Ecdat)
library(car)
```

Das Kommando `data` macht den Datensatz aus einer Bibliothek zugänglich.

```
data(Caschool)
```

Wir können jetzt auf Elemente dieses Datensatzes zugreifen. `summary` kann ein Überblicksstatistik anzeigen.

```
summary(Caschool$str)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  14.00   18.58   19.72   19.64   20.87   25.80
```

Allerdings ist es recht umständlich, immer wieder den Namen eines Datensatzes, hier `Caschool`, dazuzuschreiben. Wenn wir immer wieder den gleichen Datensatz verwenden, hilft das Kommando `attach(Caschool)` das erklärt, dass R fortan alle Variablen zunächst im Datensatz `Caschool` suchen soll.

```
attach(Caschool)
```

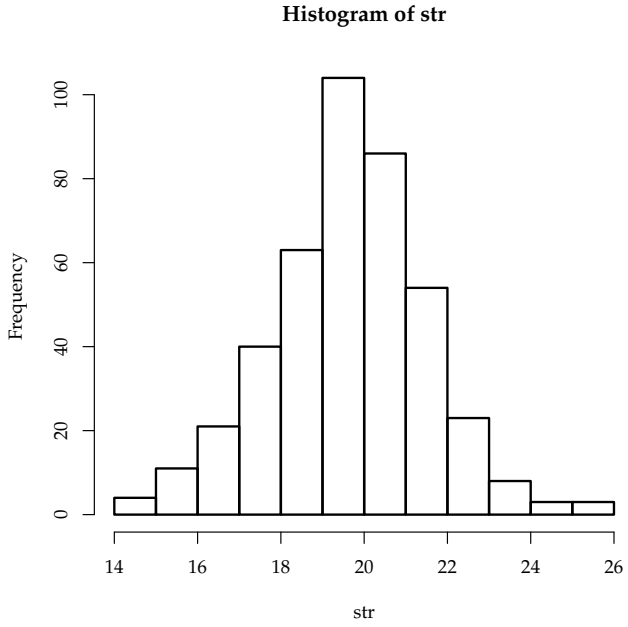
Das Kommando `summary` wird nun einfacher.

```
summary(str)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  14.00   18.58   19.72   19.64   20.87   25.80
```

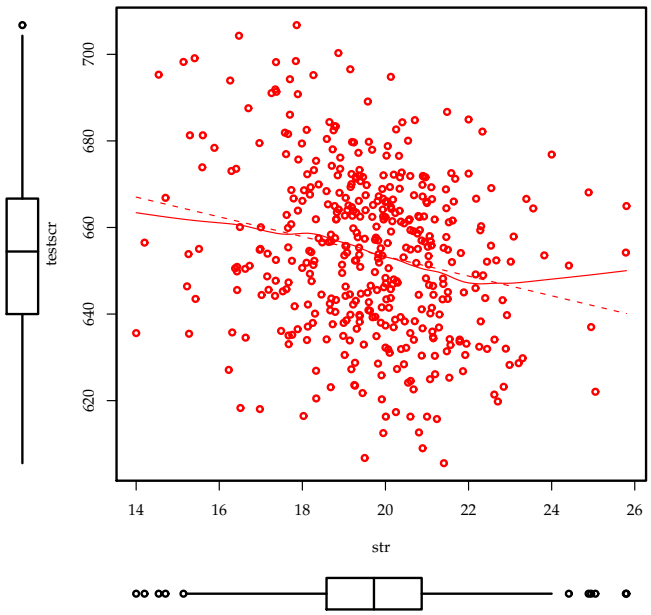
`hist` zeichnet ein Histogramm.

```
hist(str)
```



`scatterplot` zeichnet einen Scatterplot.

```
scatterplot(testscr ~ str)
```



Wir sehen, dass mit steigendem Student-Teacher Ratio `str` das Testergebnis `testscr` schlechter zu werden scheint.
Kann man zeigen, ob Distrikte mit kleinerem Student-Teacher ratio `str` größere test scores `testscr` erreichen?

- Vergleiche durchschnittliche test-scores in Distrikten mit kleinem `str` mit test-scores in Distrikten mit großem `str` (Schätzung)
- Teste die Nullhypothese, die mittleren test-scores seien gleich, gegen die alternative Hypothese, dass sie verschieden sind (Hypothesentest)
- Schätze ein Intervall für den Unterschied der mittleren test-scores (Konfidenzintervall)
- Ist der Unterschied groß genug für
  - eine Schulreform
  - für die Entscheidung der Eltern
  - für die Entscheidung der Schulbehörde

Im folgenden Beispiel wollen wir den Datensatz in zwei Teile zerlegen — Schulen mit einem student/teacher ratio größer und kleiner als 20. Das heißt, wir führen eine kategoriale Variable ein. In R heißt so etwas factor und das Kommando `factor` wandelt eine metrische Variable (`str`) in den factor um.

`t.test` führt einen student-t Test zum Vergleich von Mittelwerten durch. Die Notation `Caschool$testsrc ~ large` gibt vor der Tilde an, welche Variable überhaupt getestet werden soll (`testscr`). Hinter der Tilde steht der Faktor, der die zwei Gruppen beschreibt (`large`).

```
large <- str > 20
t.test(testscr ~ large)
```

```
        Welch Two Sample t-test

data:  testscr by large
t = 3.9231, df = 393.721, p-value = 0.0001031
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
  3.584445 10.785813
sample estimates:
mean in group FALSE  mean in group TRUE
           657.1846            649.9994
```

Dieser sehr einfache Test sagt uns also schon, dass es zwischen großen und kleinen Klassen einen signifikanten Unterschied der mittleren Testscores `testscr` gibt.
Wir können den Unterschied zwischen den beiden Gruppen schätzen, wir können eine Hypothese testen, und wir können ein Konfidenzintervall bestimmen.

1.5 Plan

- Schätzen, Hypothesentest, Konfidenzintervalle kennen Sie bereits.
- Wir werden diese Konzepte für Regressionen verallgemeinern.
- Vorher werden wir noch einen kurzen Blick auf die zugrundeliegende Theorie werden.

2 Statistische Theorie

- Population, Zufallsvariable, Verteilung
- Momente einer Verteilung (Mittelwert, Varianz, Standardabweichung, Kovarianz, Korrelation)
- Bedingte Verteilung, bedingte Mittelwerte
- Verteilung einer zufällig gezogenen Stichprobe

2.1 Population

- Die Menge aller möglichen Beobachtungseinheiten (z.B. alle denkbaren Schuldistrikte zu allen möglichen Zeitpunkten bei allen denkbaren Rahmenbedingungen)
- Wir gehen oft davon aus, dass die Population unendlich groß ist (oder wenigstens sehr groß)

2.2 Stichprobe

- Ein Teil der Population (z.B. Schuldistrikte in Kalifornien im Jahr 1998 (zu den Rahmenbedingungen in diesem Jahr))

2.3 Zufallsvariablen und Verteilungen

- Zufallsvariable (ZV) = numerische Zusammenfassung eines zufälligen Ereignisses
  - diskrete (kategoriale,Faktor-)/stetige Zufallsvariable
  - eindimensionale / mehrdimensionale Zufallsvariable
- Beschreibung von Zufallsvariablen durch Verteilungen:
  - Wahrscheinlichkeit von Ereignissen $P(x)$
    (bei diskreten ZV)

  - Kumulierte Wahrscheinlichkeitsfunktion $F(x)$
    (bei eindimensionalen ZV)
  - Dichtefunktion $f(x)$
    (bei stetigen ZV)

Eigenschaften von Zufallsvariablen

- Erwartungswert $E(X)$, μ_X, (theoretischer) Mittelwert von X
  langfristiger Mittelwert über sehr viele Realisierungen von X
- Varianz $E((X-\mu_X)^2) = \sigma_X^2$
  Maß für die langfristige mittlere quadratische Abweichung vom Mittelwert der Verteilung
- Standardabweichung $\sqrt{\text{Varianz}} = \sigma_X$

Gemeinsame Verteilung von Zufallsvariablen

- Zufallsvariablen X und Z haben eine gemeinsame Verteilung
- Kovarianz zwischen X und Z $\operatorname{cov}(X,Z) = E((X-\mu_X)(Z-\mu_Z)) = \sigma_{XZ}$
  - Kovarianz ist ein Maß für den linearen Zusammenhang zwischen X und Z
  - positive Kovarianz = positive Beziehung zwischen X und Z
  - Wenn X und Z unabhängig verteilt sind, dann ist $\operatorname{cov}(X,Z) = 0$ (aber nicht umgekehrt!!!)
  - Die Kovarianz einer ZV mit sich selbst ist die Varianz
    $\operatorname{cov}(X,X) = E((X-\mu_X)(X-\mu_X)) = E((X-\mu_X)^2) = \sigma_X^2$

Der Korrelationskoeffizient kann durch Kovarianzen ausgedrückt werden:

$$\operatorname{cor}(X,Z) = \frac{\operatorname{cov}(X,Z)}{\sqrt{\operatorname{var}(X)\operatorname{var}(Z)}} = \frac{\sigma_{XZ}}{\sigma_X \sigma_Z}$$

cor rechnet Korrelationskoeffizienten aus.

```
cor(str, testscr)
```

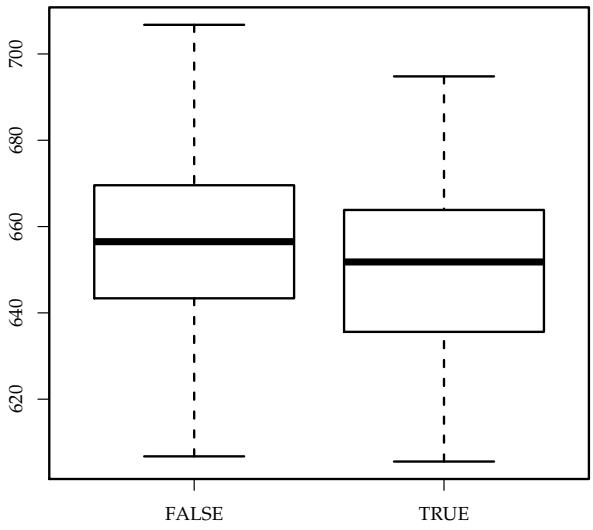
```
[1] -0.2263628
```

- $\text{cor}(X, Z) \in [-1, +1]$
- $\text{cor}(X, Z) = 1$ perfekter positiver linearer Zusammenhang
- $\text{cor}(X, Z) = -1$ perfekter negativer linearer Zusammenhang
- $\text{cor}(X, Z) = 0$ kein linearer Zusammenhang

Bedingte Verteilungen und bedingte Mittelwerte

- Bedingte Verteilungen
  - Die Verteilung von Y, gegeben den Wert einer anderen Zufallsvariablen X
  - z.B. die Verteilung der test-scores `testscr`, gegeben dass das student/teacher-ratio `str`< 20

```
boxplot(testscr ~ large)
```



  - z.B. Löhne von Männern und Frauen

```
data(Wages)
```

Der Datensatz `Wages` enthält z.B. die folgenden beiden Variablen:

`exp`	years of full-time work experience
`lwage`	logarithm of wage

Jetzt haben wir zwei Datensätze im Speicher. Um R klarzumachen, über welchen der beiden wir reden, gibt es verschiedene Möglichkeiten:

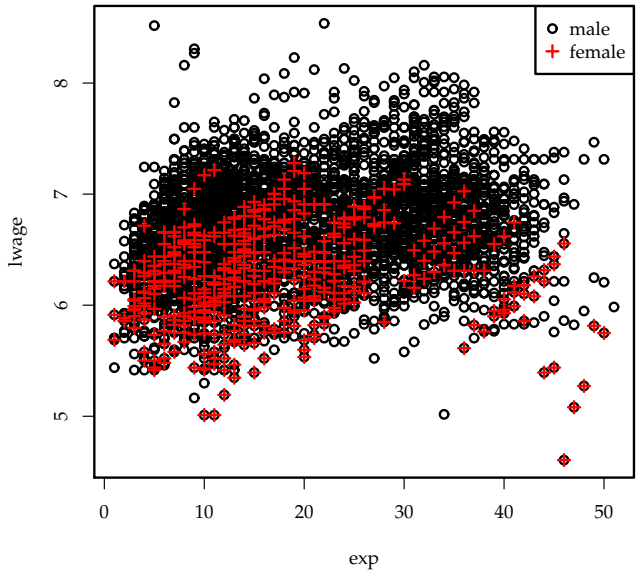
Oben haben wir bereits das Kommando `attach(Caschool)` kennengelernt. Es erklärt, dass R fortan alle Variablen zunächst im Datensatz `Caschool` suchen soll. Wir können nur mit `detach` diese Suchanweisung entfernen, und stattdessen mit `attach(Wages)` R sagen, dass wir fortan im Datensatz `Wages` suchen wollen.

Alternativ können wir `Caschool$large` etwa die Variable `large` im Datensatz `Caschool`, und mit `Wages$exp` die Variable `exp` im Datensatz `Wages` bezeichnen.

Dann gibt es das Kommando `with(Wages,...)`, welches bedeutet, dass alles was in der Klammer hinter `with` steht, sich auf den Datensatz `Wages` bezieht.

Neu im folgenden Beispiel ist `subset`. Damit können wir einen Teil des Datensatzes auswählen, hier z.B. alle Arbeiter bei denen das Kriterium `sex=="female"` erfüllt ist.

```
with(Wages, plot(lwage ~ exp))
with(subset(Wages, sex == "female"), points(lwage ~ exp,
    col = "red", pch = 3))
legend("topright", c("male", "female"), pch = c(1, 3),
    col = c("black", "red"))
```



2.3.1 Bedingter Erwartungswert und bedingte Varianz

- Bedingter Erwartungswert: $\mathrm{E}(X|Y=y)$ (← wichtige Notation)

- Bedingte Varianz: Varianz der bedingten Verteilung

Beispiele:

- $\mathrm{E}(\text{testscr}|\text{str}<20)$ = erwartete Mittelwert der test-scores aller Distrikte mit kleiner Klassengröße

```
with(subset(Caschool, str < 20), mean(testscr))
```

```
[1] 657.3513
```

```
with(subset(Caschool, str >= 20), mean(testscr))
```

```
[1] 649.9788
```

- Lohn weiblicher Arbeiter (X=Lohn, Y=Geschlecht)

```
with(subset(Wages, sex == "female"), mean(lwage))
```

```
[1] 6.255308
```

```
with(subset(Wages, sex == "male"), mean(lwage))
```

```
[1] 6.729774
```

- Heilungsrate aller Patienten die ein bestimmtes Medikament bekommen (X=Heilung, Y=Medikament)

Wenn $E(Y|X)$ =const dann ist $\text{cor}(X,Z) = 0$ (nicht umgekehrt!!!)

2.4 Stichproben einer Population

- Betrachte eine Stichprobe $Y_1 \dots Y_n$ einer Population Y
- Bevor die Stichprobe gezogen wird, sind $Y_1 \dots Y_n$ zufällig.
- Nachdem die Stichprobe gezogen wurde, sind die Werte von $Y_1 \dots Y_n$ beobachtet und Zahlen — nicht zufällig.
- Der Datensatz ist $Y_1 \dots Y_n$, wobei Y_i der Wert von Y für die i-te Beobachtung (die i-te Person, Distrikt i, etc.)
- Bei zufälliger Stichprobenziehung gilt:
  - Weil zwei Beobachtungen zufällig gezogen wurden, enthält der Wert von Y_i keine Information über Y_j.
  - Y_i und Y_j sind unabhängig verteilt
  - Da Y_i und Y_j auch von der gleichen Verteilung kommen, sind sie auch identisch verteilt
  - Wir sagen auch, dass Y_i und Y_j unabhängig und identisch (independent and identical = i.i.d.) verteilt sind.
  - Allgemeiner: Y_i sind i.i.d. für $i = 1, \dots, n$.

2.5 Schätzungen

In der Ökonometrie werden wir oft einen unbekannten Wert „schätzen“. Nehmen wir an, wir haben eine Stichprobe $Y_1 \dots Y_n$ einer Zufallsvariablen Y. Wir beginnen mit einem einfachen Problem: Wie können wir beispielsweise den Mittelwert von Y schätzen?
Idee:

- Wir nehmen einfach den Mittelwert $\bar{Y}$ der Stichprobe $Y_1 \dots Y_n$
- Wir nehmen einfach die erste Beobachtung Y_1
- Wir nehmen den Median der Stichprobe $Y_1 \dots Y_n$
- $\dots$

2.5.1 Die Verteilung von $\bar{Y}$

- Die Beobachtungen der Stichprobe sind zufällig gezogen.
- Also sind auch die Werte von $Y_1 \dots Y_n$ zufällig.
- Also sind auch Funktionen von $Y_1 \dots Y_n$ zufällig (z.B. der Mittelwert)

  Hätten wir eine andere Stichprobe gezogen, dann hätte die Funktion (z.B. der Mittelwert) auch einen anderen Wert.
- Die Verteilung von $\bar{Y}$ über verschiedene mögliche Stichproben nennen wir die Stichprobenverteilung von $\bar{Y}$.
- Mittelwert und Varianz von $\bar{Y}$ sind Mittelwert und Varianz der Stichprobenverteilung $E(\bar{Y})$ und $\operatorname{var}(\bar{Y})$.

2.5.2 Eigenschaften von Stichprobenverteilungen

Erwartungswert von $\bar{Y}$
$E(\bar{Y}) = \mu_Y$, d.h. $\bar{Y}$ ist ein unverzerrter Schätzer von μ_Y

Varianz von $\bar{Y}$
Wie hängt die Varianz von der Stichprobengröße n ab?

$$\operatorname{var}(\bar{Y}) = \frac{\sigma_Y^2}{n}$$

Frage: Konvergiert $\bar{Y}$ gegen μ_Y wenn n groß ist?

Gesetz der großen Zahl:
$\bar{Y}$ ist ein konsistenter Schätzer von μ_Y.
Formal: Wenn $Y_1, \ldots, Y_n$ i.i.d. und $\sigma_Y^2 < \infty$, dann ist $\bar{Y}$ ein konsistenter Schätzer von μ_Y, d.h.

$$\forall_{\epsilon>0} : \lim_{n\to\infty} \Pr(|\bar{Y} - \mu_Y| < \epsilon) = 1 \qquad \text{wir sagen auch} \qquad Y \xrightarrow{p} \mu_Y$$

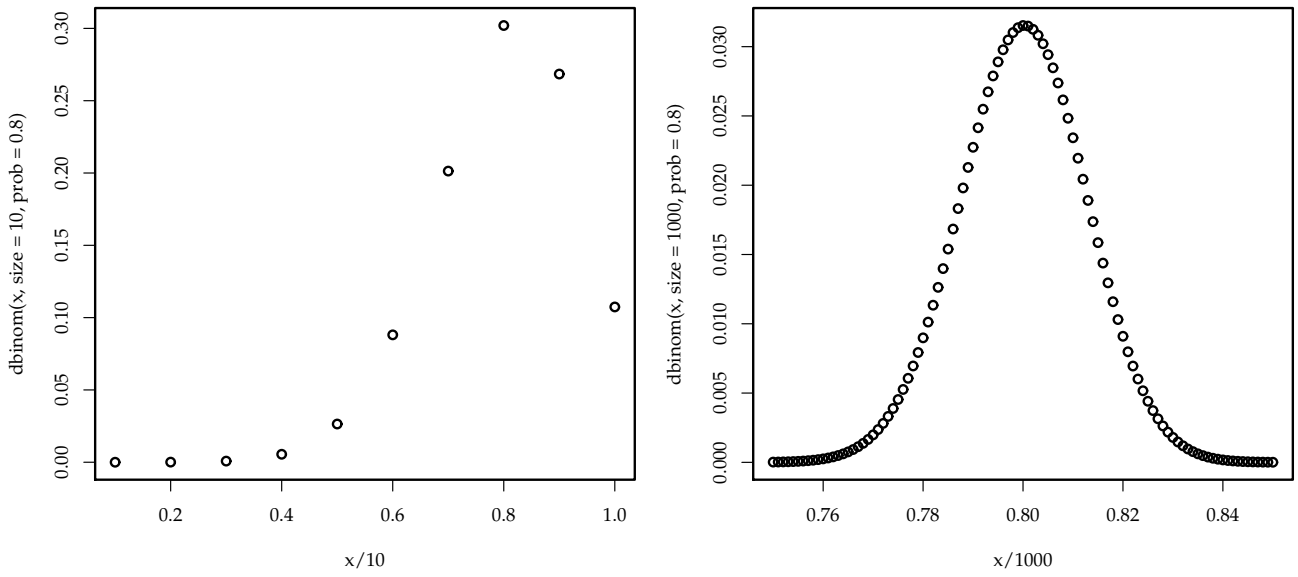
Zentraler Grenzwertsatz:
Wenn $Y_1, \ldots, Y_n$ i.i.d. und $0 < \sigma_Y^2 < \infty$ und n groß ist, dann approximiert die Verteilung von $\bar{Y}$ eine Normalverteilung

$$\bar{Y} \sim N\left(\mu_Y, \frac{\sigma_Y^2}{n}\right)$$

Natürlich kennt R auch Verteilungen. Im folgenden Beispiel zeichnen wir z.B. zwei Dichtefunktionen einer binomialverteilten Variablen mit `dbinom`.

```
x <- 1:10
plot(x/10, dbinom(x, size = 10,
    prob = 0.8))
```

```
x = 750:850
plot(x/1000, dbinom(x, size = 1000,
    prob = 0.8))
```



Die linke Verteilung, die auf einer kleinen Stichprobe basiert, sieht noch nicht so aus wie eine Normalverteilung. Bei der rechten Verteilung, die auf einer sehr viel größeren Stichprobe ($n = 1000$) basiert, ist die Nähe zur Normalverteilung sehr viel ausgeprägter.

2.5.3 Warum sollten wir $\bar{Y}$ verwenden, um μ_Y zu schätzen?

- $\bar{Y}$ ist unverzerrt: $E(\bar{Y}) = \mu_Y$
- $\bar{Y}$ ist konsistent: $\bar{Y} \xrightarrow{p} \mu_Y$
- $\bar{Y}$ ist der „kleinste Quadrate“ Schätzer für μ_Y
  $\bar{Y}$ ist die Lösung von $\min_x \sum_{i=1}^{n} (Y_i - x)^2$
- $\bar{Y}$ hat eine kleinere Varianz als alle anderen linearen unverzerrten Schätzer.
  Für jeden Schätzer $\hat{\mu}_Y = \frac{1}{n} \sum_{i=1}^{n} a_i Y_i$ mit $\{a_i\}$ so dass $\hat{\mu}_Y$ unverzerrt ist, gilt $\text{var}(\bar{Y}) \leq \text{var}(\hat{\mu}_Y)$

- Es gibt aber auch nicht lineare Schätzer...

2.5.4 Hypothesentests

Kann es sein, dass der mittlere `testscr` 652 ist?

```
t.test(Caschool$testscr, mu = 652)
```

```
        One Sample t-test

data:  Caschool$testscr
t = 2.3196, df = 419, p-value = 0.02084
alternative hypothesis: true mean is not equal to 652
95 percent confidence interval:
 652.3291 655.9840
sample estimates:
mean of x
 654.1565
```

- $H_0 : E(Y) = \mu_{Y,0}$ versus $H_1 : E(Y) \neq \mu_{Y,0}$ (zweiseitiger Test)
- $H_0 : E(Y) = \mu_{Y,0}$ versus $H_1 : E(Y) > \mu_{Y,0}$ (einseitiger Test)
- $H_0 : E(Y) = \mu_{Y,0}$ versus $H_1 : E(Y) < \mu_{Y,0}$ (einseitiger Test)

- Signifikanzniveau eines Tests = Vorspezifizierte Wahrscheinlichkeit die Nullhypothese fälschlich abzulehnen, obwohl sie wahr ist.
- p-Wert einer Statistik (z.B. für $\bar{Y}$) = Wahrscheinlichkeit eine Stichprobe $Y_1, \ldots, Y_N$ zu ziehen, die wenigstens so advers zu unserer Nullhypothese ist wie unsere Daten — gegeben, dass unsere Nullhypothese wahr ist.

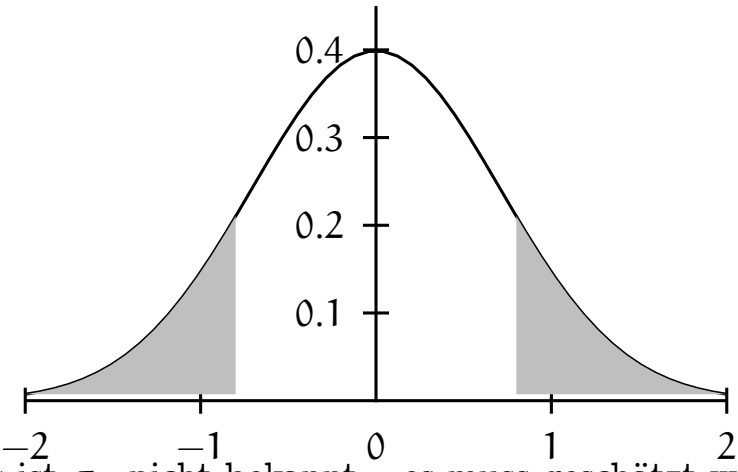
  z.B. bei $\bar{Y}$: p-Wert $= \Pr_{H_0}(|\bar{Y} - \mu_{Y,0}| > |\bar{Y}^{\text{Stichp.}} - \mu_{Y,0}|)$

  wobei $\bar{Y}^{\text{Stichp.}}$ der Wert für $\bar{Y}$ für unsere Daten ist.
- Um den p-Wert zu berechnen, muss man die Stichprobenverteilung von $\bar{Y}$ kennen. Das ist kompliziert, wenn n klein ist.

  Wenn n groß ist, kann man die Stichprobenverteilung von $\bar{Y}$ durch die Normalverteilung approximieren (Zentraler Grenzwertsatz)

$$
\begin{aligned}
p-\text{Wert} &= \Pr_{H_0}\left(|\bar{Y}-\mu_{Y,0}| > |\bar{Y}^{\text{Stichp.}}-\mu_{Y,0}|\right) && (1)\\
&= \Pr_{H_0}\left(\left|\frac{\bar{Y}-\mu_{Y,0}}{\sigma_Y/\sqrt{n}}\right| > \left|\frac{\bar{Y}^{\text{Stichp.}}-\mu_{Y,0}}{\sigma_Y/\sqrt{n}}\right|\right) && (2)\\
&= \Pr_{H_0}\left(\left|\frac{\bar{Y}-\mu_{Y,0}}{\sigma_{\bar{Y}}}\right| > \left|\frac{\bar{Y}^{\text{Stichp.}}-\mu_{Y,0}}{\sigma_{\bar{Y}}}\right|\right) && (3)
\end{aligned}
$$

Wenn n groß ist: $p-\text{Wert}$ = die Wahrscheinlichkeit, dass eine $N(0,1)$ verteilte Zufallsvariable außerhalb $\left|\frac{\bar{Y}^{\text{Stichp.}}-\mu_{Y,0}}{\sigma_{\bar{Y}}}\right|$ liegt.



- In der Praxis ist $\sigma_{\bar{Y}}$ nicht bekannt – es muss geschätzt werden.

2.5.5 Schätzen der Varianz von Y

$$s_Y^2 = \frac{1}{n-1}\sum_{i=1}^{n}\left(Y_i-\bar{Y}\right)^2 = \text{Stichprobenvarianz von } Y$$

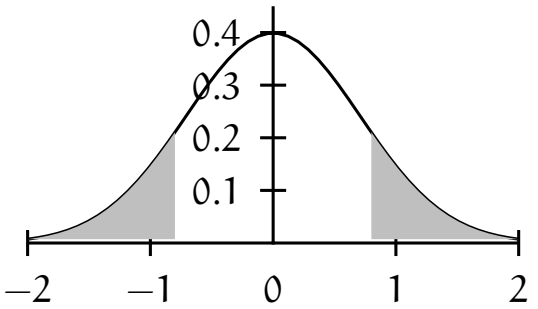
Wenn $Y_1,\ldots,Y_n$ i.i.d. sind und $E(Y^4)<\infty$, dann $s_Y^2 \xrightarrow{p} \sigma_Y^2$
Warum gilt das Gesetz der großen Zahl?

- s_Y^2 ist ein Stichprobenmittelwert
- Wir fordern hier $E(Y^4)<\infty$ weil der Mittelwert nicht von Y_i sondern von seinem Quadrat gebildet wird.

2.5.6 Berechnung des p-Wertes mit geschätztem σ_Y^2

2.5.7 Berechnung des p-Wertes mit geschätztem σ_Y^2

$$\begin{aligned} p-\text{Wert} &= \Pr_{H_0}\left(|\bar{Y}-\mu_{Y,0}| > |\bar{Y}^{\text{Stichp.}}-\mu_{Y,0}|\right) && (4)\\ &= \Pr_{H_0}\left(\left|\frac{\bar{Y}-\mu_{Y,0}}{\sigma_Y/\sqrt{n}}\right| > \left|\frac{\bar{Y}^{\text{Stichp.}}-\mu_{Y,0}}{\sigma_Y/\sqrt{n}}\right|\right) && (5)\\ &= \Pr_{H_0}\left(\left|\underbrace{\frac{\bar{Y}-\mu_{Y,0}}{s_Y/\sqrt{n}}}_{t}\right| > \left|\underbrace{\frac{\bar{Y}^{\text{Stichp.}}-\mu_{Y,0}}{s_Y/\sqrt{n}}}_{t^{\text{Stichp.}}}\right|\right) && (6)\end{aligned}$$



t ist die übliche t-Statistik und der p-Wert $= \Pr_{H_0}(|t| > |t^{\text{Stichp.}}|)$

2.5.8 Beziehung zwischen p-Wert und Signifikanzniveau

Das Signifikanzniveau wird vorgegeben. Z.B. wenn das vorgegebene Signifikanzniveau 5% ist,...

- ...wird die Nullhypothese verworfen wenn $|t| > 1.96$ ist,
- ...äquivalent wird die Nullhypothese verworfen $p < 0.05$ ist.
- Wir nennen den p-Wert auch marginales Signifikanzniveau.
- Oft ist es informativer den p-Wert anzugeben als zu sagen, ob der Test ablehnt oder nicht.

2.5.9 Was ist mit der t Tabelle und den Freiheitsgraden passiert?

- Wenn $Y_1, \ldots, Y_n$ i.i.d. ist und normalverteilt entsprechend $N(\mu_Y, \sigma_Y^2)$, dann folgt die t Statistik der Student-t Verteilung mit $n-1$ Freiheitsgraden.

- Die kritischen Werte der t-Verteilung finden Sie in allen alten Statistikbüchern. Das Rezept lautet:
  1. Berechne die t-Statistik
  2. Berechne die Freiheitsgrade $n-1$
  3. Schlage den 5% kritischen Wert nach.
  4. Wenn die t Statistik (absolut) größer als der kritische Wert ist, lehne die Nullhypothese ab.

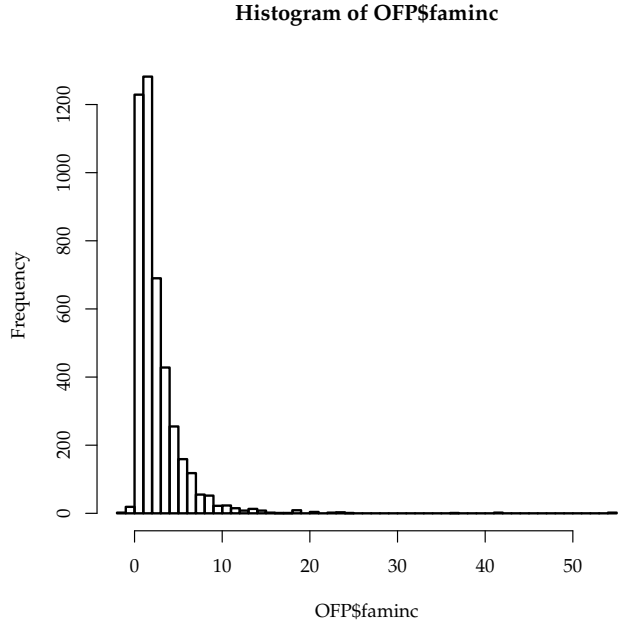
2.5.10 Kommentar

- Die Theorie der t Verteilung ist ein mathematisch schönes und interessantes Resultat.
- Wenn Y i.i.d. und normalverteilt ist, dann kennen wir die exakte Verteilung der t Statistik.

Aber

- Wenn die Y nicht genau normalverteilt sind, dann nützt uns das alles nichts.

```
data(OFP, package = "Ecdat")
hist(OFP$faminc, breaks = 40)
```



Das ist aber halb so schlimm:

- Egal wie Y verteilt ist, wenn n groß wird, konvergiert $\bar{Y}$ sowieso zur Normalverteilung.

2.5.11 Ein weiteres Problem:

Wenn wir zwei Gruppen vergleichen wollen, betrachten wir

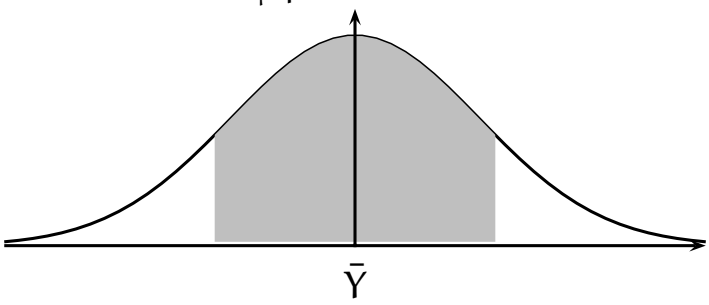
$$t = \frac{\bar{Y}_A - \bar{Y}_B}{\sqrt{\frac{s_A^2}{n_A} + \frac{s_B^2}{n_B}}}$$

Diese Statistik folgt nur dann der t-Verteilung, wenn

- Y normalverteilt und i.i.d. ist,
- und die Varianz $s_A^2 = s_B^2$ in beiden Gruppen gleich ist. Das ist oft eine heroische Annahme (Löhne von Männern vs. Löhne von Frauen)

2.6 Konfidenzintervalle

Ein 95% Konfidenzintervall für μ_Y ist das Intervall das in 95% aller wiederholter Stichproben den wahren Wert von μ_Y enthält.



- Beachte: Da die Stichprobe $Y_1, \ldots, Y_n$ zufällig sind, ist auch das Konfidenzintervall zufällig.

  Der Parameter μ_Y der Population ist nicht zufällig — wir kennen ihn aber nicht.

2.7 Zusammenfassung

Ausgehen von den Annahmen

- einfache Zufallsstichproben einer Population ($Y_1, \ldots, Y_n$ sind i.i.d.)
- $E\left(Y^4\right) < \infty$
- die Stichprobe ist groß (n ist groß)

kennen wir jetzt das Vorgehen

- zum Schätzen (Stichprobenverteilung von $\bar{Y}$)
- zum Testen von Hypothesen ($\bar{Y}$ ist t-verteilt, approximativ normalverteilt, daraus kann man den p-Wert berechnen)
- zum Berechnen von Konfidenzintervallen

Sind die obigen Annahmen plausibel?

3 Lineare Regression mit einem Regressor

- lege eine gerade Linie durch zweidimensionale Daten Y und X
- schätze einen kausalen Zusammenhang zwischen Y und X

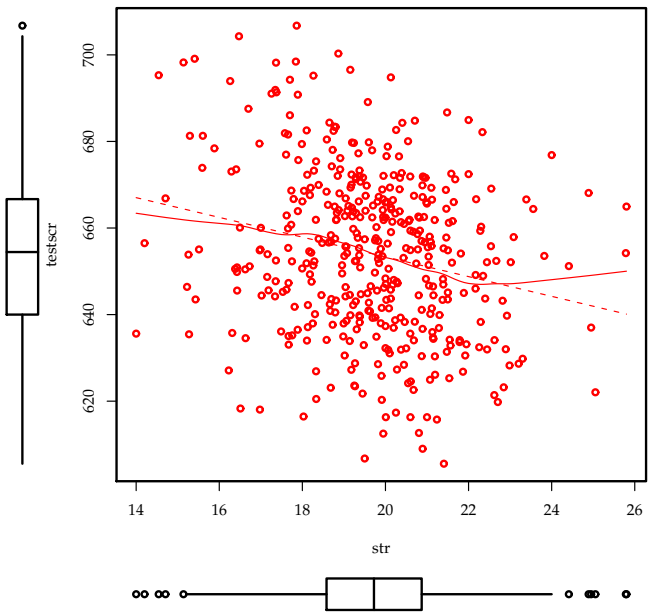
Linien haben Steigung und Achsenabschnitt
$\rightarrow$ Schätzung, Hypothesentest, Konfidenzintervalle

$$\texttt{testscr} = \beta_1 \texttt{str} + \beta_0$$

- β_0 und β_1 sind Parameter der Population
- wir kennen sie nicht — also müssen wir sie schätzen (wie μ)

In der folgenden Graphik wird solch eine Gerade durch die Punktwolke gelegt:

```
data(Caschool)
attach(Caschool)
scatterplot(testscr ~ str)
```



$$Y_i = \beta_1 X_i + \beta_0 + u_i \qquad i = 1, \ldots, n$$

- Y abhängige Variable
- X unabhängige Variable
- β_1 Steigung
- β_0 Achsenabschnitt
- u Fehlerterm
  (andere Faktoren die Y beeinflussen)

Wie könnte man β_0 und β_1 schätzen?
Erinnern wir uns: $\bar{Y}$ war der Kleinste-Quadrate-schätzer für μ_Y.
$\bar{Y}$ ist die Lösung von $\min_m \sum_{i=1}^n (Y_i - m)^2$
Probiere den gleichen Ansatz für β_0 und β_1:

$$\min_{b_0, b_1} \sum_{i=1}^{n} (Y_i - (b_1 X_i + b_0))^2$$

`lm` schätzt eine OLS Regression. Das Ergebnis wird in einer Variablen (hier `est1`) abgespeichert.

Wenn man das Ergebnis sehen will, muß es angezeigt werden, z.B. mit `summary(est1)`.

Das Ergebnis kann aber auch graphisch dargestellt werden, z.B. mit `abline(est1)`.

Natürlich muss man diese Werte nicht von Hand ausrechnen. R kann das für uns erledigen

```
lm(testscr ~ str, data = Caschool)
```

```
Call:
lm(formula = testscr ~ str, data = Caschool)

Coefficients:
(Intercept)          str
     698.93        -2.28
```

- Approximation von Y

$$\hat{Y}_i = \hat{\beta}_1 X_i + \hat{\beta}_0 \qquad i = 1, \ldots, n$$

- Residuen

$$\hat{u}_i = Y_i - \hat{Y}_i \qquad i = 1, \ldots, n$$

$$\texttt{testsrc} = -2.2798 \cdot \texttt{str} + 698.93$$

$$\frac{\Delta \texttt{testsrc}}{\Delta \texttt{str}} = -2.2798$$

3.1 Bestimmtheitsmaße

- R^2 relativer Anteil der Varianz von Y der durch X erklärt wird.

$$Y_i = \hat{Y}_i + \hat{u}_i = \text{OLS Approximation} + \text{OLS Residuen}$$

$$\text{var}(Y_i) = \text{var}(\hat{Y}_i) + \text{var}(\hat{u}_i)$$

$$R^2 = \frac{\sum(\hat{Y}_i - \bar{\hat{Y}})^2}{\sum(Y_i - \bar{Y})^2}$$

$$0 \leq R^2 \leq 1$$

- Bei Regressionen mit einem einzigen Regressor X ist R^2 der Korrelationskoeffizient zwischen X und Y.

Hier sind zwei Wege, das R^2 in unserem Beispiel zu finden:

```
summary(lm(testscr ~ str, data = Caschool))
```

```
Call:
lm(formula = testscr ~ str, data = Caschool)

Residuals:
     Min       1Q   Median       3Q      Max
-47.7267 -14.2507   0.4826  12.8222  48.5404

Coefficients:
            Estimate Std. Error t value   Pr(>|t|)
(Intercept) 698.9330     9.4675  73.825    < 2e-16 ***
str          -2.2798     0.4798  -4.751 0.00000278 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 18.58 on 418 degrees of freedom
Multiple R-squared: 0.05124,        Adjusted R-squared: 0.04897
F-statistic: 22.58 on 1 and 418 DF,  p-value: 0.000002783
```

```
cor(testscr, str)^2
```

```
[1] 0.0512401
```

Was bedeutet es, wenn R^2 im Beispiel nur 0.05 ist?

- Standardfehler der Residuen $SER = \sqrt{\frac{1}{n-2}\sum_{i=1}^{n}(\hat{u}_i - \bar{\hat{u}})^2} = \sqrt{\frac{1}{n-2}\sum_{i=1}^{n}\hat{u}_i^2}$

Die SER wird nicht nur in der summary angegeben, wir können sie auch so ausrechnen:

```
est <- lm(testscr ~ str, data = Caschool)
sqrt(with(est, sum(residuals^2)/df.residual))
```

```
[1] 18.58097
```

3.2 OLS Annahmen

$$Y_i = \beta_1 X_i + \beta_0 + u_i \qquad i = 1, \ldots, n$$

1. $E(u_i | X_i = x) = 0$
2. (X_i, Y_i) sind i.i.d.
3. Große Ausreißer in X und Y sind selten (die vierten Momente von X und Y existieren).

3.2.1 Exkurs — Existenz von Momenten

Es gibt einige Verteilungen, bei denen nicht alle Momente existieren.
Beispiel: Die Cauchy Verteilung:

$$f(x) = \frac{1}{\pi \cdot (1 + x^2)} \qquad F(x) = \frac{1}{\pi} \arctan(x) + \frac{1}{2}$$

$$\sigma_X^2 = \int_{-\infty}^{\infty} x^2 \frac{1}{\pi \cdot (1 + x^2)} \, dx \to \infty$$

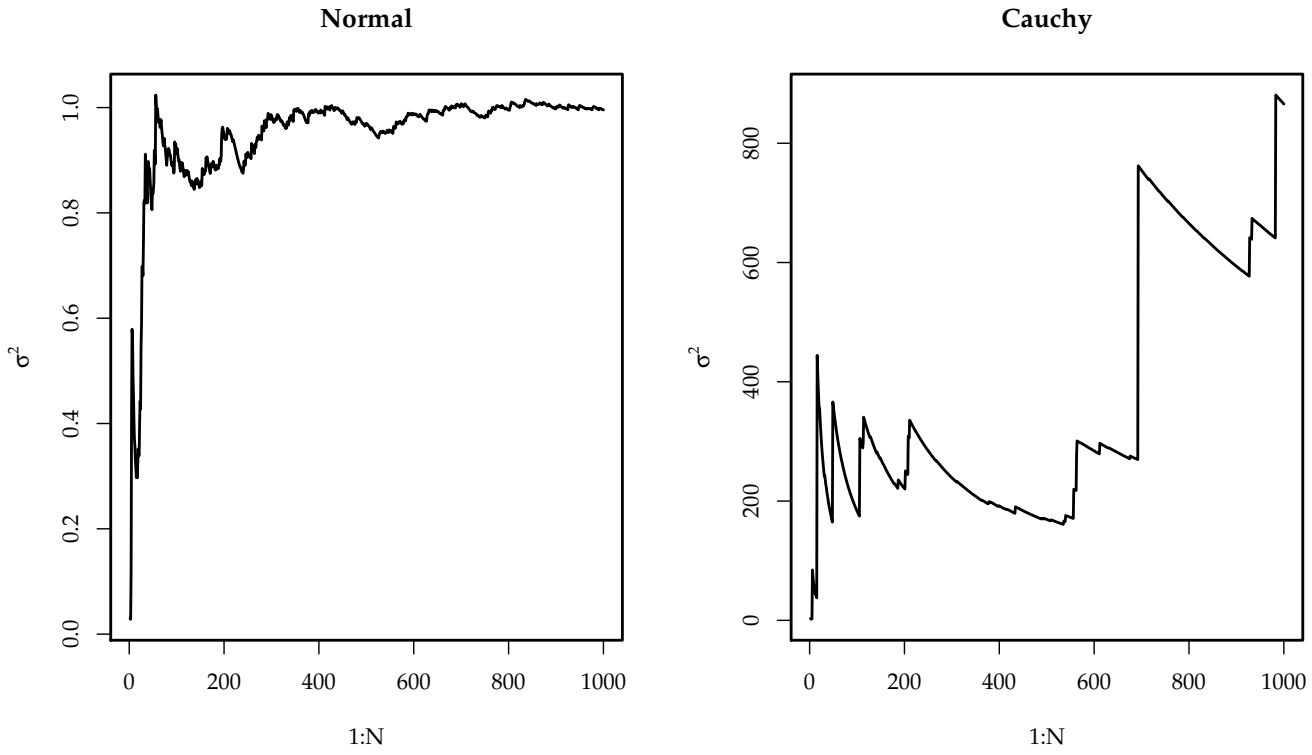
Während die Stichprobenvarianz einer normalverteilten Zufallsvariablen konvergiert, ist das bei der Cauchyverteilung nicht der Fall.

Im folgenden Beispiel werden zwei Plots in einem Bild dargestellt. Das macht das Kommando `par(mfrow=c(1,2))`. Mit dem Kommando `par(mfrow=c(1,1))` wird der Ausgangszustand wieder hergestellt.

`rnorm` erzeugt einen Vektor normalverteilter (pseudo-) Zufallsvariablen.

`rcauchy` erzeugt einen Vektor Cauchyverteilter (pseudo-) Zufallsvariablen.

```
set.seed(127)
N <- 1000
par(mfrow = c(1, 2))
z <- rnorm(N)
plot(1:N, sapply(1:N, function(x) {
    var(z[1:x])
}), ylab = expression(sigma^2), main = "Normal", t = "l")
z <- rcauchy(N)
plot(1:N, sapply(1:N, function(x) {
    var(z[1:x])
}), ylab = expression(sigma^2), main = "Cauchy", t = "l")
par(mfrow = c(1, 1))
```



3.3 Die Verteilung des OLS Schätzers

Beispiel: Wir approximieren die Verteilung des Schätzers in unserem Beispiel unter der Nullhypothese. Dazu schätzen wir wiederholt mit immer anderen Permutationen der unabhängigen Variablen.

`sample` zieht grundsätzlich ein Sample einer bestimmten Größe aus einem Vektor. Wenn keine Größe angegeben wird (wie hier), erhält man eine zufällige Permutation des gesamten Vektors.

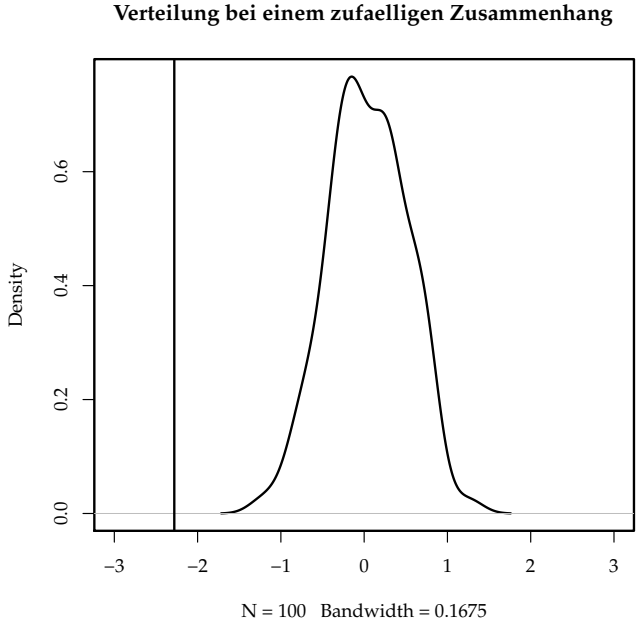
`coef` extrahiert die Koeffizienten aus einer Regression.

`replicate` führt einen Ausdruck mehrmals aus.

`density` schätzt eine Dichtefunktion.

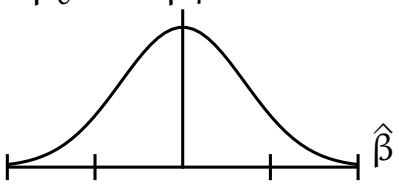
```
set.seed(123)
est <- lm(testscr ~ str)
strdist <- replicate(100, coef(lm(testscr ~ sample(str)))[2])
plot(density(strdist), xlim = c(-3, 3), main = "Verteilung bei einem zufaelligen
abline(v = coef(est)["str"])
coef(est)["str"]/sd(strdist)
```

```
     str
-4.87813
```



- $\hat{\beta}_0$ und $\hat{\beta}_1$ werden mit Hilfe des Samples berechnet. Ein anderes Sample ergibt auch andere Werte für $\hat{\beta}_0$ und $\hat{\beta}_1$.

Genauso wie $\bar{Y}$ gibt es auch für $\hat{\beta}_0$ und $\hat{\beta}_1$ eine Verteilung.



- Ist $E(\hat{\beta}_1) = \beta_1$? (OLS ist unverzerrt)
- Ist $\text{var}(\hat{\beta}_1)$ klein?
- Wie testen wir Hypothesen? (z.B. $\beta_1 = 0$)
- Wie bestimmen wir ein Konfidenzintervall für β_0 und β_1?

Mittelwert und Varianz von $\hat{\beta}_1$

Wir interessieren uns für $\beta_1 - \hat{\beta}_1$. Wir wissen

$$\begin{aligned} Y_i &= \beta_0 + \beta_1 X_i + u_i \\ \bar{Y} &= \beta_0 + \beta_1 \bar{X} + \bar{u} \\ \text{also } Y_i - \bar{Y} &= \beta_1 (X_i - \bar{X}) + (u_i - \bar{u}) \end{aligned}$$

$$\begin{aligned} \hat{\beta}_1 &= \frac{\sum_{i=1}^n (X_i - \bar{X})(Y_i - \bar{Y})}{\sum_{i=1}^n (X_i - \bar{X})^2} \\ &= \frac{\sum_{i=1}^n (X_i - \bar{X}) \left(\beta_1 (X_i - \bar{X}) + (u_i - \bar{u}) \right)}{\sum_{i=1}^n (X_i - \bar{X})^2} \\ &= \beta_1 \frac{\sum_{i=1}^n (X_i - \bar{X})(X_i - \bar{X})}{\sum_{i=1}^n (X_i - \bar{X})^2} + \frac{\sum_{i=1}^n (X_i - \bar{X})(u_i - \bar{u})}{\sum_{i=1}^n (X_i - \bar{X})^2} \\ \hat{\beta}_1 - \beta_1 &= \frac{\sum_{i=1}^n (X_i - \bar{X})(u_i - \bar{u})}{\sum_{i=1}^n (X_i - \bar{X})^2} \end{aligned}$$

Nun ist

$$\begin{aligned}\sum_{i=1}^{n}(X_i-\bar{X})(u_i-\bar{u}) &= \sum_{i=1}^{n}(X_i-\bar{X})u_i-\left(\sum_{i=1}^{n}(X_i-\bar{X})\right)\bar{u}\\ &= \sum_{i=1}^{n}(X_i-\bar{X})u_i-\left(\left(\sum_{i=1}^{n}X_i\right)-n\cdot\bar{X}\right)\bar{u}\\ &= \sum_{i=1}^{n}(X_i-\bar{X})u_i\end{aligned}$$

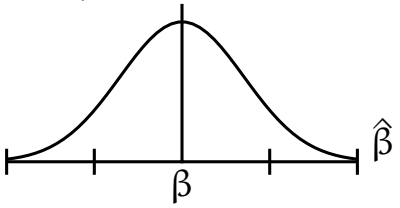
Also

$$\hat{\beta}_1-\beta_1=\frac{\sum_{i=1}^{n}(X_i-\bar{X})(u_i-\bar{u})}{\sum_{i=1}^{n}(X_i-\bar{X})^2}=\frac{\sum_{i=1}^{n}(X_i-\bar{X})u_i}{\sum_{i=1}^{n}(X_i-\bar{X})^2}$$

Jetzt können wir $E(\hat{\beta}_1)-\beta_1$ berechnen:

$$\begin{aligned}E\left(\hat{\beta}_1\right)-\beta_1 &= E\left(\frac{\sum_{i=1}^{n}(X_i-\bar{X})u_i}{\sum_{i=1}^{n}(X_i-\bar{X})^2}\right)\\ &= E\left(E\left(\frac{\sum_{i=1}^{n}(X_i-\bar{X})u_i}{\sum_{i=1}^{n}(X_i-\bar{X})^2}\,\middle|\,X_1,\ldots,X_n\right)\right)\\ &\qquad \text{da nach Annahme 1: } E(u_i|X_i=x)=0\\ E(\hat{\beta}_1)-\beta_1 &= 0\end{aligned}$$

$\hat{\beta}_1$ ist ein **unverzerrter Schätzer** von β_1



Jetzt zur Varianz:

$$\hat{\beta}_1-\beta_1=\frac{\sum_{i=1}^{n}(X_i-\bar{X})u_i}{\sum_{i=1}^{n}(X_i-\bar{X})^2}$$

$$\begin{aligned}
\text{nenne } (X_i - \bar{X})u_i &= \nu_i \\
\text{Ausserdem gilt } s_X^2 &= \frac{\sum_{i=1}^n (X_i - \bar{X})^2}{n-1} \\
\hat{\beta}_1 - \beta_1 &= \frac{\sum_{i=1}^n \nu_i}{(n-1)s_X^2} \\
&= \frac{\frac{1}{n}\sum_{i=1}^n \nu_i}{\frac{n-1}{n}s_X^2}
\end{aligned}$$

für große n gilt $s_x^2 \approx \sigma_x^2$ und $\frac{n-1}{n} \approx 1$, also

$$\hat{\beta}_1 - \beta_1 \approx \frac{\frac{1}{n}\sum_{i=1}^n \nu_i}{\sigma_X^2}$$

$$\text{nun ist } \operatorname{var}(\hat{\beta}_1) = \operatorname{var}(\hat{\beta}_1 - \beta_1) \approx \operatorname{var}\left(\frac{\frac{1}{n}\sum_{i=1}^n \nu_i}{\sigma_X^2}\right) = \frac{\operatorname{var}\left(\frac{1}{n}\sum_{i=1}^n \nu_i\right)}{(\sigma_X^2)^2}$$

$$= \frac{\operatorname{var}(\nu_i)/n}{(\sigma_X^2)^2} = \frac{1}{n}\frac{\operatorname{var}((X_i - \mu_X)\cdot u_i)}{\sigma_X^4}$$

Zusammenfassung

Wenn die drei OLS Annahmen gelten,...

1. $E(u_i|X_i = x) = 0$
2. (X_i, Y_i) sind i.i.d.
3. Große Ausreißer in X und Y sind selten (die vierten Momente von X und Y existieren).

...dann gilt auch...

- $E(\hat{\beta}_1) = \beta_1$ ($\hat{\beta}$ ist unverzerrt)
- $\operatorname{var}(\hat{\beta}_1) = \frac{1}{n}\frac{\operatorname{var}((X_i - \mu_X)u_i)}{\sigma_X^4}$

3.4 Verteilung von $\hat{\beta}_1$

- Mittelwert: $E(\hat{\beta}_1) = \beta_1$

  $\hat{\beta}_1 - \beta_1 = \frac{\sum_{i=1}^{n} v_i}{(n-1)s_X^2}$

  Wenn n groß ist, dann ist $\frac{1}{n}\sum_{i=1}^{n} v_i$ etwa normalverteilt (zentraler Grenzwertsatz).

- $\text{var}(\hat{\beta}_1) = \frac{1}{n}\frac{\text{var}(v_i)}{\sigma_X^4}$

$\hat{\beta}_1 \sim N\left(\beta_1, \frac{\sigma_v^2}{n\sigma_X^4}\right)$ wobei $v_i = (X_i - \mu_X)u_i$

Je größer die Varianz von X, um so kleiner die Varianz von $\hat{\beta}_1$

- mathematisch: $\hat{\beta}_1 \sim N\left(\beta_1, \frac{\sigma_v^2}{n\sigma_X^4}\right)$ wobei $v_i = (X_i - \mu_X)u_i$

- intuitiv:

  Wir bestimmen die Regressionsgerade in zwei Fällen: Zunächst nehmen wir das ganze Sample.

```
est1 <- lm(testscr ~ str)
summary(est1)
```

```
Call:
lm(formula = testscr ~ str)

Residuals:
     Min       1Q   Median       3Q      Max
-47.7267 -14.2507   0.4826  12.8222  48.5404

Coefficients:
            Estimate Std. Error t value   Pr(>|t|)
(Intercept) 698.9330     9.4675  73.825    < 2e-16 ***
str          -2.2798     0.4798  -4.751 0.00000278 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 18.58 on 418 degrees of freedom
```

```
Multiple R-squared: 0.05124,        Adjusted R-squared: 0.04897
F-statistic: 22.58 on 1 and 418 DF,  p-value: 0.000002783
```

Dann nehmen wir nur die Beoachtungen, bei denen str wenig vom Mittelwert abweicht:

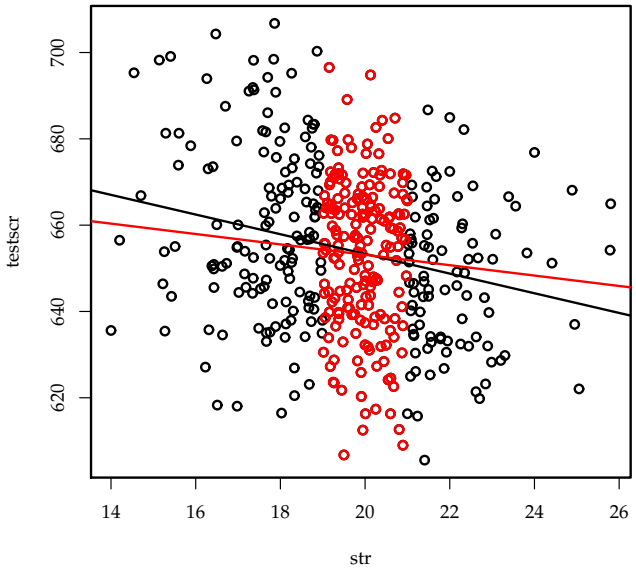
```
lowVar <- str > 19 & str < 21
est2 <- lm(testscr ~ str, subset = lowVar)
summary(est2)
```

```
Call:
lm(formula = testscr ~ str, subset = lowVar)

Residuals:
   Min     1Q Median     3Q    Max
-46.98 -13.39   2.82  12.74  42.40

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  677.205     44.538  15.205   <2e-16 ***
str           -1.204      2.233  -0.539     0.59
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 17.68 on 187 degrees of freedom
Multiple R-squared: 0.001552,        Adjusted R-squared: -0.003787
F-statistic: 0.2906 on 1 and 187 DF,  p-value: 0.5904
```

Wir sehen, dass im zweiten Fall Standardfehler und p-Werte größer werden.

Die folgende Graphik verdeutlicht das Problem:

```
plot(testscr ~ str)
points(testscr ~ str, col = "red", subset = lowVar)
abline(est1)
abline(est2, col = "red")
```



$$\hat{\beta}_1 \xrightarrow{p} \beta_1 \qquad (\hat{\beta}_1 \text{ ist konsistent})$$

Bei großen n gilt

$$\frac{\hat{\beta}_1 - E(\hat{\beta}_1)}{\sqrt{\text{var}(\hat{\beta}_1)}} \sim N(0,1)$$

(zentraler Grenzwertsatz, genau so wie bei $\bar{Y}$)

3.5 Verteilung von $\hat{\beta}_0$

Auch $\hat{\beta}_0$ ist, bei großem n, normalverteilt mit $\hat{\beta}_0 \sim N\left(\beta_0, \sigma^2_{\hat{\beta}_0}\right)$ wobei

$$\sigma^2_{\hat{\beta}_0} = \frac{1}{n}\frac{\text{var}(H_i u_i)}{\left(E\left(H_i^2\right)\right)^2} \qquad \text{und } H_i = 1 - \left(\frac{\mu_X}{E\left(X_i^2\right)}\right) \cdot X_i$$

↑ Verteilung für β_1 und β_0

→ Hypothesentests und Konfidenzintervalle für β_1 und β_0

3.6 Hypothesentests für $\hat{\beta}_1$

- zweiseitiger Test: $H_0 : E(\beta_1) = \beta_{1,0}$ versus $H_1 : E(\beta_1) \neq \beta_{1,0}$
- einseitiger Test: $H_0 : E(\beta_1) = \beta_{1,0}$ versus $H_1 : E(\beta_1) > \beta_{1,0}$
  $H_0 : E(\beta_1) = \beta_{1,0}$ versus $H_1 : E(\beta_1) < \beta_{1,0}$

wobei $\beta_{1,0}$ der hypothetische Wert der Nullhypothese ist
Ansatz: Konstruiere t Statistik und bestimme den p-Wert (oder vergleiche die t Statistik mit dem $N(0,1)$ kritischen Wert.

- Allgemein:

$$t = \frac{\text{Schätzer - hypothetischer Wert}}{\text{Standardfehler des Schätzers}}$$

  wobei der Standardfehler des Schätzers aus der geschätzten Varianz des Schätzers hergeleitet wird.

- um den Mittelwert von $\bar{Y}$ zu testen:

$$t = \frac{\bar{Y} - \mu_{Y,0}}{s_Y/N}$$

- um den Regressionskoeffizienten β_1 zu testen:

$$t = \frac{\hat{\beta}_1 - \beta_{1,0}}{\sigma_{\hat{\beta}_1}}$$

Erinnern wir uns an die theoretische Varianz von $\hat{\beta}_1$:

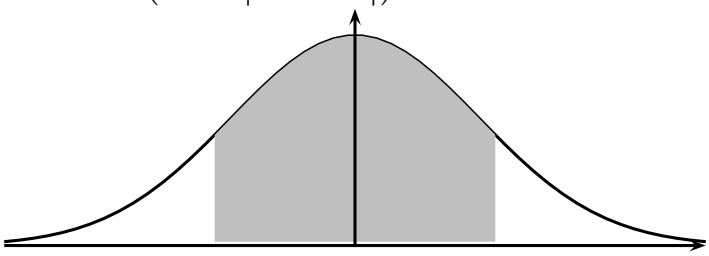
$$\sigma^2_{\hat{\beta}_1} = \frac{1}{n}\frac{\operatorname{var}((X_i - \mu_X)u_i)}{\sigma_X^4} = \frac{\sigma_v^2}{n \cdot \sigma_X^4}$$

analog die empirische Varianz:

$$\begin{aligned}\hat{\sigma}^2_{\hat{\beta}_1} &= \frac{1}{n}\frac{\text{Schätzer für } \sigma_v^2}{(\text{Schätzer für } \sigma_x^2)^2} \\ &\qquad \left(\text{mit } \hat{v} = \left(X_i - \bar{X}\right) \cdot \hat{u}_i\right) \\ &= \frac{1}{n}\frac{\frac{1}{n-2}\sum_{i=1}^n \hat{v}^2}{\left(\frac{1}{n}\sum_{i=1}^n \left(X_i - \bar{X}\right)^2\right)^2}\end{aligned}$$

Zusammenfassung: Um die Hypothese $H_0 : E(\beta_1) = \beta_{1,0}$ versus $H_1 : E(\beta_1) \neq \beta_{1,0}$ zu testen:

- bestimme die t Statistik:
$$t = \frac{\hat{\beta}_1 - \beta_{1,0}}{\hat{\sigma}_{\hat{\beta}_1}}$$
- Ablehnung auf 5% falls $t > 1.96$
- Der p-Wert ist $p = \Pr\left(|t| > \left|t^{\text{Stichp.}}\right|\right)$



- Wir benötigen die Annahme dass n groß ist ($n = 50$ ist „groß“)

`confint` berechnet Konfidenzintervalle für ein geschätztes Modell.

`pnorm` und `pt` berechnen die Verteilungsfunktion der Normalverteilung und der t-Verteilung. `qnorm` und `qt` berechnen die Quantile zu einem gegebenen Verteilungswert.

```
est1 <- lm(testscr ~ str)
confint(est1)
```

```
                2.5 %     97.5 %
(Intercept) 680.32313 717.542779
str          -3.22298  -1.336637
```

Natürlich können wir auch die Konfidenzintervalle von Hand ausrechnen:

`vcov` berechnet die Varianz-Kovarianz Matrix für $\hat{\beta}$ unter der Annahme homoskedastischer Residuen. `diag` nimmt die Diagonale einer Matrix. Bei der Varianz-Kovarianz Matrix sind das die Varianzen der Koeffizienten. `sqrt` berechnet Quadtratwurzeln.

```
coef(est1) + qnorm(0.025) * sqrt(diag(vcov(est1)))
```

```
(Intercept)         str
 680.377010   -3.220249
```

```
coef(est1) - qnorm(0.025) * sqrt(diag(vcov(est1)))
```

```
(Intercept)         str
 717.488895   -1.339367
```

Auch den p-Wert hatten wir oben in der summary schon gesehen. Auch den können wir selbst ausrechnen:

```
2 * pnorm(-abs(coef(est1)/sqrt(diag(vcov(est1)))))
```

```
   (Intercept)            str
0.000000000000 0.000002020858
```

Gerade haben wir Approximation durch die Normalverteilung benutzt. R verwendet in der summary die t-Verteilung:

```
2 * pt(-abs(coef(est1)/sqrt(diag(vcov(est1)))), est1$df.resid)
```

```
  (Intercept)           str
6.569925e-242  2.783307e-06
```

Wir sehen, die beiden Werte weichen etwas voneinander ab.
Die folgenden beiden Aussagen sind äquivalent:

- Das 95% Konfidenzintervall enthält nicht die Null
- Die Hypothese $\beta_1 = 0$ wird auf dem 5% Niveau abgelehnt

Darstellung von Schätzergebnissen:

$$\underset{}{\texttt{testscr} =} \underset{(9.4675)}{698.933} - \underset{(0.4798)}{2.2798} \cdot \texttt{str}, \quad R^2 = 0.05, \mathrm{SER} = 18.58$$

Oft finden wir Standardfehler in Klammern unter den geschätzten Koeffizienten. Diese Darstellung ist praktisch:

- Die geschätzte Regressionsgerade ist $\texttt{testscr} = 698.933 - 2.2798 \cdot \texttt{str}$
- Der Standardfehler von $\beta_0 = 9.4675$
- Der Standardfehler von $\beta_1 = 0.4798$
- Das $R^2 = 0.05$, der Standardfehler der Residuen ist $SER = 18.58$.

Damit hat man (fast) alle für den Hypothesentest und die Bestimmung der Konfidenzintervalle notwendigen Zahlen.

3.7 Metrische und kategoriale Variablen

- Metrisch / stetig:
  - Bruttosozialprodukt
  - Einkommen in Euro
  - `str`
- Kategorial / diskret
  - Geschlecht
  - Beruf
  - Sektor
  - Einkommen in Kategorien
- Binäre Variablen / Dummy-Variablen sind ein Spezialfall kategorialer Variablen
  - Geschlecht W/M
  - Einkommen größer als 40 000 Euro Ja/Nein
  - Arbeitslosigkeit Ja/Nein
  - Hochschulabschluss Ja/Nein

In der Gleichung

$$\texttt{testsrc} = \beta_0 + \beta_1 \texttt{str} + u$$

war die unabhängige Variable `str` metrisch. Was aber, wenn wir über `str` nur binäre Information haben?

$$\texttt{large} = \begin{cases} 1 & \text{falls } \texttt{str}>20 \\ 0 & \text{sonst} \end{cases}$$

Schätze nun

$$\texttt{testsrc} = \beta_0 + \beta_1 \texttt{large} + u$$

```
large <- Caschool$str > 20
est <- lm(testscr ~ large)
summary(est)
```

```
Call:
lm(formula = testscr ~ large)

Residuals:
     Min       1Q   Median       3Q      Max
-50.4346 -14.0707  -0.2845  12.7779  49.5654

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  657.185      1.202  546.62  < 2e-16 ***
largeTRUE     -7.185      1.852   -3.88 0.000121 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 18.74 on 418 degrees of freedom
Multiple R-squared: 0.03476,        Adjusted R-squared: 0.03245
F-statistic: 15.05 on 1 and 418 DF,  p-value: 0.0001215
```

Allgemein (wenn X eine binäre Variable / Dummy-variable ist)

$$Y_i = \beta_0 + \beta_1 X_i + u_i$$

Interpretation:

Falls $X_i = 0$: $Y_i = \beta_0 + u_i$

Der Mittelwert $\bar{Y} = \beta_0$

$E(Y_i|X_i = 0) = \beta_0$

Falls $X_i = 1$: $Y_i = \beta_0 + \beta_1 + u_i$

Der Mittelwert $\bar{Y} = \beta_0 + \beta_1$

$E(Y_i|X_i = 1) = \beta_0 + \beta_1$

Mithin ist $\beta_1 = E(Y_i|X_i = 1) - E(Y_i|X_i = 0)$ die Differenz der Mittelwerte der beiden Gruppen in der Population.

`t.test` vergleicht zwei Mittelwerte mit einem t-test.

`tapply` wendet eine Funktion (hier den Mittelwert `mean` und die Standardabweichung `sd`) für einzelne Gruppen eines Datensatzes an. Diese Gruppen werden hier durch die Variable `large` beschrieben.

```
est1 <- lm(testscr ~ large)
summary(est1)
```

```
Call:
lm(formula = testscr ~ large)

Residuals:
     Min       1Q   Median       3Q      Max
-50.4346 -14.0707  -0.2845  12.7779  49.5654

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  657.185      1.202  546.62  < 2e-16 ***
largeTRUE     -7.185      1.852   -3.88 0.000121 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 18.74 on 418 degrees of freedom
Multiple R-squared: 0.03476,        Adjusted R-squared: 0.03245
F-statistic: 15.05 on 1 and 418 DF,  p-value: 0.0001215
```

```
confint(est1)
```

```
                2.5 %     97.5 %
(Intercept) 654.82130 659.547833
largeTRUE   -10.82554  -3.544715
```

```
t.test(testscr ~ large)
```

```
        Welch Two Sample t-test

data:  testscr by large
t = 3.9231, df = 393.721, p-value = 0.0001031
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
  3.584445 10.785813
sample estimates:
mean in group FALSE  mean in group TRUE
           657.1846            649.9994
```

```
tapply(testscr, large, mean)
```

```
   FALSE     TRUE
657.1846 649.9994
```

```
tapply(testscr, large, sd)
```

```
   FALSE     TRUE
19.28629 17.96589
```

- Es ist egal ob wir mit einem Student t Test Mittelwerte zwischen Gruppen vergleichen,
- oder eine Regression mit einer binären Variablen rechnen.

Eine Regression zu verwenden, kann praktisch sein, wenn wir weitere Regressoren einführen.

3.8 Heteroskedastische und Homoskedastische Fehlerterme

Zur Erinnerung: Die drei OLS Annahmen:

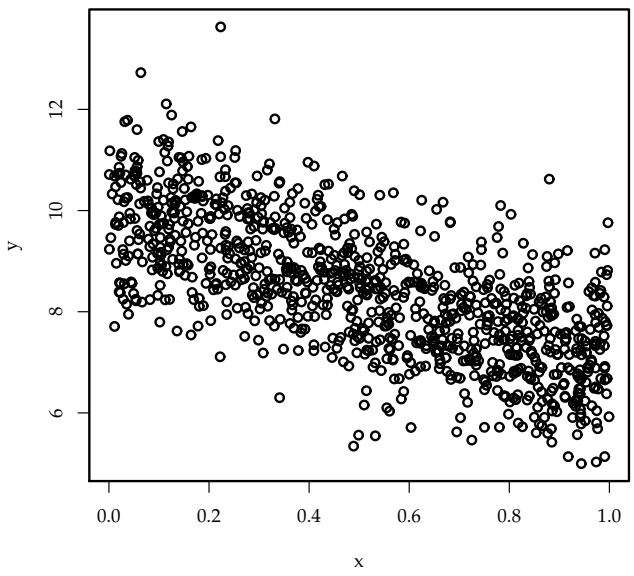
1. $E(u_i|X_i = x) = 0$
2. (X_i, Y_i) sind i.i.d.
3. Große Ausreißer in X und Y sind selten (die vierten Momente von X und Y existieren).

Nun kommt eine weitere Annahme:

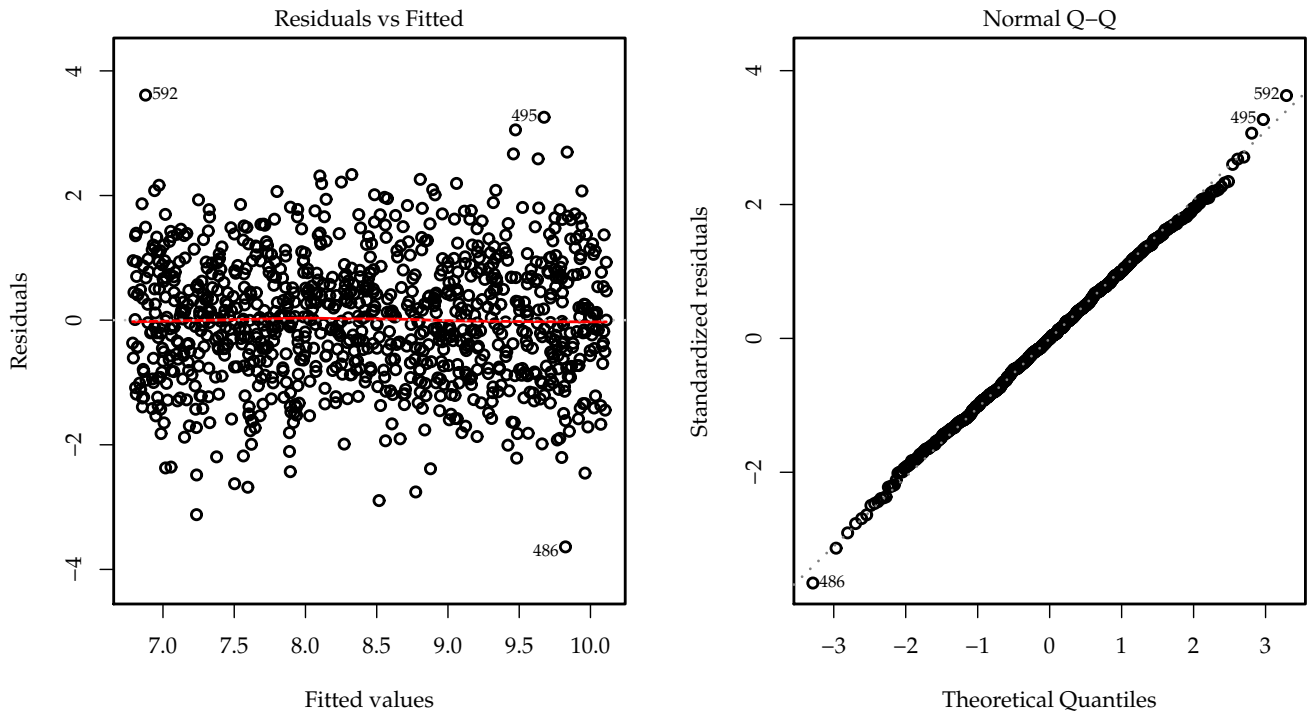
- $\text{var}(u|X = x)$ ist konstant, u ist homoskedastisch

  `runif` erzeugt einen Vektor gleichverteilter (pseudo-) Zufallsvariablen. Den brauchen wir hier für die Simulation eines Schätzmodells.

```
x <- runif(1000)
u <- rnorm(1000)
y <- 10 - 3.1 * x + u
plot(y ~ x)
```

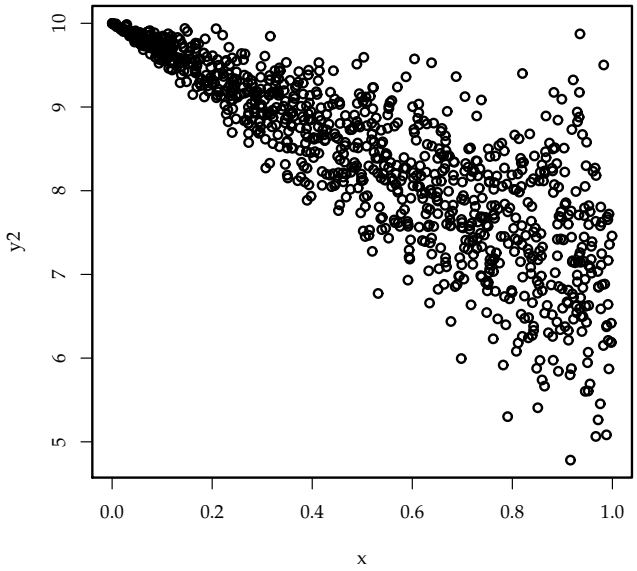


```
par(mfrow = c(1, 2))
est <- lm(y ~ x)
plot(est, which = 1:2)
```

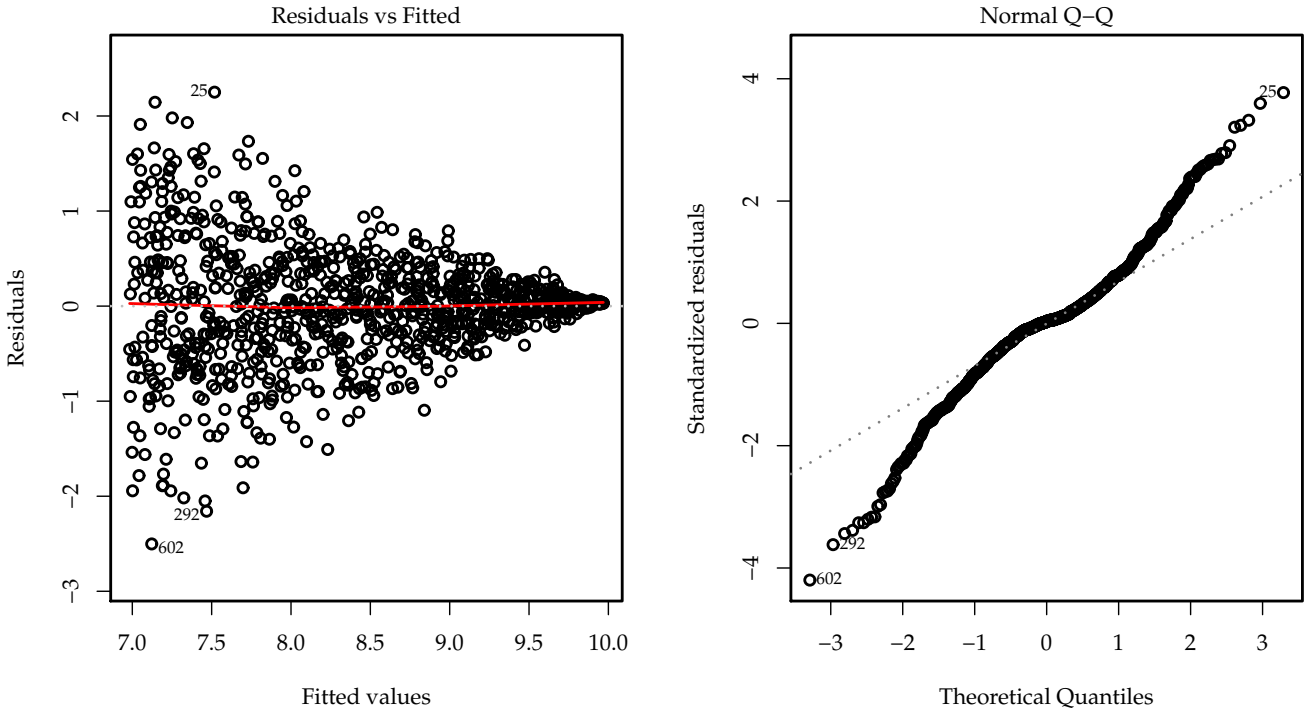


Im folgenden Beispiel sind die Störterme nicht mehr unabhängig von x

```
u2 <- rnorm(1000) * x
y2 <- 10 - 3.1 * x + u2
plot(y2 ~ x)
```




```
par(mfrow = c(1, 2))
est2 <- lm(y2 ~ x)
plot(est2, which = 1:2)
```



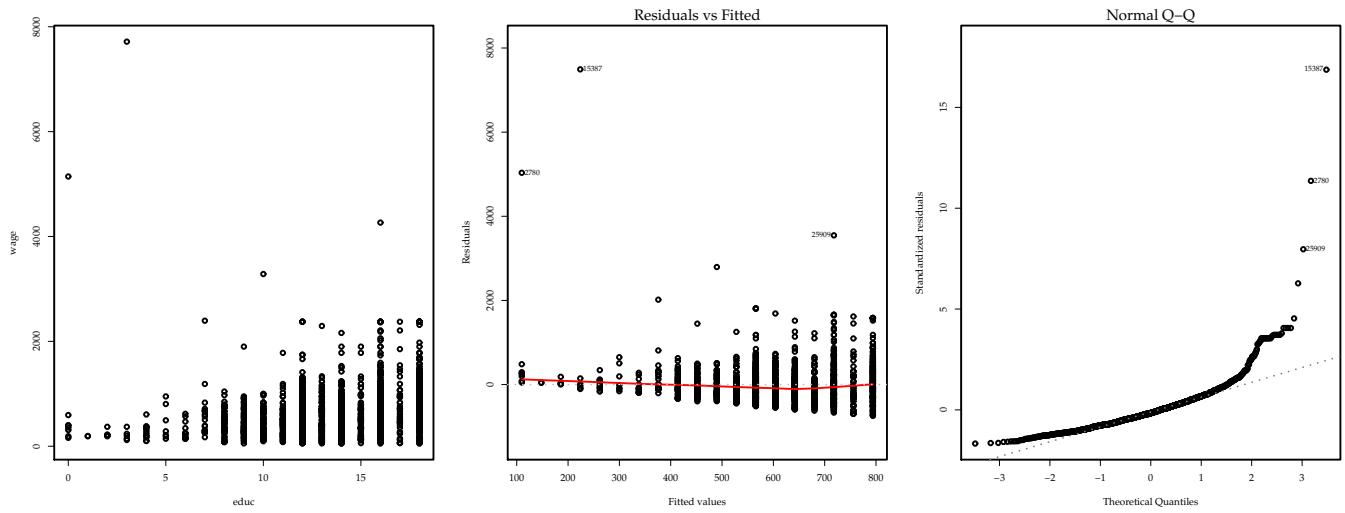
In beiden Beispielen ist $E(u_i|X_i = x) = 0$

Im ersten Beispiel ist u homoskedastisch
Im zweiten Beispiel ist u heteroskedastisch

3.8.1 Ein Beispiel aus der Arbeitsökonomie

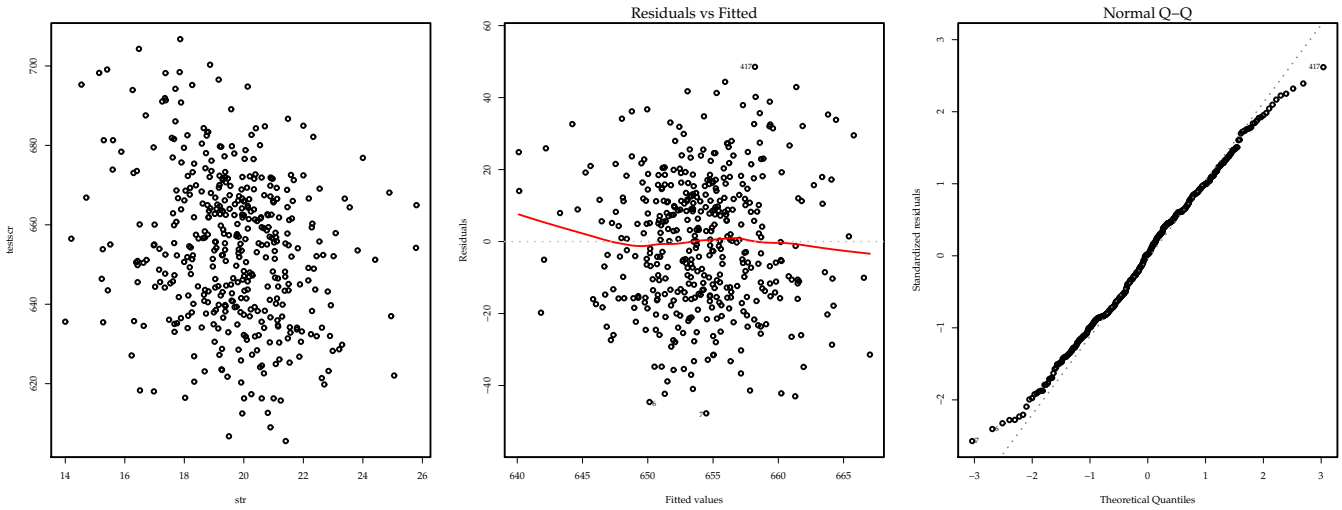
wage	weekly wages for US make workers from current population survey 1988
educ	years of education

```
par(mfrow = c(1, 3))
data(uswages, package = "faraway")
plot(wage ~ educ, data = uswages)
plot(lm(wage ~ educ, data = uswages), which = 1:2)
```



3.8.2 Zurück zu Caschool

```
par(mfrow = c(1, 3))
est <- lm(testscr ~ str)
plot(testscr ~ str)
plot(est, which = 1:2)
```



3.8.3 Was bringt uns Homoskedastizität?

- OLS hat minimale Varianz unter allen Schätzern, die linear in Y sind (Gauss-Markov Theorem)
- $\operatorname{var}\left(\hat{\beta}_1\right)$ kann leichter ausgerechnet werden.

Zur Erinnerung:

$$\begin{aligned}\operatorname{var}\left(\hat{\beta}_1\right) &= \frac{1}{n}\frac{\operatorname{var}((X_i-\mu_X)u_i)}{\sigma_X^4}\\ &= \frac{1}{n}\frac{E((X_i-\mu_X)^2u_i^2)}{\sigma_X^4} = \frac{\sigma_u^2}{n\cdot\sigma_X^2}\end{aligned}$$

Wir sehen, dass $\operatorname{var}(\hat{\beta}_1)$ sinkt wenn $\operatorname{var}(X)$ steigt.

$$s_{\hat{\beta}_1} = \sqrt{\frac{1}{n}\frac{\frac{1}{n-2}\sum_{i=1}^n \hat{u}_i^2}{\frac{1}{n}\sum_{i=1}^n (X_i-\bar{X})^2}}$$

Diese Formel für die Standardabweichung von $\hat{\beta}_1$ ist die Standardeinstellung in Statistikprogrammen und oft die einzige Möglichkeit in Officeprogrammen.
Ohne die Annahme $\operatorname{var}(u|X=x)$ ist konstant gilt aber:

$$\operatorname{var}\left(\hat{\beta}_1\right) = \frac{1}{n}\frac{\operatorname{var}((X_i-\mu_X)u_i)}{\sigma_X^4}$$

Was ist, wenn $(X_i - \mu_X)$ und $\text{var}(u_i)$ nicht unabhängig sind?

Heteroskedastizität (immer richtig):

$$\hat{\sigma}_{\hat{\beta}_1} = \sqrt{\frac{1}{n} \frac{\frac{1}{n-2}\sum_{i=1}^{n} \hat{v}^2}{\left(\frac{1}{n}\sum_{i=1}^{n}\left(X_i - \bar{X}\right)^2\right)^2}}$$

$$(\text{mit } \hat{v} = \left(X_i - \bar{X}\right) \cdot \hat{u}_i)$$

Homoskedastizität:

$$\hat{\sigma}_{\hat{\beta}_1} = \sqrt{\frac{1}{n} \frac{\frac{1}{n-2}\sum_{i=1}^{n} \hat{u}_i^2}{\frac{1}{n}\sum_{i=1}^{n}(X_i - \bar{X})^2}}$$

Die Formel für den Fall homoskedastischer Störterme ist einfacher, aber nur richtig, wenn tatsächlich die Annahme homoskedastischer Störterme erfüllt ist.

- Da die Formeln unterschiedlich sind, bekommt man normalerweise ein unterschiedliches Ergebnis.
- Homoskedastizität ist die Standardeinstellung in der Software (wenn nicht die einzig mögliche).

  Damit bekommt man typischerweise kleinere Standardfehler für $\hat{\beta}$ als mit der Einstellung für Heteroskedastizität.

  `hccm` berechnet die Varianz-Kovarianz Matrix für $\hat{\beta}$ unter der Annahme heteroskedastischer Residuen.

```
est <- lm(testscr ~ str)
summary(est)
```

```
Call:
lm(formula = testscr ~ str)

Residuals:
     Min       1Q   Median       3Q      Max
-47.7267 -14.2507   0.4826  12.8222  48.5404
```

```
Coefficients:
            Estimate Std. Error t value   Pr(>|t|)
(Intercept) 698.9330     9.4675  73.825    < 2e-16 ***
str          -2.2798     0.4798  -4.751 0.00000278 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 18.58 on 418 degrees of freedom
Multiple R-squared: 0.05124,        Adjusted R-squared: 0.04897
F-statistic: 22.58 on 1 and 418 DF,  p-value: 0.000002783
```

R benutzt für p-Werte und Konfindenzintervalle die homoskedastischen Standardfehler.

```
sqrt(vcov(est))
```

```
            (Intercept)       str
(Intercept)    9.467491       NaN
str                 NaN 0.4798256
```

```
sqrt(hccm(est))
```

```
            (Intercept)       str
(Intercept)    10.46053       NaN
str                 NaN 0.5243585
```

3.8.4 Zusammenfassung

- Wenn die Daten homoskedastisch sind, und wir Heteroskedastizität annehmen, sind wir auf der sicheren Seite.
- Wenn die Daten heteroskedastisch sind, und wir Homoskedastizität annehmen, sind die Standardfehler falsch (der Schätzer ist dann nicht konsistent)
- Beide Formeln geben für große n und bei Homoskedastizität das gleiche Ergebnis.

$\rightarrow$ man sollte immer heteroskedastizität-robuste Standardfehler verwenden

Was wir über OLS wissen:

- OLS ist unverzerrt
- OLS ist konsistent
- wir können Konfidenzintervalle für $\hat{\beta}$ berechnen
- wir können Hypothesen über $\hat{\beta}$ testen

Ein großer Teil ökonometrischer Auswertung wird in Form von OLS präsentiert. Ein Grund das zu machen ist schon, dass sehr viele Leute verstehen, wie OLS funktioniert.
Wenn wir einen anderen Schätzer verwenden, kann es sein, dass man uns nicht mehr versteht.

- Reicht das als Begründung um OLS zu verwenden?
- Gibt es bessere Schätzer? Schätzer mit kleinerer Varianz?

Um diese Fragen zu beantworten, werden wir weitere Annahmen machen

3.9 Erweiterte OLS Annahmen

1. $E(u_i|X_i = x) = 0$
2. (X_i, Y_i) sind i.i.d.
3. Große Ausreißer in X und Y sind selten (die vierten Momente von X und Y existieren).
4. $\text{var}(u|X = x)$ ist konstant, u ist homoskedastisch
5. u ist normalverteilt $u \sim N(0, \sigma^2)$

Annahmen 4 und 5 sind restriktiver — lassen sich also seltener rechtfertigen.

Gauss Markov

Unter den Annahmen 1-4 hat $\hat{\beta}_1$ die kleinste Varianz unter allen linearen Schätzern (unter allen Schätzer die lineare Funktionen von Y sind).

Effizienz von OLS-II
Unter den Annahmen 1-5 hat $\hat{\beta}_1$ die kleinste Varianz unter allen konsistenten Schätzern wenn $n \to \infty$ (egal ob sie linear oder nichtlinear sind)

3.10 Probleme mit OLS

Gauss Markov:

- Die Annahmen des Gauss Markov Theorems (Homoskedastizität) sind oft nicht erfüllt.
- Das Resultat gilt nur für lineare Schätzer. Das ist nur ein kleiner Teil aller möglichen Schätzer.

kleinste Varianz unter allen konsistenten Schätzern erfordert homoskedastische normalverteilte Residuen – das ist oft nicht plausibel (z.B. wenn wir über Löhne reden)

Ausreißer: OLS ist sensitiver zu Ausreißern als viele andere Schätzer.

Erinnern wir uns: Schätzung des Mittelwertes: Der Median ist weniger sensitiv als der empirische Mittelwert gegenüber Ausreißern.

Ähnliche Dinge kann man auch beim Schätzen linearer Gleichungen machen:

$$\text{OLS: } \min_{b_0,b_1} \sum_{i=1}^{n} (Y_i - (b_0 + b_1 X_i))^2$$

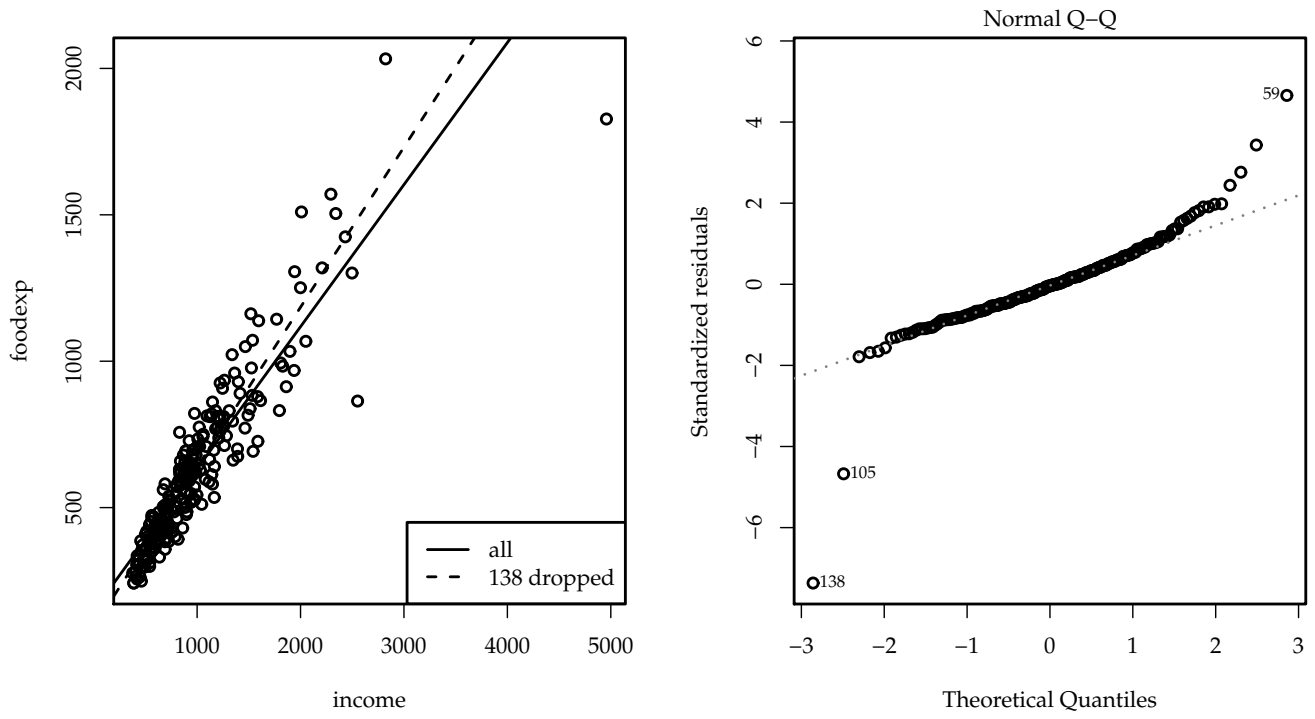
$$\text{LAD: } \min_{b_0,b_1} \sum_{i=1}^{n} |Y_i - (b_0 + b_1 X_i)|$$

in fast allen Anwendungen wird jedoch OLS verwendet – das werden wir hier auch machen.

3.10.1 Alternativen

- Identifizieren und Eliminieren von Ausreißern
- Quantilsregression
- Robuste Regression

```
library(quantreg)
data(engel)
attach(engel)
par(mfrow = c(1, 2))
plot(foodexp ~ income)
est <- lm(foodexp ~ income)
abline(est)
abline(lm(foodexp ~ income, data = engel[-138, ]), lty = 2)
legend("bottomright", c("all", "138 dropped"), lty = 1:2)
plot(est, which = 2)
```



Ist Beobachtung 138 ein Ausreißer?

3.10.2 Robuste Regression

Bislang haben wir Kleinste Quadrate minimiert:

$$\sum \left(y_i - (\beta_0 + \beta_1 x_i)\right)^2$$

Allgemein minimieren wir die Summe irgendeiner Funktion:

$$\sum \rho\left(y_i - (\beta_0 + \beta_1 x_i)\right)$$

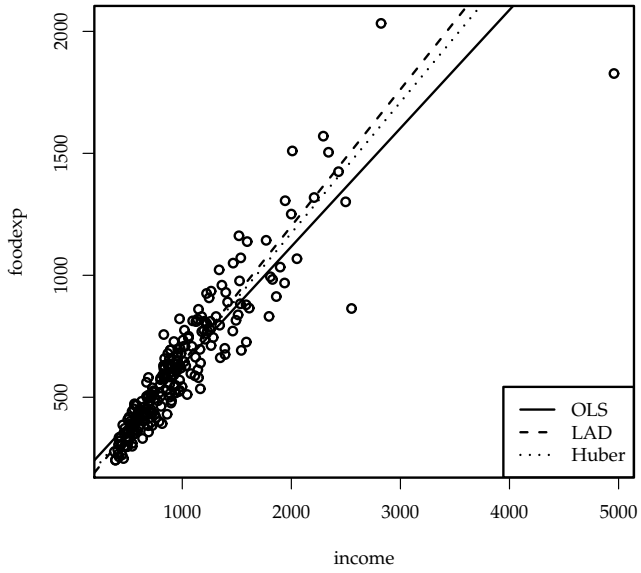
wobei

1. $\rho(x) = x^2$ OLS
2. $\rho(x) = |x|$ LAD (Quantilsregression)
3. $\rho(x) = \begin{cases} x^2/2 & \text{falls } |x| \leq c \\ c|x| - c^2/2 & \text{sonst} \end{cases}$

   Hubers Methode. c ist ein Schätzwert für σ_u.

   `rq` berechnet allgemein eine Quantilregression. Dazu wird die Summe der absoluten Residuen minimiert. `rlm` berechnet eine robuste Regression.

```
library(MASS)
plot(foodexp ~ income)
abline(lm(foodexp ~ income))
abline(rq(foodexp ~ income), lty = 2)
abline(rlm(foodexp ~ income), lty = 3)
legend("bottomright", c("OLS", "LAD", "Huber"), lty = 1:3)
```



4 Modelle mit mehr als einer unabhängigen Variablen (multiple Regression)

$$\texttt{testsrc} = \beta_1 \texttt{str} + \beta_0 + u$$

- `testscr` test score
- `str` student / teacher ratio

Wie kann man mehrere Faktoren gleichzeitig berücksichtigen?

- Halte einen Faktor „konstant“ indem nur eine kleine Gruppe betrachtet wird (z.B. alle Schüler mit einem sehr ähnlichen `elpct` (english learner percentage)

  Die Option `subset` im Kommando `lm` beschränkt die Schätzung auf einen Teil des Datensatzes.

```
data(Caschool)
attach(Caschool)
summary(elpct)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  0.000   1.941   8.778  15.770  22.970  85.540
```

```
lm(testscr ~ str, subset = (elpct < 9))
```

```
Call:
lm(formula = testscr ~ str, subset = (elpct < 9))

Coefficients:
(Intercept)          str
    680.252       -0.835
```

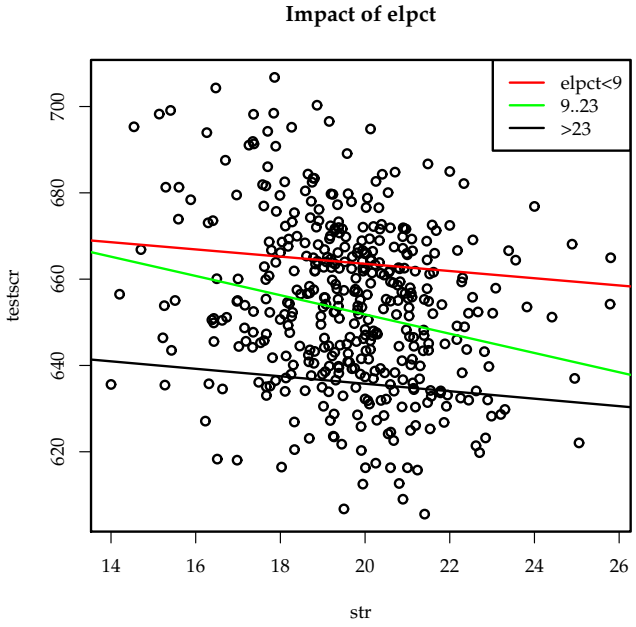
```
lm(testscr ~ str, subset = (elpct >= 9 & elpct < 23))
```

```
Call:
lm(formula = testscr ~ str, subset = (elpct >= 9 & elpct < 23))

Coefficients:
(Intercept)          str
    696.445       -2.231
```

```
lm(testscr ~ str, subset = (elpct >= 23))
```

```
Call:
lm(formula = testscr ~ str, subset = (elpct >= 23))

Coefficients:
(Intercept)          str
   653.0746      -0.8656
```

```
plot(testscr ~ str, main = "Impact of elpct")
abline(lm(testscr ~ str, subset = (elpct < 9)), col = "red")
abline(lm(testscr ~ str, subset = (elpct >= 9 & elpct <
    23)), col = "green")
abline(lm(testscr ~ str, subset = (elpct >= 23)))
legend("topright", c("elpct<9", "9..23", ">23"), lty = 1,
    col = c("red", "green", "black"))
```



abhängig von `elpct` sind die geschätzten Zusammenhänge sehr unterschiedlich

- erweitere das Regressionsmodell

$$\texttt{testsrc} = \beta_1 \texttt{str} + \beta_2 \texttt{elpct} + \beta_0 + u$$

allgemein:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k + u$$

für jede Beobachtung:

$$\begin{aligned}
y_1 &= \beta_0 + \beta_1 x_{11} + \beta_2 x_{12} + \cdots + \beta_k x_{1k} + u_1 \\
y_2 &= \beta_0 + \beta_1 x_{21} + \beta_2 x_{22} + \cdots + \beta_k x_{2k} + u_2 \\
y_3 &= \beta_0 + \beta_1 x_{31} + \beta_2 x_{32} + \cdots + \beta_k x_{3k} + u_3 \\
&\vdots \\
y_n &= \beta_0 + \beta_1 x_{n1} + \beta_2 x_{n2} + \cdots + \beta_k x_{nk} + u_n
\end{aligned}$$

```
lm(testscr ~ str + elpct)
```

```
Call:
lm(formula = testscr ~ str + elpct)

Coefficients:
(Intercept)          str        elpct
   686.0322      -1.1013      -0.6498
```

4.1 Matrixschreibweise

4.1.1 Matrixschreibweise

$$y = X\beta + u$$

$$y = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{pmatrix}; \quad X = \begin{pmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1k} \\ 1 & x_{21} & x_{22} & \cdots & x_{2k} \\ 1 & x_{31} & x_{32} & \cdots & x_{3k} \\ \vdots & & & \ddots & \vdots \\ 1 & x_{n1} & x_{n2} & \cdots & x_{nk} \end{pmatrix}; \quad \beta = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_k \end{pmatrix}; \quad u = \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \\ u_n \end{pmatrix};$$

Addition

$$\underbrace{\begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}}_{n\times m} + \underbrace{\begin{pmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{pmatrix}}_{n\times m} = \underbrace{\begin{pmatrix} a_{11}+b_{11} & a_{12}+b_{12} & a_{13}+b_{13} \\ a_{21}+b_{21} & a_{22}+b_{22} & a_{23}+b_{23} \\ a_{31}+b_{31} & a_{32}+b_{32} & a_{33}+b_{33} \end{pmatrix}}_{n\times m}$$

Multiplikation

$$\underbrace{\begin{pmatrix} \cdots & \cdots & \cdots \\ a_{21} & a_{22} & a_{23} \\ \cdots & \cdots & \cdots \end{pmatrix}}_{n\times m} \cdot \underbrace{\begin{pmatrix} \cdots & b_{12} & \cdots \\ \cdots & b_{22} & \cdots \\ \cdots & b_{32} & \cdots \end{pmatrix}}_{m\times k} = \underbrace{\begin{pmatrix} \cdots & \cdots & \cdots \\ \cdots & \sum_{i=1}^{m} a_{2i}\cdot b_{i2} & \cdots \\ \cdots & \cdots & \cdots \end{pmatrix}}_{n\times k}$$

4.2 Herleitung des OLS Schätzers in Matrixschreibweise

Die Residuen ergeben sich nun als

$$u = y - X\beta$$

Die Quadratsumme der Residuen also

$$\begin{aligned} S(\beta) = \sum_{i=1}^{n} u_i^2 = u'u &= (y - X\beta)'(y - X\beta) \\ &= y'y - y'X\beta - \beta'X'y + \beta'X'X\beta \\ &= y'y - 2\beta'X'y + \beta'X'X\beta \end{aligned}$$

(zur Erinnerung: $(AB)' = B'A'$)
Um $S(\beta)$ zu minimieren, bilden wir die Ableitung nach β:

$$\frac{\partial S(\beta)}{\partial \beta} = -2X'y + 2X'X\beta \stackrel{!}{=} 0$$

$$\text{Normalengleichung:} \qquad X'X\hat{\beta} = X'y \tag{7}$$

Nun ist $X'X$ eine $(k+1) \times (k+1)$ Matrix. Falls diese Matrix nicht singulär ist, bilden wir die Inverse

$$\begin{aligned} (X'X)^{-1}X'X\hat{\beta} &= (X'X)^{-1}X'y \\ \hat{\beta} &= \underbrace{(X'X)^{-1}X'}_{X^+} y \end{aligned}$$

$$X^+ = (X'X)^{-1}X'$$

zur Übung: zeigen Sie

- $XX^+X = X$
- $X^+XX^+ = X^+$
- $(XX^+)' = XX^+$
- $X^+X = I$

Nenne

$$\hat{y} = X\hat{\beta} \qquad \hat{u} = y - \hat{y}$$

Orthogonalität

$$X'\hat{u} = X'(y - \hat{y}) = X'y - X'X\hat{\beta} = X'y - X'X(X'X)^{-1}X'y = 0$$
$$\hat{y}'\hat{u} = \hat{\beta}'X'\hat{u} = 0$$

Multiplikation der Normalengleichung $X'X\hat{\beta} = X'y$ mit $\hat{\beta}'$ ergibt

$$\hat{\beta}'X'X\hat{\beta} = \hat{\beta}'X'y$$

dann ist

$$\begin{aligned}
\hat{u}'\hat{u} &= (y - X\hat{\beta})'(y - X\hat{\beta}) \\
&= y'y - 2\hat{\beta}'X'y + \hat{\beta}'X'X\hat{\beta} \\
&= y'y - 2\hat{\beta}'X'X\hat{\beta} + \hat{\beta}'X'X\hat{\beta} \\
&= y'y - \hat{\beta}'X'X\hat{\beta} \\
&= y'y - (X\hat{\beta})'X\hat{\beta} \\
&= y'y - \hat{y}'\hat{y}
\end{aligned}$$

Quadratische Zerlegung (Streuungszerlegung:)

$$y'y = \hat{y}'\hat{y} + \hat{u}'\hat{u}$$

Im Fall der inhomogenen Regression: $y = X\beta + u$

$$y = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{pmatrix}; \quad X = \begin{pmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1k} \\ 1 & x_{21} & x_{22} & \cdots & x_{2k} \\ 1 & x_{31} & x_{32} & \cdots & x_{3k} \\ \vdots & & & \ddots & \vdots \\ 1 & x_{n1} & x_{n2} & \cdots & x_{nk} \end{pmatrix}; \quad \beta = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_k \end{pmatrix}; \quad u = \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \\ u_n \end{pmatrix};$$

$$\frac{1}{n}y'y = \frac{1}{n}\hat{y}'\hat{y} + \frac{1}{n}\hat{u}'\hat{u}$$
$$\frac{1}{n}y'y - \bar{y}^2 = \frac{1}{n}\hat{y}'\hat{y} - \bar{y}^2 + \frac{1}{n}\hat{u}'\hat{u}$$

$$\underbrace{s_y^2}_{TSS} = \underbrace{s_{\hat{y}}^2}_{ESS} + \underbrace{s_{\hat{u}}^2}_{SSR}$$

TSS	total sum of squares
ESS	explained sum of squares
SSR	sum of squares of residuals

$$R^2 = \frac{s^2_{\hat{y}}}{s^2_y} = 1 - \frac{s^2_{\hat{u}}}{s^2_y}$$

4.3 Spezifikationsfehler

Was kann passieren, wenn wir eine Variable in unserem Modell vergessen? Betrachten wir nochmals unsere einfache Schätzgleichung:

$$\texttt{testsrc} = \beta_1 \texttt{str} + \beta_0 + u$$

- `testscr` test score
- `str` student / teacher ratio

Was könnte sonst noch einen Einfluss auf `testscr` haben?

	korr. mit Regressor `str`	beeinflusst abh. Var. `testscr`
percent of English learners	x	x
time of day of the test		x
parking lot space per pupil	x	

Wenn wir eine Variable nicht in unsere Schätzgleichung aufnehmen, diese Variable aber

- mit dem Regressor korreliert ist
- und die abhängige Variable beeinflusst

ist unsere Schätzung für β verzerrt (omitted variable bias)

$\rightarrow$ Die Annahme $E(u_i|X_i) = 0$ ist nicht mehr erfüllt

4.3.1 Beispiele:

- Klassische Musik $\rightarrow$ Intelligenz von Kindern (Rauscher, Shaw, Ky; Nature; 1993)

  (fehlende Variable: Einkommen)

- French paradox: Rotwein, Leberpastete → weniger Erkrankungen der Herzkranzgefäße (Samuel Black, 1819)

  (fehlende Variable: Fisch und Zucker in der Ernährung,...)

- Störche in Niedersachsen → Geburtenrate

  (fehlende Variable Faktoren: Industrialisierung)

4.3.2 Spezifikationsfehler allgemein

Das wahre Modell sei

$$y = X_1\beta_1 + X_2\beta_2 + u$$

was passiert, wenn wir in der Spezifikation des Modells X_2 vergessen?

$$\widehat{\beta} = \underbrace{(X'X)^{-1}X'}_{X^+} y$$

$$\begin{aligned} b_1 &= (X_1'X_1)^{-1}X_1'y \\ &= (X_1'X_1)^{-1}X_1'(X_1\beta_1 + X_2\beta_2 + u) \\ &= (X_1'X_1)^{-1}X_1'X_1\beta_1 + (X_1'X_1)^{-1}X_1'X_2\beta_2 + (X_1'X_1)^{-1}X_1'u \\ E(b_1) &= \beta_1 + (X_1'X_1)^{-1}X_1'X_2\beta_2 \end{aligned}$$

dieser Ausdruck ist $= \beta_1$ nur wenn

- $\beta_2 = 0$
- oder $X_1'X_2 = 0$, d.h. X_1 und X_2 sind orthogonal

4.4 Annahmen für das multiple Regressionsmodell

1. $E(u_i|X_i = x) = 0$
2. (X_i, Y_i) sind i.i.d.
3. Große Ausreißer in X und Y sind selten (die vierten Momente von X und Y existieren).
4. X hat Rang gleich der Anzahl der Spalten (keine Multikollinearität)

5. $\text{var}(u|X = x)$ ist konstant, u ist homoskedastisch
6. u ist normalverteilt $u \sim N(0, \sigma^2)$

4.5 Die Verteilung der OLS Schätzer in der multiplen Regression

↑ Modell mit einem Regressor: die OLS Schätzer für $\hat{\beta}_0$ und $\hat{\beta}_1$ sind unverzerrte und konsistente Schätzer. Für große Samples sind $\hat{\beta}_0$ und $\hat{\beta}_1$ normalverteilt.

- multiple Regression: unter obigen Annahmen 1–4 ist der OLS Schätzer $\hat{\beta} = (X'X)^{-1}X'y$ unverzerrt und konsistent. Für große Samples ist $\hat{\beta}$ multivariat normalverteilt.

4.6 Multikollinearität

Beispiel

$$\texttt{testscr} = \beta_1 \texttt{str} + \beta_2 \texttt{elpct} + \beta_0$$

Jetzt erweitern wir das Modell um eine Variable: Anteil der „English learners“ `FracEL=elpct/100`:

$$\texttt{testscr} = \beta_1 \texttt{str} + \beta_2 \texttt{elpct} + \beta_3 \texttt{FracEL} + \beta_0$$

```
lm(testscr ~ str + elpct)
```

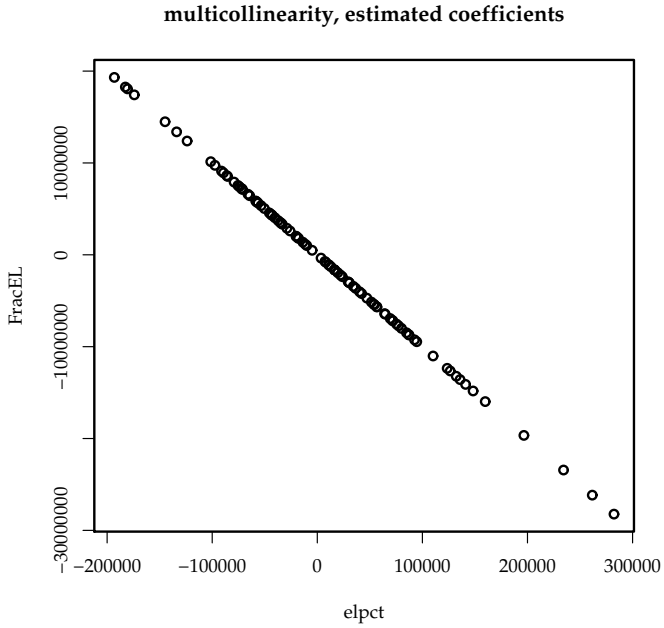
```
Call:
lm(formula = testscr ~ str + elpct)

Coefficients:
(Intercept)          str        elpct
   686.0322      -1.1013      -0.6498
```

```
FracEL <- elpct/100
summary(lm(testscr ~ str + elpct + FracEL))
```

```
Call:
lm(formula = testscr ~ str + elpct + FracEL)

Residuals:
     Min       1Q   Median       3Q      Max
-48.8447 -10.2404  -0.3078   9.8153  43.4607

Coefficients: (1 not defined because of singularities)
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 686.03225    7.41131  92.566  < 2e-16 ***
str          -1.10130    0.38028  -2.896  0.00398 **
elpct        -0.64978    0.03934 -16.516  < 2e-16 ***
FracEL             NA         NA      NA       NA
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 14.46 on 417 degrees of freedom
Multiple R-squared: 0.4264,        Adjusted R-squared: 0.4237
F-statistic:   155 on 2 and 417 DF,  p-value: < 2.2e-16
```

Wir sehen, R erkennt selbst, dass Multikolliniearität vorliegt, und vereinfacht das Modell entsprechend.
Das gelingt aber nicht immer.
Wir perturbieren diese Variable nun etwas. Das kann auch unabsichtlich (z.B. durch Rundungsfehler) passieren. Die Variablen sind dann nicht mehr (perfekt) Multikollinear. Wir bekommen deshalb Ergebnisse für alle Koeffizienten. Für jede (kleine) Perturbation ändert sich das Ergebnis erheblich. Die Standardfehler werden sehr groß.

```
set.seed(123)
perturbedEstimate <- function(x) {
    FracEL = elpct/100 + rnorm(4) * 0.0000001
    est <- lm(testscr ~ str + elpct + FracEL)
    coef(est)[3:4]
}
estList <- sapply(1:100, perturbedEstimate)
plot(t(estList), main = "multicollinearity, estimated coefficients")
```



Große Koeffizienten für `elpct` werden ausgeglichen durch kleine von `FracEL`. Was ist passiert? Dies ist der wahre Zusammenhang:

$$\texttt{testscr} = 686.0322 - 1.1013\texttt{str} - 0.6498\texttt{elpct}$$

Nun sei $\texttt{FracEL} = \texttt{elpct}/100$

$$\texttt{testscr} = 686.0322 - 1.1013 \cdot \texttt{str} + (\underline{a} - 0.6498) \cdot \texttt{elpct} - \underline{100a \cdot \texttt{elpct}/100}$$

$$\texttt{testscr} = 686.0322 - 1.1013 \cdot \texttt{str} + (a - 0.6498) \cdot \texttt{elpct} - 100a \cdot \texttt{FracEL}$$

Koeffizienten β können nicht mehr identifiziert werden.

4.6.1 Beispiel 2

Dummy Variable mit dem Wert 1 falls `str>12` (Klasse ist nicht „sehr“ klein)

```
NSK <- str > 12
lm(testscr ~ str + elpct + NSK)
```

```
Call:
lm(formula = testscr ~ str + elpct + NSK)

Coefficients:
```

```
(Intercept)          str        elpct      NSKTRUE
   686.0322      -1.1013      -0.6498           NA
```

Hier kann der Koeffizient von NSK nicht geschätzt werden. Warum?

```
table(NSK)
```

```
NSK
TRUE
 420
```

Die neue Variable NSK ist immer TRUE und damit perfekt kollinear zum konstanten Term. Grund: Es gibt keine Klassen mit $\texttt{str} < 12$, also können wir auch nicht den Effekt einer solchen Klassengröße beurteilen.

4.6.2 Beispiel 3

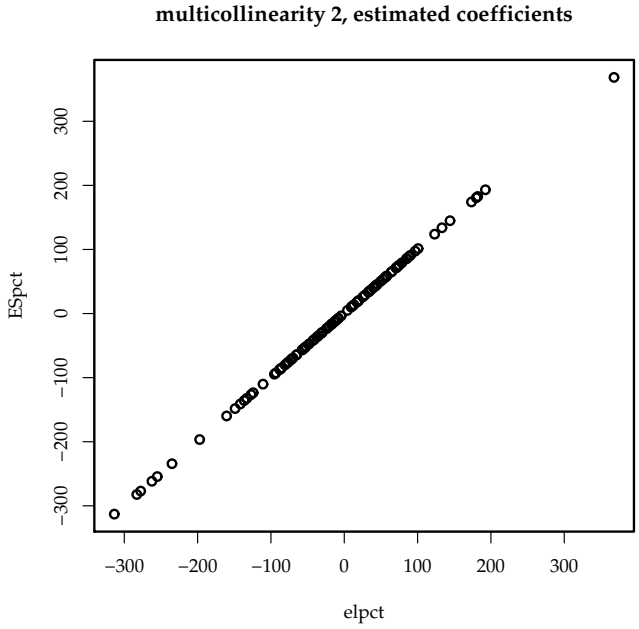
$\texttt{ESpct} = 100 - \texttt{elpct}$

```
ESpct <- 100 - elpct
lm(testscr ~ str + elpct + ESpct)
```

```
Call:
lm(formula = testscr ~ str + elpct + ESpct)

Coefficients:
(Intercept)          str        elpct        ESpct
   686.0322      -1.1013      -0.6498           NA
```

Wieder stellt R Kollinearität fest. Eine kleine Perturbation reicht jedoch, ein „Ergebnis“ zu erhalten. Allerdings ist dieses Ergebnis nicht gerade aussagekräftig.

```
set.seed(123)
perturbedEstimate2 <- function(x) {
    ESpct <- 100 - elpct + rnorm(4) * 0.01
    est <- lm(testscr ~ str + elpct + ESpct)
    coef(est)[3:4]
}
estList <- sapply(1:100, perturbedEstimate2)
plot(t(estList), main = "multicollinearity 2, estimated coefficients")
```



Der wahre Zusammenhang ist:

$$\texttt{testscr} = 686.0322 - 1.1013\texttt{str} - 0.6498\texttt{elpct}$$

Nun sei $\texttt{ESpct} = 100 - \texttt{elpct}$

$$\texttt{testscr} = 686.0322 - a \cdot \underline{100} - 1.1013 \cdot \texttt{str} - (0.6498 - \underset{\sim}{a}) \cdot \texttt{elpct} + a \cdot (\underline{100} - \underset{\sim\sim\sim\sim}{\texttt{elpct}})$$

$$\texttt{testscr} = 686.0322 - a \cdot 100 - 1.1013 \cdot \texttt{str} - (0.6498 - a) \cdot \texttt{elpct} + a \cdot \texttt{ESpct}$$

Koeffizienten β können nicht mehr identifiziert werden.

4.6.3 Welcher Regressor ist verantwortlich für Kollinearität?

Wir regressieren alle k Regressoren auf die verbleibenden $1 - k$ Regressoren:

$$x_i = \beta_0 + \beta_1 x_1 + \ldots + \beta_{i-1} x_{i-1} + \beta_{i+1} x_{i+1} + \beta_k x_k + u$$

Ein großes R_i^2 ist ein Anzeichen für Kollinearität.
Wir betrachten deshalb den Variance Inflation Factor

$$\text{VIF} = \frac{1}{1 - R_i^2}$$

Im folgenden Beispiel konstruieren wir einen (fast) linear abhängigen Wert, `elpct2`. Außerdem fügen wir als offensichtlich sinnlosen Regressor die Nummer des Distrikts in unsere Gleichung ein.

```
set.seed(123)
elpct2 <- elpct + rnorm(4)
est <- lm(testscr ~ str + elpct2 + elpct + as.numeric(district))
summary(est)
```

```
Call:
lm(formula = testscr ~ str + elpct2 + elpct + as.numeric(district))

Residuals:
     Min       1Q   Median       3Q      Max
-48.3282 -10.2119  -0.1676   9.5185  43.8725

Coefficients:
                       Estimate Std. Error t value Pr(>|t|)
(Intercept)          685.454379   7.617035  89.990   <2e-16 ***
str                   -1.096242   0.381800  -2.871   0.0043 **
elpct2                 0.544466   0.874086   0.623   0.5337
elpct                 -1.196071   0.876663  -1.364   0.1732
as.numeric(district)   0.001910   0.006071   0.315   0.7532
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 14.49 on 415 degrees of freedom
Multiple R-squared: 0.4271,        Adjusted R-squared: 0.4216
F-statistic: 77.34 on 4 and 415 DF,  p-value: < 2.2e-16
```

Wir sehen, es gibt einige Faktoren mit großer Varianz. Um Kollinearität zu testen, berechnen wir den variance inflation factor:

`model.matrix` extrahiert die X-Matrix aus einer Regression. Hier lassen wir mit `[,-1]` die erste Spalte (Intercept) weg.

```
X <- model.matrix(est)[, -1]
names <- colnames(X)
```

```
sapply(1:length(names), function(i) {
    r2 <- summary(lm(X[, i] ~ X[, -i]))$r.squared
    vif <- 1/(1 - r2)
    names(vif) = names[i]
    vif
})
```

```
                str               elpct2                elpct
           1.040984           512.718019           512.761866
as.numeric(district)
           1.008520
```

Wir sehen (wenn wir es nicht schon vorher gewusst hätten), die Nummer des Distrikts ist zwar nicht signifikant, aber auch nicht kollinear. Die beiden Varianten von `elpct` sind jedoch kollinear. Wenn man eine wegstreicht, wird die Varianz der anderen kleiner.

```
summary(lm(testscr ~ str + elpct + as.numeric(district)))
```

4.6.4 Multikollinearität von Dummy Variablen

$$\begin{matrix} \text{Konstante} & \text{Gruppe 1} & \text{Gruppe 2} & \text{Gruppe 3} \end{matrix}$$
$$\begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{pmatrix}$$

Diese Matrix hat nicht vollen Spaltenrang

4.7 Spezifikationsfehler: Zusammenfassung

- unterspezifiziertes Modell, ein Regressor β_2 fehlt:
  - $\hat{\beta}$ ist nur unverzerrt wenn $\beta_2 = 0$ oder $X_1'X_2 = 0$.
- überspezifiziertes Modell, Regressoren sind kollinear:
  - $\hat{\beta}$ kann nicht geschätzt werden ($X'X$ ist nicht invertiertbar)
- überspezifiziertes Modell, Regressoren sind fast kollinear:
  - $\hat{\beta}$ kann nur ungenau geschätzt werden

4.8 Die Verteilung von $\hat{\beta}$

4.8.1 Varianz von $\hat{\beta}$

Im Fall der einfachen Regression (zur Erinnerung):

$$\hat{\sigma}^2_{\hat{\beta}_1} = \frac{1}{n} \frac{\frac{1}{n-2}\sum_{i=1}^n \hat{u}_i^2}{\frac{1}{n}\sum_{i=1}^n (X_i - \bar{X})^2} \qquad \text{Homoskedastizität}$$

$$\hat{\sigma}^2_{\hat{\beta}_1} = \frac{1}{n} \frac{\frac{1}{n-2}\sum_{i=1}^n \hat{v}^2}{\left(\frac{1}{n}\sum_{i=1}^n \left(X_i - \bar{X}\right)^2\right)^2} \qquad \text{Heteroskedastizität (immer richtig)}$$

$$(\text{mit } \hat{v} = \left(X_i - \bar{X}\right) \cdot \hat{u}_i)$$

Im Fall der multiplen Regression:

$$\Sigma_{\hat{\beta}\hat{\beta}} = \hat{\sigma}_u^2 (X'X)^{-1} \qquad \text{Homoskedastizität}$$

$$\Sigma_{\hat{\beta}\hat{\beta}} = (X'X)^{-1} X' \mathbf{I} \mathbf{u}^2 X (X'X)^{-1} \qquad \text{Heteroskedastizität (immer richtig)}$$

4.8.2 Imperfekte Multikollinearität

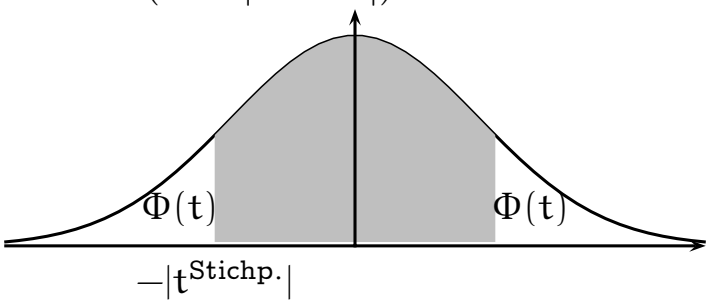
- Wenn X fast multikollinear ist, dann ist $(X'X)^{-1}$ sehr groß und die Schätzung für $\hat{\beta}$ sehr ungenau.

Um die Hypothese $H_0 : \beta_j = \beta_{j,0}$ versus $H_1 : \beta_j \neq \beta_{j,0}$ zu testen:

- bestimme die t Statistik:

$$t = \frac{\hat{\beta}_j - \beta_{j,0}}{\hat{\sigma}_{\hat{\beta}_j}}$$

- Der p-Wert ist $p = \Pr\left(|t| > \left|t^{\text{Stichp.}}\right|\right) = 2\Phi(-|t^{\text{Stichp.}}|)$




```
est <- lm(testscr ~ str + elpct)
```

diag(X) beschreibt die Diagonalmatrix von X falls X eine Matrix ist. Für x einen Vektor x spannt diag(x) die Diagonalmatrix auf. coef extrahiert die geschätzten Koeffizienten aus einem Modell.

Homoskedastische Standardabweichung von $\hat{\beta}$

$$\Sigma_{\hat{\beta}\hat{\beta}} = \hat{\sigma}_u^2 (X'X)^{-1} \qquad \text{Homoskedastizität}$$

```
(stddevh <- sqrt(diag(vcov(est))))
```

```
(Intercept)         str       elpct
 7.41131248  0.38027832  0.03934255
```

```
coef(est)/stddevh
```

```
(Intercept)         str       elpct
  92.565554   -2.896026  -16.515879
```

```
round(2 * pnorm(-abs(coef(est))/stddevh), 5)
```

```
(Intercept)         str       elpct
    0.00000     0.00378     0.00000
```

```
summary(est)
```

```
Call:
lm(formula = testscr ~ str + elpct)

Residuals:
     Min       1Q   Median       3Q      Max
-48.8447 -10.2404  -0.3078   9.8153  43.4607

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 686.03225    7.41131  92.566  < 2e-16 ***
str          -1.10130    0.38028  -2.896  0.00398 **
elpct        -0.64978    0.03934 -16.516  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 14.46 on 417 degrees of freedom
Multiple R-squared: 0.4264,         Adjusted R-squared: 0.4237
F-statistic:   155 on 2 and 417 DF,  p-value: < 2.2e-16
```

Heteroskedastische Varianz-Kovarianz Matrix von $\hat{\boldsymbol{\beta}}$

$$\Sigma_{\hat{\beta}\hat{\beta}} = (X'X)^{-1}X'\mathbf{I}\mathbf{u}^2X(X'X)^{-1} \qquad \text{Heteroskedastizität (immer richtig)}$$

Nun machen wir die gleichen Schritte mit Heteroskedastie-robusten Standardfehlern:

```
(stddev <- sqrt(diag(hccm(est))))
```

```
(Intercept)         str       elpct
 8.81224085  0.43706612  0.03129693
```

```
coef(est)/stddev
```

```
(Intercept)         str       elpct
  77.849920   -2.519747  -20.761681
```

```
round(2 * pnorm(-abs(coef(est))/stddev), 5)
```

```
(Intercept)         str       elpct
    0.00000     0.01174     0.00000
```

4.8.3 Exkurs: Multiplikation:

Inneres Produkt von Vektoren mit %*%

$$(a_0, a_1, a_2, \cdots, a_k) \ \%*\% + \begin{pmatrix} b_0 \\ b_1 \\ b_2 \\ \vdots \\ b_k \end{pmatrix} = \sum_{i=0}^{k} a_i b_i$$

Elementweises Produkt $*$

$$\begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ \vdots \\ a_k \end{pmatrix} * \begin{pmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ b_k \end{pmatrix} = \begin{pmatrix} a_1 \cdot b_1 \\ a_2 \cdot b_2 \\ a_3 \cdot b_3 \\ \vdots \\ a_k \cdot b_k \end{pmatrix}$$

äußeres Produkt A%o%B

$$\begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ \vdots \\ a_k \end{pmatrix} \%o\% (b_0, b_1, b_2, \cdots, b_m) = \begin{pmatrix} a_1 \cdot b_0 & a_1 \cdot b_1 & a_1 \cdot b_2 & \cdots & a_1 \cdot b_m \\ a_2 \cdot b_0 & a_2 \cdot b_1 & a_2 \cdot b_2 & \cdots & a_2 \cdot b_m \\ a_3 \cdot b_0 & a_3 \cdot b_1 & a_3 \cdot b_2 & \cdots & a_3 \cdot b_m \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_k \cdot b_0 & a_k \cdot b_1 & a_k \cdot b_2 & \cdots & a_k \cdot b_m \end{pmatrix}$$

$$\begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ \vdots \\ a_k \end{pmatrix} \%o\% (+1, -1) = \begin{pmatrix} a_1 & -a_1 \\ a_2 & -a_2 \\ a_3 & -a_3 \\ \vdots & \vdots \\ a_k & -a_k \end{pmatrix}$$

Konfidenzintervall für $\hat{\beta}$

```
qnorm(0.975)
```

```
[1] 1.959964
```

```
coef(est) + qnorm(0.975) * stddev %o% c(-1, 1)
```

```
                  [,1]        [,2]
(Intercept) 668.7605740 703.3039234
str          -1.9579298  -0.2446620
elpct        -0.7111176  -0.5884359
```

Was passiert, wenn die Klassengröße z.B. um 2 verkleinert wird?

```
-2 * (coef(est) + qnorm(0.975) * stddev %o% c(-1, 1))["str",
    ]
```

```
[1] 3.9158595 0.4893241
```

4.8.4 Erweiterung der Schätzgleichung um Ausgaben pro Schüler

```
(est <- lm(testscr ~ str + elpct))
```

```
Call:
lm(formula = testscr ~ str + elpct)
```

```
Coefficients:
(Intercept)          str         elpct
   686.0322      -1.1013      -0.6498
```

```
(est <- lm(testscr ~ str + elpct + expnstu))
```

```
Call:
lm(formula = testscr ~ str + elpct + expnstu)

Coefficients:
(Intercept)          str         elpct       expnstu
 649.577947    -0.286399    -0.656023     0.003868
```

```
(stddev <- sqrt(diag(hccm(est))))
```

```
 (Intercept)          str         elpct       expnstu
15.668622170  0.487512918  0.032114291  0.001607407
```

```
coef(est)
```

```
  (Intercept)           str         elpct       expnstu
649.577947257  -0.286399240  -0.656022660   0.003867902
```

```
coef(est)/stddev
```

```
(Intercept)          str        elpct      expnstu
  41.457247    -0.587470   -20.427749     2.406299
```

```
round(2 * pnorm(-abs(coef(est)/stddev)), 5)
```

```
(Intercept)          str        elpct      expnstu
    0.00000      0.55689      0.00000      0.01612
```

Vergleiche die Standardabweichung des Koeffizienten von `str` in den verschiedenen Schätzgleichungen:

```
sqrt(diag(hccm(lm(testscr ~ str))))["str"]
```

```
     str
0.5243585
```

```
sqrt(diag(hccm(lm(testscr ~ str + elpct))))["str"]
```

```
     str
0.4370661
```

```
sqrt(diag(hccm(lm(testscr ~ str + elpct + expnstu))))["str"]
```

```
     str
0.4875129
```

4.9 Verbundene Hypothesen

z.B. $\beta_{\texttt{str}} = 0$ und $\beta_{\texttt{expnstu}} = 0$, oder $\beta_1 = 0$ und $\beta_2 = 0$

Formal: $H_0 : \beta_1 = \beta_{1,0} \wedge \beta_2 = \beta_{2,0}$

versus

$H_1 : \beta_1 \neq \beta_{1,0} \vee \beta_2 \neq \beta_{2,0}$

Idee: man könnte einfach $\beta_1 = 0$ und $\beta_2 = 0$ getrennt testen.

$$t_1 = \frac{\hat{\beta}_1 - \beta_{1,0}}{\hat{\sigma}_{\hat{\beta}_1}} \qquad t_2 = \frac{\hat{\beta}_2 - \beta_{2,0}}{\hat{\sigma}_{\hat{\beta}_2}}$$

Die Nullhypothese $H_0 : \beta_1 = \beta_{j1,0} \wedge \beta_1 = \beta_{j1,0}$ würde dann verworfen, wenn entweder $\beta_1 = 0$ oder $\beta_2 = 0$ verworfen wird.

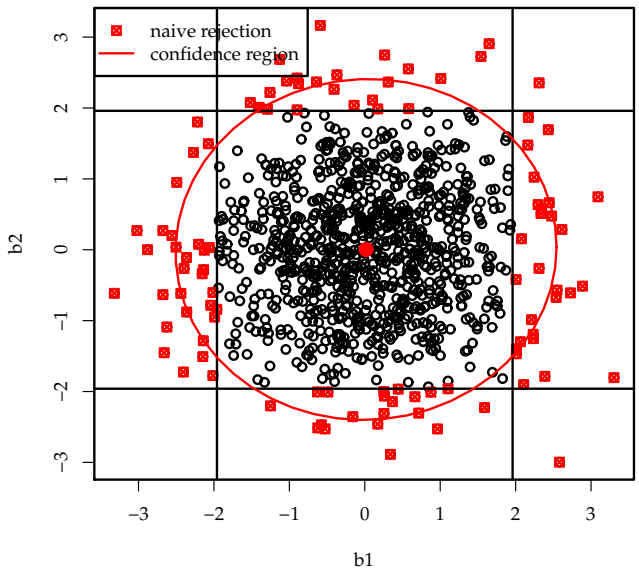
Man sieht leicht, dass dies schon bei unkorrelierten βs fehlschlägt:

```
set.seed(100)
N <- 1000
p <- 0.05
qcrit <- -qnorm(p/2)
b1 <- rnorm(N)
b2 <- rnorm(N)
reject <- abs(b1) > qcrit | abs(b2) > qcrit
mean(reject) * 100
```

```
[1] 10.3
```

Es werden nicht 5% der Werte zurückgewiesen, sondern im Beispiel 10.3 %. Das ist kein Zufall. In der folgenden Graphik sehen wir, dass wir sowohl links und rechts 5% abschneiden, als auch (ein weiteres Mal) oben und unten.

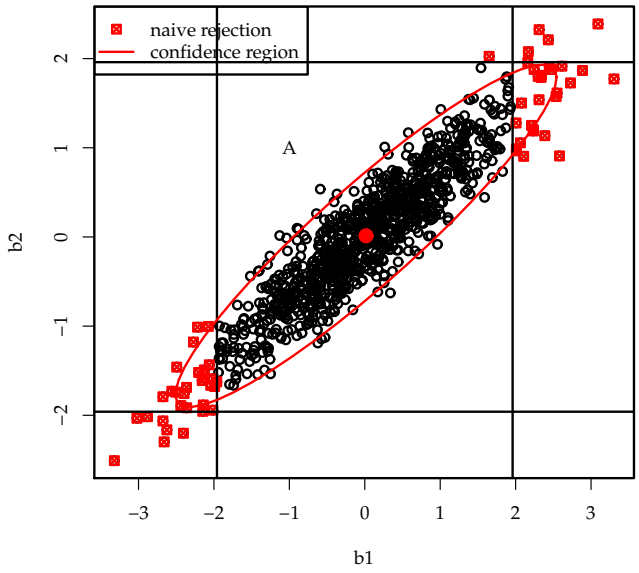
```
plot(b2 ~ b1)
points(b2 ~ b1, subset = reject, col = "red", pch = 7)
abline(v = c(qcrit, -qcrit), h = c(qcrit, -qcrit))
data.ellipse(b1, b2, levels = 1 - p, plot.points = FALSE)
legend("topleft", c("naive rejection", "confidence region"),
    pch = c(7, NA), col = "red", lty = c(NA, 1))
```



Wir sehen außerdem, dass das naive Verfahren nur auf die maximale Abweichung der Variablen achtet. Sinnvoller wäre es, alle Beobachtungen außerhalb des roten Kreises auszuschließen.
Noch lästiger wird das zweite Problem, wenn die Zufallsvariablen korreliert sind:

```
set.seed(100)
b1 <- rnorm(N)
b2 <- 0.3 * rnorm(N) + 0.7 * b1
reject <- abs(b1) > qcrit | abs(b2) > qcrit
```

```
plot(b2 ~ b1)
points(b2 ~ b1, subset = reject, col = "red", pch = 7)
abline(v = c(qcrit, -qcrit), h = c(qcrit, -qcrit))
data.ellipse(b1, b2, levels = 1 - p, plot.points = FALSE)
text(-1, 1, "A")
legend("topleft", c("naive rejection", "confidence region"),
    pch = c(7, NA), col = "red", lty = c(NA, 1))
```



Beispielsweise liegt der Punkt „A“ in der Graphik deutlich außerhalb des Konfindenzbereiches, obwohl keine der beiden Koordinaten „auffällig“ ist.

4.9.1 F Statistik für zwei Restriktionen

$$t_1 = \frac{\hat{\beta}_1 - \beta_{1,0}}{\hat{\sigma}_{\hat{\beta}_1}} \qquad t_2 = \frac{\hat{\beta}_2 - \beta_{2,0}}{\hat{\sigma}_{\hat{\beta}_2}}$$

$$F = \frac{1}{2} \cdot \frac{t_1^2 + t_2^2 - 2\hat{\rho}_{t_1 t_2} \cdot t_1 \cdot t_2}{1 - \hat{\rho}_{t_1 t_2}^2}$$

wobei $\hat{\rho}_{t_1 t_2}$ die geschätzte Korrelation zwischen t_1 und t_2 ist.
Falls $\hat{\rho}_{t_1 t_2} = 0$:

$$F = \frac{1}{2}\left(t_1^2 + t_2^2\right)$$

zur Erinnerung:

$$\frac{N(0,1)}{\sqrt{\chi^2_n/n}} \sim t_n \qquad \sum_{i=1}^{n} (N(0,1))^2 \sim \chi^2_n \qquad \frac{\chi^2_{n_1}/n_1}{\chi^2_{n_2}/n_2} \sim F_{n_1,n_2}$$

4.9.2 Mehr als zwei Restriktionen

Schreibe Restriktionen als

$$R\boldsymbol{\beta} = \mathbf{r}$$

z.B.

$$(0,1,0,\cdots,0)\begin{pmatrix}\beta_0\\ \beta_1\\ \beta_2\\ \vdots\\ \beta_k\end{pmatrix} = 0$$

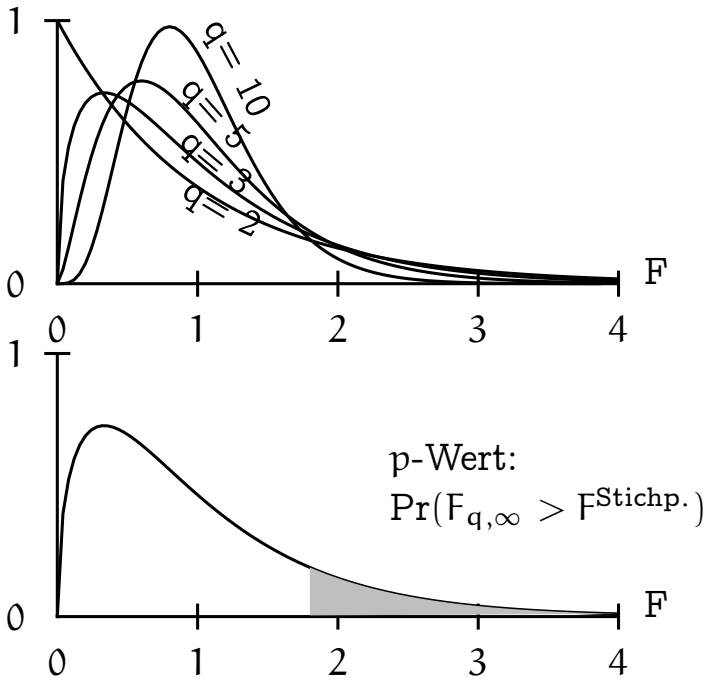
$$\begin{pmatrix}0&1&0&\cdots&0\\ 0&0&1&\cdots&0\end{pmatrix}\begin{pmatrix}\beta_0\\ \beta_1\\ \beta_2\\ \vdots\\ \beta_k\end{pmatrix} = \begin{pmatrix}0\\ 0\end{pmatrix}$$

$$\begin{pmatrix}1&0&-1&\cdots&0\\ 0&1&0&\cdots&0\\ 0&1&2&\cdots&0\end{pmatrix}\begin{pmatrix}\beta_0\\ \beta_1\\ \beta_2\\ \vdots\\ \beta_k\end{pmatrix} = \begin{pmatrix}0\\ 7\\ 0\end{pmatrix}$$

$$F = \frac{1}{q}(R\widehat{\boldsymbol{\beta}} - \mathbf{r})' \left(R\hat{\Sigma}_{\hat{\beta}\hat{\beta}}R'\right)^{-1} (R\widehat{\boldsymbol{\beta}} - \mathbf{r})$$

wobei q die Anzahl der Restriktionen ist.
Falls Annahmen 1–4 gelten:

$$F \xrightarrow{p} F_{q,\infty}$$



4.9.3 Spezialfälle:

Restriktionen für „die“ F-Statistik einer Schätzung (β_0 wird nicht getestet)

$$\begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{pmatrix} \begin{pmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \\ \vdots \\ \beta_k \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

Test eines einzelnen Koeffizienten:

$$\begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \end{pmatrix} \begin{pmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_k \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

$$t_1 = \frac{\hat{\beta}_1 - \beta_{1,0}}{\hat{\sigma}_{\hat{\beta}_1}} \qquad X \sim t_{(k)} \Leftrightarrow X^2 \sim F_{(k,k)}$$

`c` verbindet die Argumente zu einem Vektor. `rbind` setzt die Argumente der Funktion (Vektoren, Matrizen) zeilenweise zusammen. `cbind` setzt die Argumente der Funktion (Vektoren, Matrizen) spaltenweise zusammen. `linear.hypothesis` testet lineare Hypothesen. `pf` berechnet die Verteilungsfunktion der F-Verteilung, `df` die Dichtefunktion, `qf` die Verteilungsquantile, `rf` eine F-verteilte Zufallsvariable.

```
est <- lm(testscr ~ str + elpct + expnstu)
R <- rbind(c(0, 1, 0, 0), c(0, 0, 0, 1))
r <- c(0, 0)
linear.hypothesis(est, R, r)
```

```
Linear hypothesis test

Hypothesis:
str = 0
expnstu = 0

Model 1: testscr ~ str + elpct + expnstu
Model 2: restricted model

  Res.Df   RSS  Df Sum of Sq      F   Pr(>F)
1    416 85700
2    418 89000  -2     -3300 8.0101 0.000386 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
testh = linear.hypothesis(est, R, r)
pf(testh$F, 2, Inf, lower.tail = FALSE)
```

```
[1]           NA 0.0003320828
```

```
linear.hypothesis(est, R, r, vcov = hccm)
```

```
Linear hypothesis test
```

```
Hypothesis:
str = 0
expnstu = 0

Model 1: testscr ~ str + elpct + expnstu
Model 2: restricted model

Note: Coefficient covariance matrix supplied.

  Res.Df  Df      F   Pr(>F)
1    416
2    418  -2 5.2617 0.005537 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
test = linear.hypothesis(est, R, r, vcov = hccm)
pf(test$F[2], 2, Inf, lower.tail = FALSE)
```

```
[1] 0.005186642
```

```
linear.hypothesis(est, c("str=0", "expnstu=0"), vcov = hccm)
```

```
Linear hypothesis test

Hypothesis:
str = 0
expnstu = 0

Model 1: testscr ~ str + elpct + expnstu
Model 2: restricted model

Note: Coefficient covariance matrix supplied.

  Res.Df  Df      F   Pr(>F)
1    416
2    418  -2 5.2617 0.005537 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

4.9.4 Spezialfall: Homoskedastische Störterme

zur Erinnerung: bei Heteroskedastizität:

$$F = \frac{1}{q}(R\hat{\boldsymbol{\beta}} - \mathbf{r})' \left(R\hat{\Sigma}_{\hat{\beta}\hat{\beta}}R'\right)^{-1} (R\hat{\boldsymbol{\beta}} - \mathbf{r})$$

bei Homoskedastizität, wenn wir $\beta_1 = \beta_2 = \ldots = \beta_k = 0$ testen:

$$F = \frac{n-k-1}{q} \cdot \frac{SSR_{\text{restricted}} - SSR_{\text{unrestricted}}}{SSR_{\text{unrestricted}}}$$

$SSR_{\text{restricted}}$	$\sum_{i=1}^{n} \hat{u}_i^2$ des „restricted“ Modells
$SSR_{\text{unrestricted}}$	$\sum_{i=1}^{n} \hat{u}_i^2$ des „unrestricted“ Modells
k	Anzahl Regressoren des „unrestricted“ Modells
q	Anzahl Restriktionen

wir erinnern uns:

$$R^2 = \frac{s_{\hat{y}}^2}{s_y^2} = 1 - \frac{s_{\hat{u}}^2}{s_y^2} = 1 - \frac{SSR}{TSS}$$

teile Zähler und Nenner von F durch TSS:

$$F = \frac{n-k-1}{q} \cdot \frac{R^2_{\text{unrestricted}} - R^2_{\text{restricted}}}{1 - R^2_{\text{unrestricted}}}$$

F ist verteilt ensprechend $F_{q,n-k-1}$

4.10 Restriktionen mit mehreren Koeffizienten

z.B. in `RetSchool` schätzen wir

```
wage76 ~ grade76 + age76  + black + daded + momed
```

Hypothese: $\beta_{\texttt{daded}} = \beta_{\texttt{momed}}$
1. Lösungsweg (F-test): $R = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & -1 \end{pmatrix}$, $r = 0$
2. Lösungsweg (t-test, Gleichung umformen): Allgemein:

$$\begin{aligned} y &= \beta_0 + \beta_1 X_1 + \beta_2 X_2 + u \\ &= \beta_0 + (\beta_1 - \underbrace{\beta_2}) X_1 + \beta_2 (X_2 + \underbrace{X_1}) + u \end{aligned}$$

```
data(RetSchool, package = "Ecdat")
attach(RetSchool)
summary(wage76)
```

```
    Min.  1st Qu.   Median     Mean  3rd Qu.     Max.     NA's
   0.000    1.377    1.683    1.658    1.957    3.180 2147.000
```

```
table(grade76)
```

```
grade76
   0    1    2    3    4    5    6    7    8    9   10   11   12   13
   3    2    2    4    6   13   22   42   90   92  148  194 1213  332
  14   15   16   17   18
 314  209  539  182  264
```

```
est <- lm(wage76 ~ grade76 + age76 + black + daded +
    momed)
linear.hypothesis(est, c("daded=momed"), vcov = hccm)
```

```
Linear hypothesis test

Hypothesis:
daded - momed = 0

Model 1: wage76 ~ grade76 + age76 + black + daded + momed
Model 2: restricted model

Note: Coefficient covariance matrix supplied.

  Res.Df   Df      F Pr(>F)
1   3053
2   3054   -1 1.9809 0.1594
```

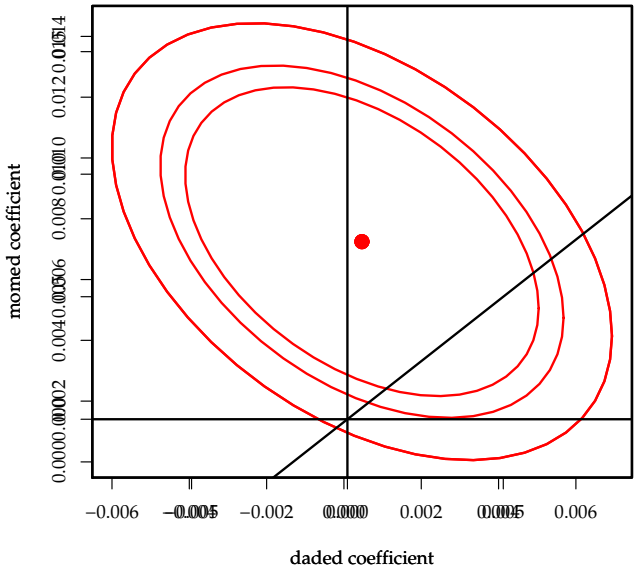
alternativ:

```
mombettered <- momed - daded
momdaded <- momed + daded
est2 <- lm(wage76 ~ grade76 + age76 + black + momdaded +
    momed)
linear.hypothesis(est2, "momed=0", vcov = hccm)
```

```
Linear hypothesis test

Hypothesis:
momed = 0

Model 1: wage76 ~ grade76 + age76 + black + momdaded + momed
Model 2: restricted model

Note: Coefficient covariance matrix supplied.

  Res.Df   Df      F Pr(>F)
1   3053
2   3054   -1 1.9809 0.1594
```

`confidence.ellipse` zeichnet den Konfizenzbereich für Koeffizienten eines linearen Modells.

```
confidence.ellipse(est, c("daded", "momed"))
confidence.ellipse(est, c("daded", "momed"), levels = c(0.9,
    0.95, 0.99))
abline(v = 0)
abline(h = 0)
abline(a = 0, b = 1)
```




```
linear.hypothesis(est, c("daded=0", "momed=0"), vcov = hccm)
```

```
Linear hypothesis test

Hypothesis:
daded = 0
momed = 0

Model 1: wage76 ~ grade76 + age76 + black + daded + momed
Model 2: restricted model

Note: Coefficient covariance matrix supplied.

  Res.Df   Df      F  Pr(>F)
1   3053
2   3055   -2 3.6955 0.02495 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
linear.hypothesis(est, c("daded=0", "momed=0.01"), vcov = hccm)
```

```
Linear hypothesis test
```

```
Hypothesis:
daded = 0
momed = 0.01

Model 1: wage76 ~ grade76 + age76 + black + daded + momed
Model 2: restricted model

Note: Coefficient covariance matrix supplied.

  Res.Df   Df      F Pr(>F)
1   3053
2   3055   -2 0.4433  0.642
```

```
linear.hypothesis(est, c("daded=0.01", "momed=0"), vcov = hccm)
```

```
Linear hypothesis test

Hypothesis:
daded = 0.01
momed = 0

Model 1: wage76 ~ grade76 + age76 + black + daded + momed
Model 2: restricted model

Note: Coefficient covariance matrix supplied.

  Res.Df   Df      F   Pr(>F)
1   3053
2   3055   -2 6.6741 0.001282 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
detach(RetSchool)
```

4.11 Modellspezifikation

- `testscr ~ distcod + county + district + grspan + enrltot + teachers + calwpct + mealpct + computer + compstu + expnstu + str + avginc + elpct + readscr + mathscr`

- `testscr ~ str`
- Omitted variable bias

$$E(\mathbf{b}_1) = \boldsymbol{\beta}_1 + (X_1'X_1)^{-1}X_1'X_2\boldsymbol{\beta}_2$$

- Overfitting (Multicollinearity)

$$\Sigma_{\hat{\beta}\hat{\beta}} = (X'X)^{-1}X'\mathbf{I}\mathbf{u}^2X(X'X)^{-1}$$

→ beginne mit eine „base specification“

→ darauf aufbauend werden „alternative specifications“ entwickelt

wenn sich in einer „alternative specification“ Koeffizienten ändern, kann das ein Hinweis auf omitted variable bias sein

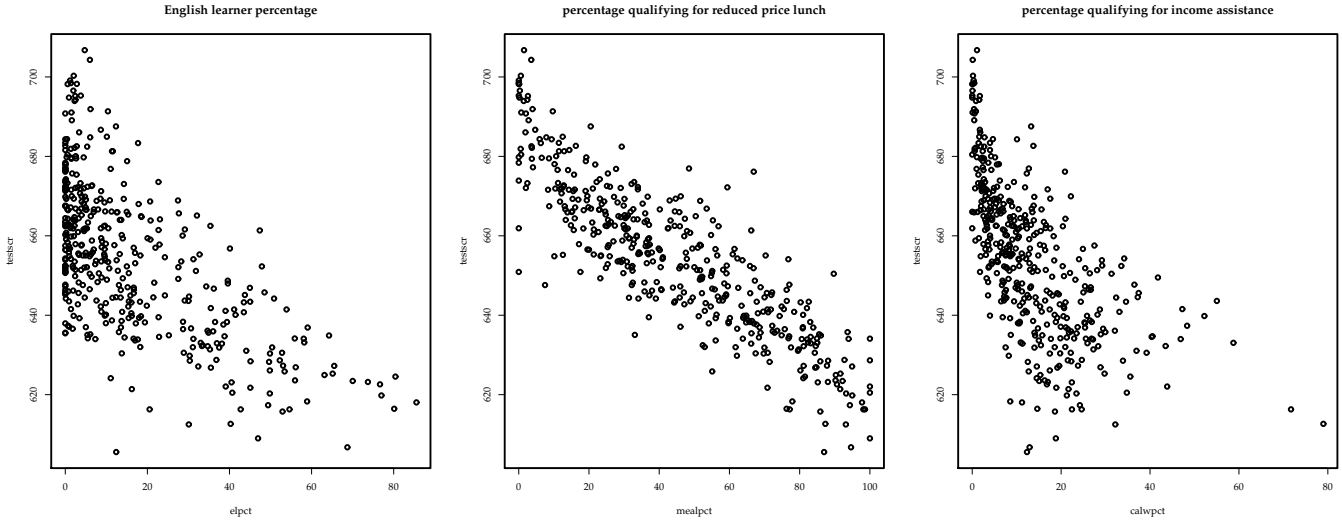
- Wie wird eine Variable in die Regression eingebracht — Skalierung

```
est <- lm(testscr ~ str + elpct + expnstu)
elratio <- elpct/100
est <- lm(testscr ~ str + elratio + expnstu)
expnstuTSD <- expnstu/1000
est <- lm(testscr ~ str + elpct + expnstuTSD)
```

Was bringt eine weitere Variable

- messe R^2
- messe Beitrag zum R^2
- betrachte p-Wert der t-Statistik
- betrachte p-Wert der Varianzanalyse

```
par(mfrow = c(1, 3))
plot(testscr ~ elpct, main = "English learner percentage")
plot(testscr ~ mealpct, main = "percentage qualifying for reduced price lunch")
plot(testscr ~ calwpct, main = "percentage qualifying for income assistance")
```



4.11.1 Messe R^2

$$R^2 = 1 - \frac{SSR}{TSS}$$

- R^2 misst nur den „fit“ der Regression
- R^2 misst keine Kausalität (z.B. Parkplatz $\rightarrow$ testscr)
- R^2 misst nicht die Abwesenheit von Omitted variable bias
- R^2 misst nicht die Korrektheit der Spezifikation

4.11.2 Messe Beitrag zum R^2

Hier gibt es sehr unterschiedliche Methoden.
Man kann sich das R^2 ansehen, wenn die jeweiligen Variable die einzigen (und erste, first) im Modell ist, oder die letzte (last), oder alle denkbaren Reihenfolgen (lmg und pmvd). pmvd bietet eine Gewichtung die unter anderem dafür sorgt, dass der Beitrag immer positiv bleibt.

```
library(relaimpo)
est <- lm(testscr ~ str + elpct + mealpct + calwpct)
calc.relimp(est, type = "first", rela = TRUE)
```

```
Response variable: testscr
Total response variance: 363.0301
```

```
Analysis based on 420 observations

4 Regressors:
str elpct mealpct calwpct
Proportion of variance explained by model: 77.49%
Metrics are normalized to sum to 100% (rela=TRUE).

Relative importance metrics:

             first
str     0.03175031
elpct   0.25708495
mealpct 0.46768098
calwpct 0.24348376
```

```
calc.relimp(est, type = "last", rela = TRUE)
```

```
Response variable: testscr
Total response variance: 363.0301
Analysis based on 420 observations

4 Regressors:
str elpct mealpct calwpct
Proportion of variance explained by model: 77.49%
Metrics are normalized to sum to 100% (rela=TRUE).

Relative importance metrics:

              last
str     0.059126952
elpct   0.048159854
mealpct 0.890678586
calwpct 0.002034608
```

```
calc.relimp(est, type = "lmg", rela = TRUE)
```

```
Response variable: testscr
Total response variance: 363.0301
Analysis based on 420 observations
```

```
4 Regressors:
str elpct mealpct calwpct
Proportion of variance explained by model: 77.49%
Metrics are normalized to sum to 100% (rela=TRUE).

Relative importance metrics:

               lmg
str     0.03119231
elpct   0.22371548
mealpct 0.53343971
calwpct 0.21165250

Average coefficients for different model sizes:

                1X        2Xs        3Xs         4Xs
str     -2.2798083 -1.4612232 -1.1371224 -1.01435328
elpct   -0.6711562 -0.4347537 -0.2510901 -0.12982189
mealpct -0.6102858 -0.5922408 -0.5645062 -0.52861908
calwpct -1.0426750 -0.5863541 -0.2639020 -0.04785371
```

```
calc.relimp(est, type = "pmvd", rela = TRUE)
```

```
Response variable: testscr
Total response variance: 363.0301
Analysis based on 420 observations

4 Regressors:
str elpct mealpct calwpct
Proportion of variance explained by model: 77.49%
Metrics are normalized to sum to 100% (rela=TRUE).

Relative importance metrics:

                pmvd
str     0.0148176134
elpct   0.0242703918
mealpct 0.9600101671
calwpct 0.0009018276
```

```
Average coefficients for different model sizes:

                1X        2Xs        3Xs         4Xs
str     -2.2798083 -1.4612232 -1.1371224 -1.01435328
elpct   -0.6711562 -0.4347537 -0.2510901 -0.12982189
mealpct -0.6102858 -0.5922408 -0.5645062 -0.52861908
calwpct -1.0426750 -0.5863541 -0.2639020 -0.04785371
```

4.11.3 Informationskriterien

Anstatt auf den p-Wert eines Koeffizienten zu schauen, vergleichen wir die Varianz der Residuen im Modelles ohne diesen Koeffizienten mit der Varianz der Residuen im Modell mit diesem Koeffizienten.

$$F_{(k_2-k_1,n-k_2)} = \frac{RSS_1 - RSS_2}{RSS_2}\frac{n-k_2}{k_2-k_1}$$

Sei L die log-Likelihood des geschätzten Modells.
Man kann zeigen, dass für lineare Modelle gilt

$$-2 \cdot L = n \cdot \log\frac{RSS}{n} + C$$

Dann ist aber

$$2 \cdot (L_2 - L_1) = n \cdot \left(\log\frac{RSS_2}{n} - \log\frac{RSS_1}{n}\right) = n \log\frac{RSS_2}{RSS_1} \sim \chi^2_{k_2-k_1}$$

```
est2 <- lm(testscr ~ str + elpct + mealpct + calwpct)
est1 <- lm(testscr ~ str + mealpct + calwpct)
sum(est2$residuals^2)
```

```
[1] 34247.46
```

```
deviance(est2)
```

```
[1] 34247.46
```

```
RSS2 = deviance(est2)
RSS1 = deviance(est1)
L2 = logLik(est2)
L1 = logLik(est1)
n <- length(est$residuals)
k2 <- est2$rank
k1 <- est1$rank
pchisq(2 * (L2 - L1), k2 - k1, lower = FALSE)
```

```
'log Lik.' 0.0001398651 (df=6)
```

```
pchisq(n * log(RSS1/RSS2), k2 - k1, lower = FALSE)
```

```
[1] 0.0001398651
```

```
anova(est1, est2, test = "Chisq")
```

```
Analysis of Variance Table

Model 1: testscr ~ str + mealpct + calwpct
Model 2: testscr ~ str + elpct + mealpct + calwpct
  Res.Df   RSS  Df Sum of Sq P(>|Chi|)
1    416 35451
2    415 34247   1      1203 0.0001342
```

```
pf((RSS1 - RSS2)/RSS2 * (n - k2)/(k2 - k1), k2 - k1,
    n - k2, lower = FALSE)
```

```
[1] 0.0001547027
```

```
anova(est1, est2)
```

```
Analysis of Variance Table

Model 1: testscr ~ str + mealpct + calwpct
Model 2: testscr ~ str + elpct + mealpct + calwpct
  Res.Df   RSS  Df Sum of Sq      F    Pr(>F)
1    416 35451
```

```
2    415 34247   1      1203 14.582 0.0001547 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Ein ähnliches Verfahren benutzt Informationskriterien Ziel: Finde ein Modell, das die Daten gut erklärt, aber möglichst wenige Parameter hat.
Sei L die log-Likelihood des geschätzten Modells.
Hirotsugo Akaike (1971): An Information Criterion:

$$\mathrm{AIC} = -2 \cdot \mathrm{L} + 2 \cdot \mathrm{k}$$

Gideon E. Schwarz (1978): Bayesian Information Criterion

$$\mathrm{BIC} = -2 \cdot \mathrm{L} + \mathrm{k} \cdot \log \mathrm{n}$$

```
est <- lm(testscr ~ str + elpct + mealpct + calwpct +
    enrltot)
extractAIC(est)
```

```
[1]    6.000 1859.498
```

```
step(est)
```

```
Start:  AIC=1859.5
testscr ~ str + elpct + mealpct + calwpct + enrltot

          Df Sum of Sq   RSS   AIC
- calwpct  1        70 34239  1858
- enrltot  1        79 34247  1858
<none>                 34169  1859
- elpct    1      1263 35431  1873
- str      1      1553 35721  1876
- mealpct  1     20702 54871  2056

Step:  AIC=1858.36
testscr ~ str + elpct + mealpct + enrltot

          Df Sum of Sq   RSS   AIC
```

```
- enrltot  1        60 34298  1857
<none>                34239  1858
- elpct    1      1208 35446  1871
- str      1      1496 35734  1874
- mealpct  1     51150 85388  2240

Step:  AIC=1857.09
testscr ~ str + elpct + mealpct

          Df Sum of Sq   RSS   AIC
<none>                34298  1857
- elpct    1      1167 35465  1869
- str      1      1441 35740  1872
- mealpct  1     52947 87245  2247

Call:
lm(formula = testscr ~ str + elpct + mealpct)

Coefficients:
(Intercept)          str        elpct      mealpct
   700.1500      -0.9983      -0.1216      -0.5473
```

4.11.4 t-Statistik für individuelle Koeffizienten

$$t_i = \frac{\hat{\beta}_i - \beta_{i,0}}{\hat{\sigma}_{\hat{\beta}_i}}$$

```
est <- lm(testscr ~ str + elpct + mealpct + calwpct)
coef(est)/sqrt(diag(hccm(est)))
```

```
(Intercept)         str       elpct     mealpct     calwpct
124.7256881  -3.7184527  -3.5192705 -13.5610567  -0.7778974
```

```
round(2 * pnorm(-abs(coef(est)/sqrt(diag(hccm(est))))),
    5)
```

```
(Intercept)         str       elpct     mealpct     calwpct
    0.00000     0.00020     0.00043     0.00000     0.43663
```

Anstatt, wie oben, immer die heteroskedastie-robusten Standardfehler von Hand auszurechnen, können wir uns auch das Leben einfacher machen. Der Funktion `coeftest` kann man sagen, welche Varianz-Kovarianzmatrix verwendet werden soll. Eine kleine Funktion `lmr` schreiben wir selbst, damit wir für jede Regression gleich die heteroskedastie-robusten Standardfehler verwenden.

```
library(lmtest)
lmr <- function(...) {
    est <- lm(...)
    print(coeftest(est, vcov = hccm, df = NA))
    cat("R2=        ", round(summary(est)$r.squared,
        2), "\n")
    est
}
```

4.11.5 Vergleich von Modellen

Wenn wir eine Tabelle verschiedener Modelle sehen wollen, und dabei heteroskedastie-robusten Standardfehler verwenden wollen, passen wir die Funktion `getSummary.lm` etwas an. Wir verwenden im wesentlich die Version `getSummary.lm` aus dem Paket `memisc`, allerdings ersetzen wir den Standardfehler durch den heteroskedastie-robusten Standardfehler:

```
library(memisc)
getSummary.lm <- function(est, ...) {
    z <- memisc::getSummary.lm(est, ...)
    z$coef[, "se"] <- sqrt(diag(hccm(est)))
    z
}
```

Nun schätzen wir einige Modelle:

```
est1 <- lm(testscr ~ str)
est2 <- lm(testscr ~ str + elpct)
est3 <- lm(testscr ~ str + elpct + mealpct)
est4 <- lm(testscr ~ str + elpct + calwpct)
est5 <- lm(testscr ~ str + elpct + mealpct + calwpct)
```

```
mtable(`(1)` = est1, `(2)` = est2, `(3)` = est3, `(4)` = est4,
    `(5)` = est5, summary.stats = c("R-squared", "N"))
```

	(1)	(2)	(3)	(4)	(5)
(Intercept)	698.933***	686.032***	700.150***	697.999***	700.392***
	(10.461)	(8.812)	(5.641)	(7.006)	(5.615)
str	−2.280***	−1.101**	−0.998***	−1.308***	−1.014***
	(0.524)	(0.437)	(0.274)	(0.343)	(0.273)
elpct		−0.650***	−0.122***	−0.488***	−0.130***
		(0.031)	(0.033)	(0.030)	(0.037)
mealpct			−0.547***		−0.529***
			(0.024)		(0.039)
calwpct				−0.790***	−0.048
				(0.070)	(0.062)
R-squared	0.051	0.426	0.775	0.629	0.775
N	420	420	420	420	420

4.11.6 Diskussion

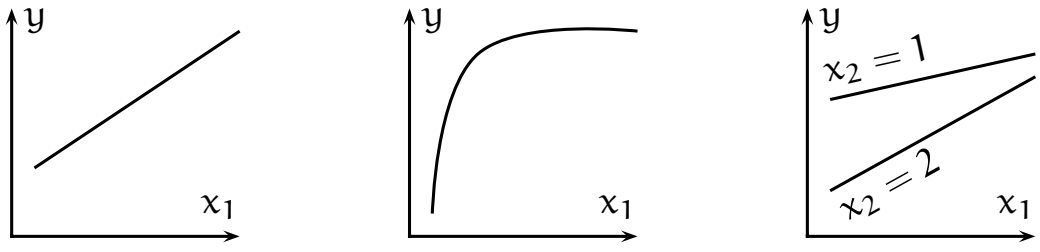
- Kontrolle für Schülercharakteristika halbiert den Koeffizienten von `str`
- Schülercharakteristika sind gute Prediktoren
- Das Vorzeichen der Koeffizienten der Schülercharakteristika stimmt mit den Bildern überein
- Kontrollvariablen sind nicht immer signifikant

  `calwpct` erscheint redundant in diesem Kontext

5 Nichtlineare Regressionsfunktionen

Bislang

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_k x_k$$



linear | nicht linear, keine Interaktion | Interaktion von x_1 und x_2

Wenn die Beziehung zwischen Y und X nicht linear ist...

- ist der marginale Effekt von X nicht konstant
- wäre eine lineare Regression falsch spezifiziert

→ der geschätzte Effekt wäre verzerrt

→ deshalb schätzen wir eine nichtlineare Regression in X

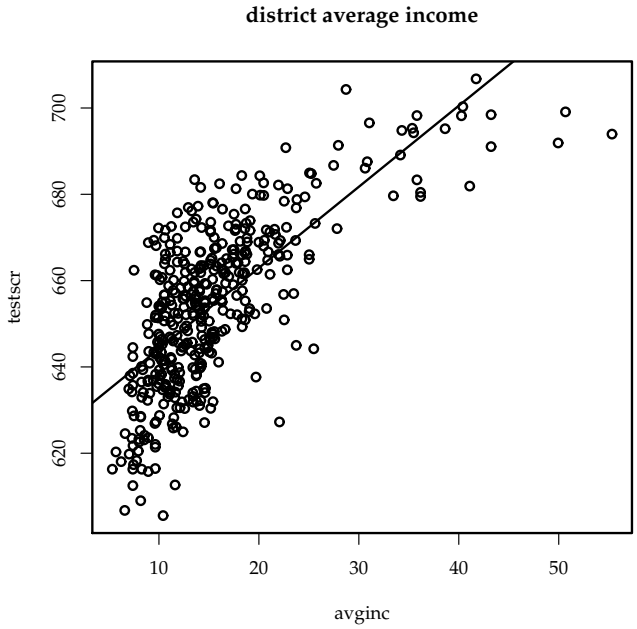
Vorgehensweise:

- nichtlinearen Funktionen einer einzelnen unabhängigen Variablen
  - Polynome in X
  - Logarithmische Transformationen
- Interaktionen

```
data(Caschool)
attach(Caschool)
plot(testscr ~ avginc, main = "district average income")
est1 <- lm(testscr ~ avginc)
summary(est1)
```

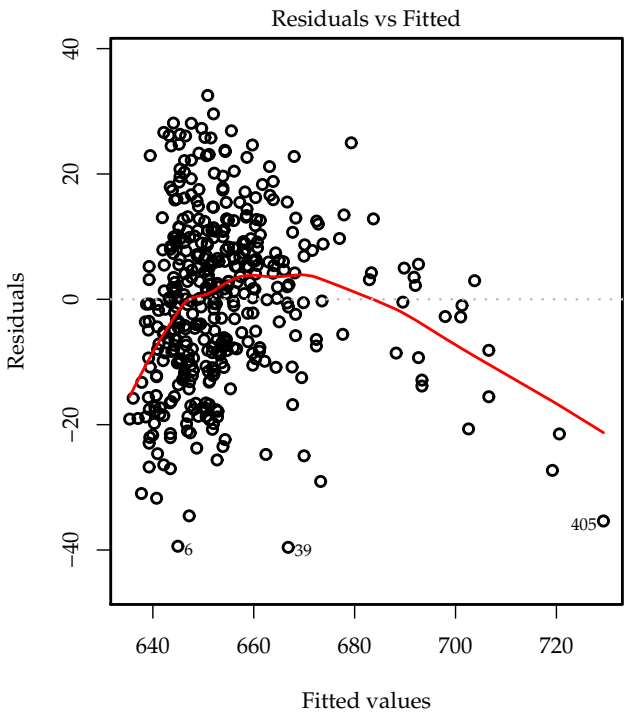
```
Call:
lm(formula = testscr ~ avginc)

Residuals:
     Min       1Q   Median       3Q      Max
-39.5744  -8.8032   0.6028   9.0318  32.5302

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 625.3836     1.5324  408.11   <2e-16 ***
avginc        1.8785     0.0905   20.76   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 13.39 on 418 degrees of freedom
Multiple R-squared: 0.5076,        Adjusted R-squared: 0.5064
F-statistic: 430.8 on 1 and 418 DF,  p-value: < 2.2e-16
```

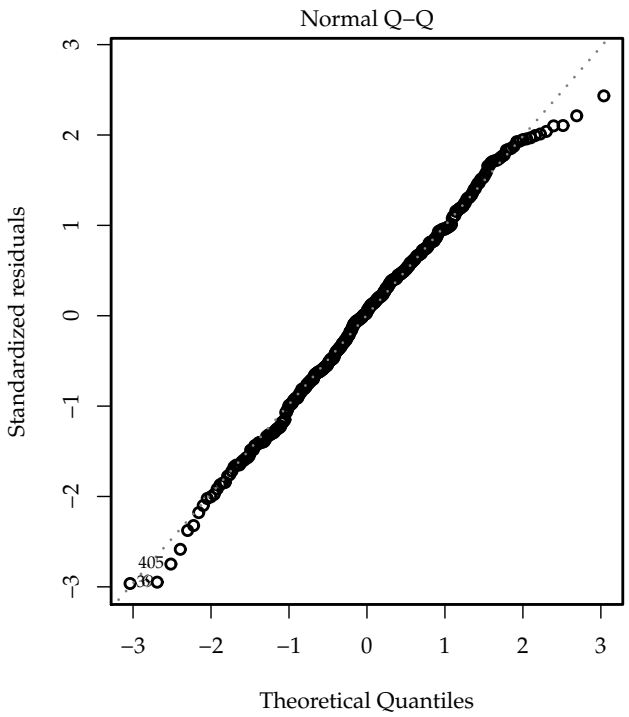
```
abline(est1)
```



Der diagnostische Plot bestätigt, dass in diesem linearen Modell die Residuen nicht unabhängig von `avginc` sind.

```
par(mfrow = c(1, 2))
plot(est1, which = 1:2)
```





Betrachte nun ein quadratisches Modell

> `order` berechnet einen Permutationsvektor den man zum Sortieren verwenden kann. Wenn man nur einen Vektor sortieren möchte, dann hilft auch `sort` das das Sortieren gleich mit übernimmt. `fitted` berechnet zu einer Regression das $\hat{y}$.

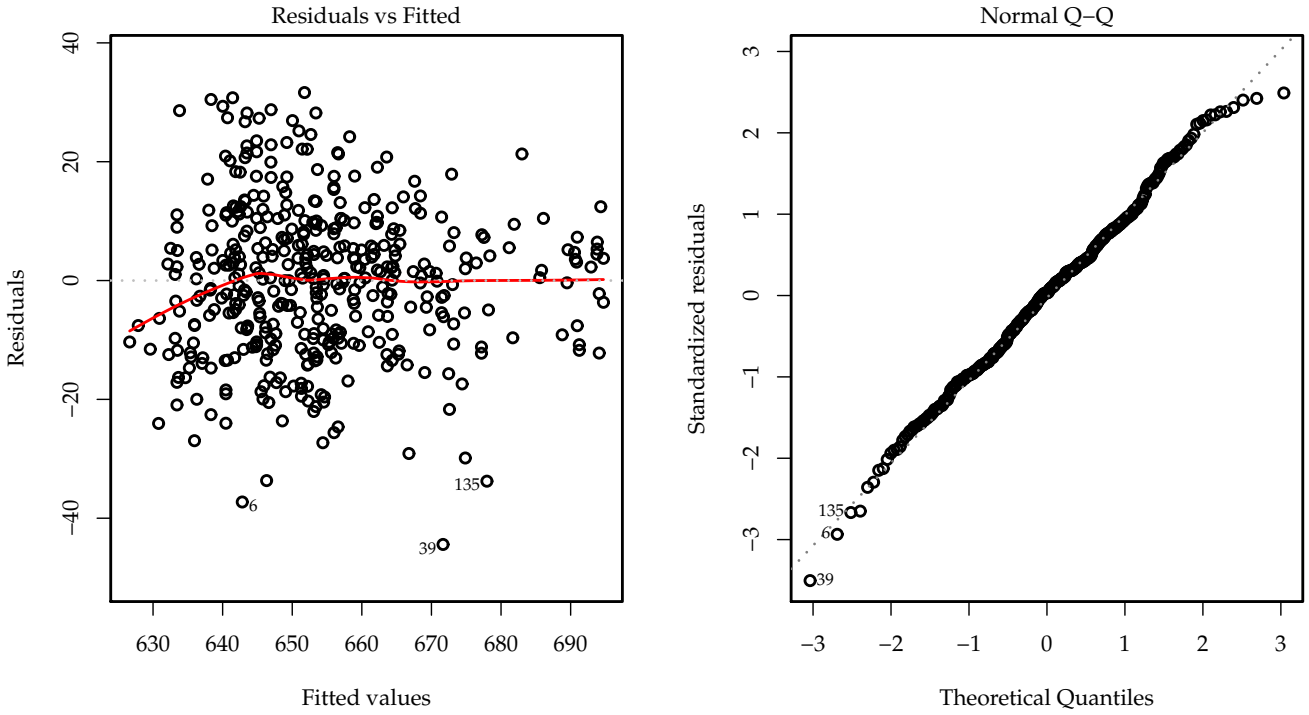
```
avginc2 <- avginc * avginc
est2 <- lm(testscr ~ avginc + avginc2)
summary(est2)
```

```
Call:
lm(formula = testscr ~ avginc + avginc2)

Residuals:
     Min       1Q   Median       3Q      Max
-44.4155  -9.0481   0.4399   8.3475  31.6393

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 607.30174    3.04622 199.362  < 2e-16 ***
```

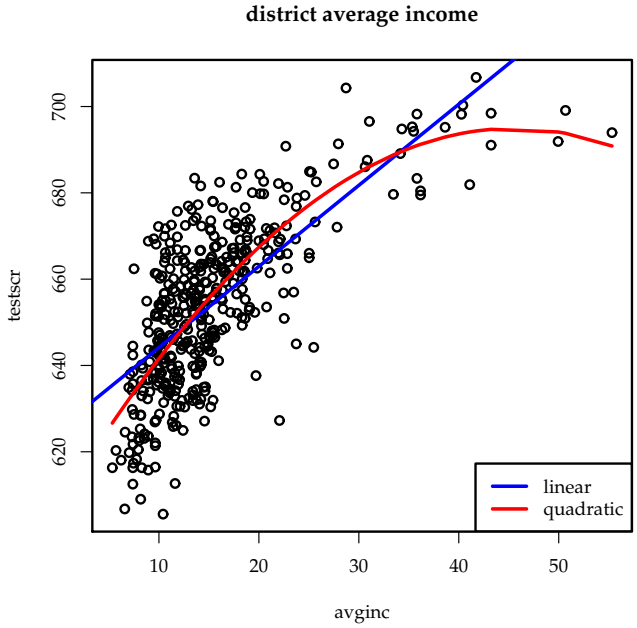
```
avginc         3.85100    0.30426  12.657  < 2e-16 ***
avginc2       -0.04231    0.00626  -6.758 4.71e-11 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 12.72 on 417 degrees of freedom
Multiple R-squared: 0.5562,         Adjusted R-squared: 0.554
F-statistic: 261.3 on 2 and 417 DF,  p-value: < 2.2e-16
```

Betrachten wir auch hier wieder den diagnostischen Plot:

```
par(mfrow = c(1, 2))
plot(est2, which = 1:2)
```




```
or <- order(avginc)
plot(testscr ~ avginc, main = "district average income")
abline(est1, col = "blue", lwd = 3)
lines(avginc[or], fitted(est2)[or], col = "red", lwd = 3)
legend("bottomright", c("linear", "quadratic"), lwd = 3,
    col = c("blue", "red"))
```



- Der Koeffizient von avginc2 ist signifikant von Null verschieden
- R^2 ist gestiegen

$$\texttt{testscr} = 607.30174 + 3.85100 \cdot \texttt{avginc} - 0.04231 \cdot \texttt{avginc}^2$$

Marginaler Effekt einer Änderung von avginc?

```
coef(est2)["avginc"] + 2 * 10 * coef(est2)["avginc2"]
```

```
  avginc
3.004826
```

```
coef(est2) %*% c(0, 1, 2 * 10)
```

```
         [,1]
[1,] 3.004826
```

```
coef(est2) %*% c(0, 1, 2 * 40)
```

```
          [,1]
[1,] 0.4663179
```

```
coef(est2) %*% c(0, 1, 2 * 60)
```

```
          [,1]
[1,] -1.226021
```

Konfidenzintervall für den Marginalen Effekt:

- Bei einem linearen Modell das Konfidenzintervall von β_i
- Im Beispiel das Konfidenzintervall von $\Delta Y = \beta_1 + 2 \cdot \beta_2 x$

  Wir wissen, $\frac{|\widehat{\Delta Y}^2|}{\sigma^2_{\widehat{\Delta Y}}} \sim F_{1,\infty}$, also $\sigma_{\widehat{\Delta Y}} = \frac{|\widehat{\Delta Y}|}{\sqrt{F}}$, also
  $\left[\widehat{\Delta Y} - \frac{|\widehat{\Delta Y}| \cdot 1.96}{\sqrt{F}}, \widehat{\Delta Y} + \frac{|\widehat{\Delta Y}| \cdot 1.96}{\sqrt{F}}\right]$

```
(lhtest <- linear.hypothesis(est2, "avginc + 20 * avginc2",
    vcov = hccm))
```

```
Linear hypothesis test

Hypothesis:
avginc + 20 avginc2 = 0

Model 1: testscr ~ avginc + avginc2
Model 2: restricted model

Note: Coefficient covariance matrix supplied.

  Res.Df  Df      F    Pr(>F)
1    417
2    418  -1 288.38 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
sqrt(lhtest$F)
```

```
[1]       NA 16.98182
```

```
coef(est2) %*% c(0, 1, 2 * 10) * (1 + qnorm(0.975)/sqrt(lhtest$F)[2])
```

```
         [,1]
[1,] 3.351629
```

```
coef(est2) %*% c(0, 1, 2 * 10) * (1 - qnorm(0.975)/sqrt(lhtest$F)[2])
```

```
         [,1]
[1,] 2.658022
```

Vorgehen:

1. theoretische Motivation für nichtlineare Beziehung
2. spezifiziere funktionale Form
3. testen ob nichtlineare Funktion gerechtfertigt ist
4. visueller Test
5. marginale Effekte

5.1 Funktionale Formen

5.1.1 Polynome

$$Y_i = \beta_0 + \beta_1 X_i + \beta_2 X_i^2 + \beta_3 X_i^3 + \ldots + \beta_r X_i^r + u_i$$

($r = 2$: quadratisches Modell, $r = 3$: kubisches Modell, ...)

- Test der Nullhypothese dass die Regressionsfunktion linear ist:

$$H_0 : \beta_2 = 0 \wedge \beta_3 = 0 \wedge \ldots \wedge \beta_r = 0$$

versus

$$H_1 : \text{wenigstens ein } \beta_j \neq 0, j \in \{2, \ldots, r\}$$

- was ist das richtige r?
  - großes r: mehr Flexibilität, besserer Fit

- kleines r: präzisere Schätzung der einzelnen Koeffizienten

sequentieller Hypothesentest bei polynomialen Modellen:

1. Wähle den größten sinnvollen Wert von r und schätze eine polynomiale Regression
2. teste $H_0 : \beta_r = 0$. Wenn H_0 abgelehnt wird, dann verwende ein Polynom r-ten Grades
3. Sonst: reduziere r um 1. Weiter bei Schritt 1.

```
avginc3 <- avginc * avginc * avginc
est3 <- lm(testscr ~ avginc + avginc2 + avginc3)
lines(avginc[or], fitted(est3)[or], col = "yellow", lwd = 3)
(lhtest <- linear.hypothesis(est3, "avginc3", vcov = hccm))
```

```
Linear hypothesis test

Hypothesis:
avginc3 = 0

Model 1: testscr ~ avginc + avginc2 + avginc3
Model 2: restricted model

Note: Coefficient covariance matrix supplied.

  Res.Df  Df      F Pr(>F)
1    416
2    417  -1 2.4615 0.1174
```

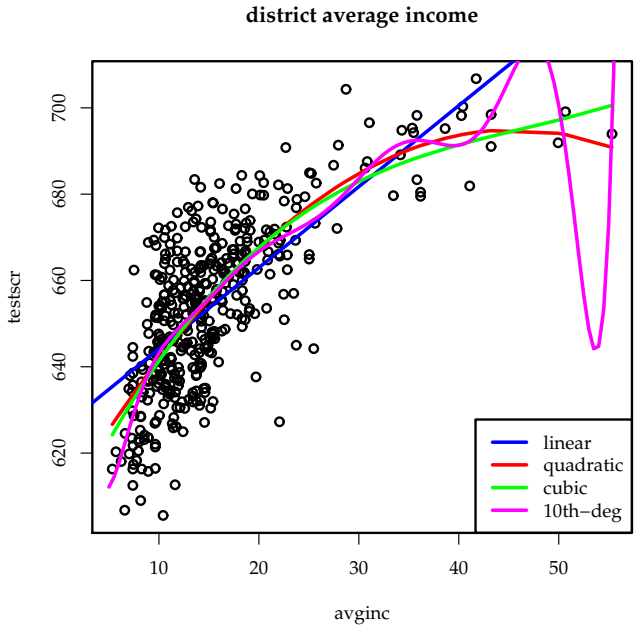
```
est2 <- lm(testscr ~ avginc + avginc2)
(lhtest <- linear.hypothesis(est3, "avginc2", vcov = hccm))
```

```
Linear hypothesis test

Hypothesis:
avginc2 = 0

Model 1: testscr ~ avginc + avginc2 + avginc3
```

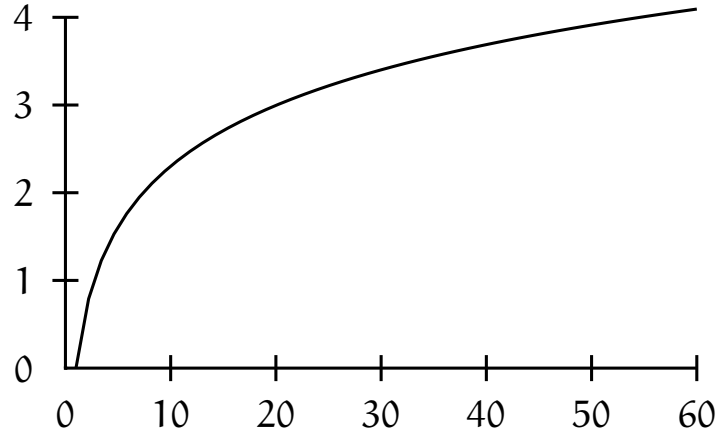
```
Model 2: restricted model

Note: Coefficient covariance matrix supplied.

  Res.Df  Df      F   Pr(>F)
1    416
2    417  -1 7.9158 0.005133 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
estp <- lm(testscr ~ poly(avginc, 10))
```

```
plot(testscr ~ avginc, main = "district average income")
abline(est1, col = "blue", lwd = 3)
lines(avginc[or], fitted(est2)[or], col = "red", lwd = 3)
lines(avginc[or], fitted(est3)[or], col = "green", lwd = 3)
smooth <- list(avginc = seq(5, 70, 0.5))
lines(smooth$avginc, predict(estp, newdata = smooth),
    col = "magenta", lwd = 3)
legend("bottomright", c("linear", "quadratic", "cubic",
    "10th-deg"), lwd = 3, col = c("blue", "red", "green",
    "magenta"))
```



5.1.2 Logarithmische Modelle



- $Y_i = \beta_0 + \beta_1 \cdot \ln X_i + u_i$ linear-log
- $\ln Y_i = \beta_0 + \beta_1 \cdot X_i + u_i$ log-linear
- $\ln Y_i = \beta_0 + \beta_1 \cdot \ln X_i + u_i$ log-log

5.1.3 Logarithmische Modelle - linear-log

$$Y_i = \beta_0 + \beta_1 \cdot \ln X_i + u_i$$

marginale Effekte:

$$\frac{\partial Y_i}{\partial X_i} = \beta_1 \frac{1}{X_i}$$

$$\Delta Y_i \approx \Delta X_i \cdot \beta_1 \frac{1}{X_i}$$

wenn sich X_i um 1% ändert ($\Delta X_i = 0.01 \cdot X_i$) ...

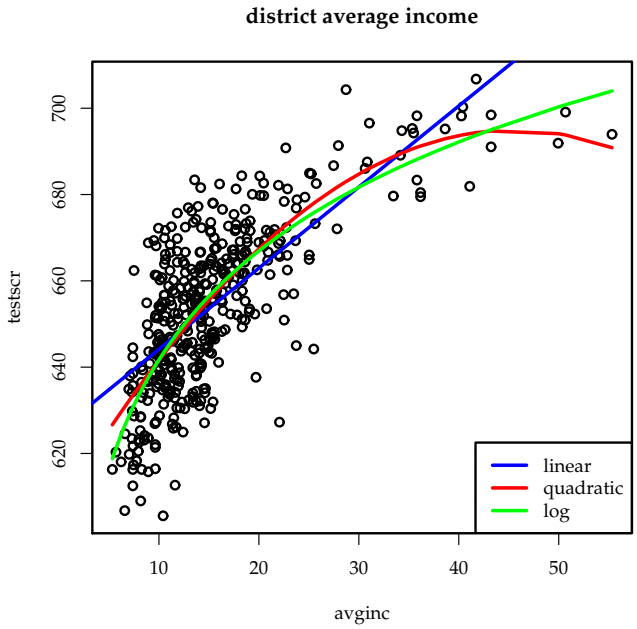
$$\Delta Y_i \approx 0.01 X_i \cdot \beta_1 \frac{1}{X_i}$$

... dann ändert sich Y_i um $0.01 \cdot \beta_1$

```
(estL <- lm(testscr ~ log(avginc)))
```

```
Call:
lm(formula = testscr ~ log(avginc))

Coefficients:
(Intercept)  log(avginc)
     557.83        36.42
```

```
plot(testscr ~ avginc, main = "district average income")
abline(est1, col = "blue", lwd = 3)
lines(avginc[or], fitted(est2)[or], col = "red", lwd = 3)
lines(avginc[or], fitted(estL)[or], col = "green", lwd = 3)
legend("bottomright", c("linear", "quadratic", "log"),
    lwd = 3, col = c("blue", "red", "green"))
```



Marginale Effekte:

```
coef(estL)[2]/10
```

```
log(avginc)
   3.641968
```

```
coef(estL)[2]/40
```

```
log(avginc)
   0.910492
```

5.1.4 Logarithmische Modelle - log-linear

$$\ln Y_i = \beta_0 + \beta_1 \cdot X_i + u_i$$

marginale Effekte

$$\frac{\partial \ln Y_i}{\partial X_i} = \beta_1$$

$$\frac{\Delta \ln Y_i}{\Delta X_i} \approx \beta_1$$

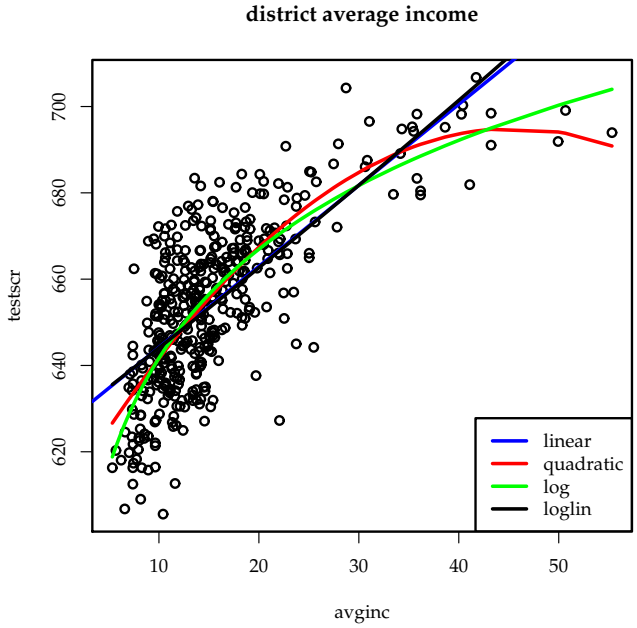
$$\frac{\Delta Y_i}{Y_i} \approx \Delta \ln Y_i \approx \beta_1 \cdot \Delta X_i$$

Eine Änderung von X_i um eine Einheit bedeutet eine Änderung von Y_i um den Anteil β_1

```
(estLL <- lm(log(testscr) ~ avginc))
```

```
Call:
lm(formula = log(testscr) ~ avginc)

Coefficients:
(Intercept)       avginc
   6.439362     0.002844
```

```
plot(testscr ~ avginc, main = "district average income")
abline(est1, col = "blue", lwd = 3)
lines(avginc[or], fitted(est2)[or], col = "red", lwd = 3)
lines(avginc[or], fitted(estL)[or], col = "green", lwd = 3)
lines(avginc[or], exp(fitted(estLL))[or], col = "black",
    lwd = 3)
legend("bottomright", c("linear", "quadratic", "log",
    "loglin"), lwd = 3, col = c("blue", "red", "green",
    "black"))
```




```
coef(estL)[2]
```

```
log(avginc)
   36.41968
```

exp years of full-time work experience
lwage logarithm of wage

```
library(lattice)
data(Wages, package = "Ecdat")
lm(lwage ~ exp, data = Wages)
```

```
Call:
lm(formula = lwage ~ exp, data = Wages)

Coefficients:
(Intercept)          exp
    6.50143      0.00881
```

5.1.5 Logarithmische Modelle - log-log

$$\ln Y_i = \beta_0 + \beta_1 \cdot \ln X_i + u_i$$

$$Y_i = e^{\beta_0} \cdot X_i^{\beta_1} \cdot e^{u_i}$$

marginaler Effekt:

$$\frac{\partial Y_i}{\partial X_i} = e^{\beta_0} \cdot \beta_1 X_i^{\beta_1 - 1} = \beta_1 \frac{Y_i}{X_i}$$

$$\frac{\partial Y_i}{\partial X_i} \cdot \frac{X_i}{Y_i} = \beta_1$$

β_1 gibt die Elastizität von Y_i auf X_i an.

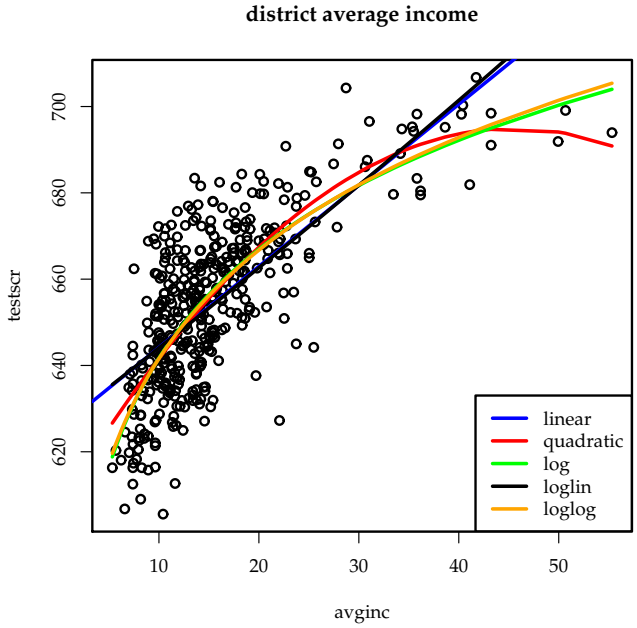
```
(estLLL <- lm(log(testscr) ~ log(avginc)))
```

```
Call:
lm(formula = log(testscr) ~ log(avginc))

Coefficients:
(Intercept)  log(avginc)
    6.33635      0.05542
```

```
(estLL <- lm(log(testscr) ~ avginc))
```

```
Call:
lm(formula = log(testscr) ~ avginc)

Coefficients:
(Intercept)       avginc
   6.439362     0.002844
```

```
plot(testscr ~ avginc, main = "district average income")
abline(est1, col = "blue", lwd = 3)
lines(avginc[or], fitted(est2)[or], col = "red", lwd = 3)
lines(avginc[or], fitted(estL)[or], col = "green", lwd = 3)
lines(avginc[or], exp(fitted(estLL))[or], col = "black",
    lwd = 3)
lines(avginc[or], exp(fitted(estLLL))[or], col = "orange",
    lwd = 3)
legend("bottomright", c("linear", "quadratic", "log",
    "loglin", "loglog"), lwd = 3, col = c("blue", "red",
    "green", "black", "orange"))
```



5.1.6 Vergleich der 3 logarithmischen Modelle

- X und/oder Y werden jeweils transformiert
- Die Regressionsgleichung ist linear in den transformierten Variablen
- Hypothesentests und Konfizendintervalle können also wie gewohnt bestimmt werden
- Die Interpretation von β ist jeweils unterschiedlich
- R^2 ist geeignet log-log und log-linear zu vergleichen
- R^2 ist geeignet linear-log und lineares Modell zu vergleichen
- Ein Vergleich der Modelle mit $\ln Y_i$ und Y_i ist nicht möglich.

$\rightarrow$ ökonomische Theorie ist erforderlich, um eine der vier Spezifikationen zu motivieren.

5.1.7 Verallgemeinerung — Box-Cox

Das logarithmische Modell

$$\log Y = X'\beta + u$$

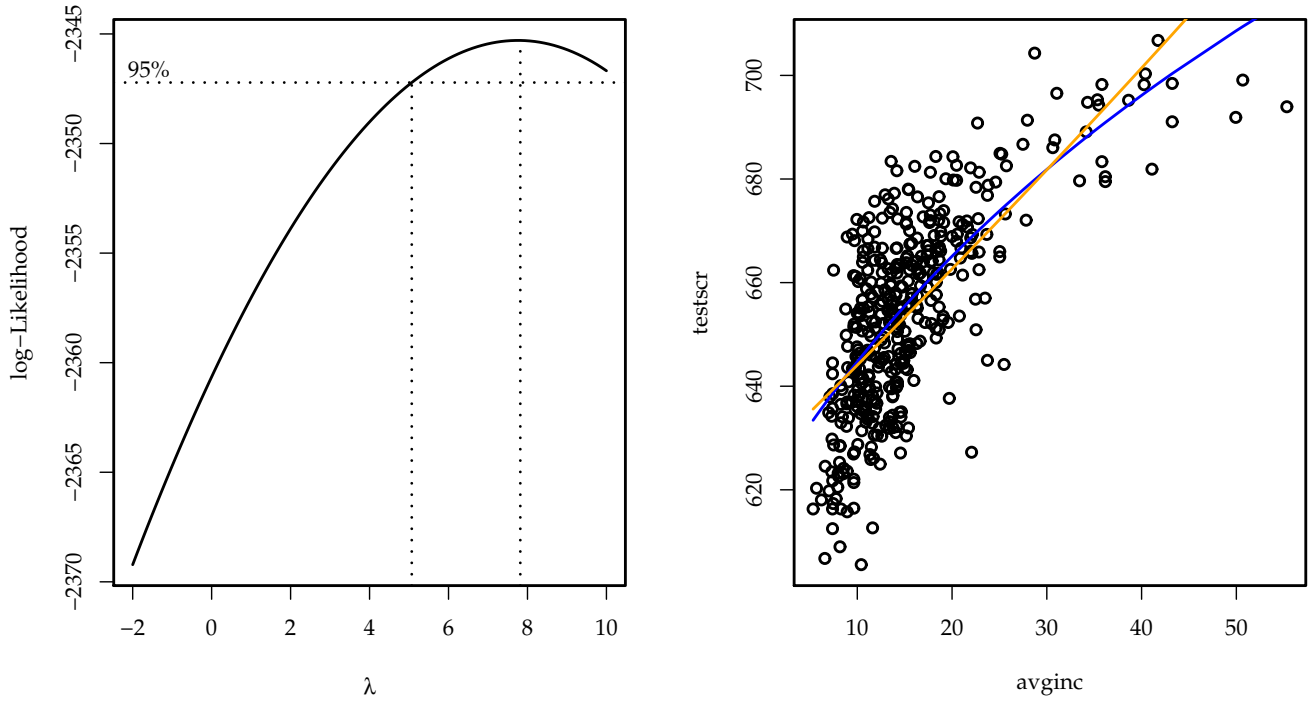
betrachte nun

$$g_\lambda(Y) = X'\beta + u$$

$$\text{wobei } g_\lambda(Y) = \begin{cases} \frac{Y^\lambda - 1}{\lambda} & \text{falls } \lambda \neq 0 \\ \log Y & \text{falls } \lambda = 0 \end{cases}$$

λ wird per maximum likelihood bestimmt.

```
library(MASS)
est <- lm(testscr ~ avginc)
par(mfrow = c(1, 2))
boxcox(est, lambda = seq(-2, 10, by = 0.5), plotit = TRUE)
est2 <- lm(testscr^8 ~ avginc)
plot(testscr ~ avginc)
lines(fitted(est2)[or]^(1/8) ~ avginc[or], col = "blue")
lines(exp(fitted(estLL)[or]) ~ avginc[or], col = "orange")
par(mfrow = c(1, 1))
```



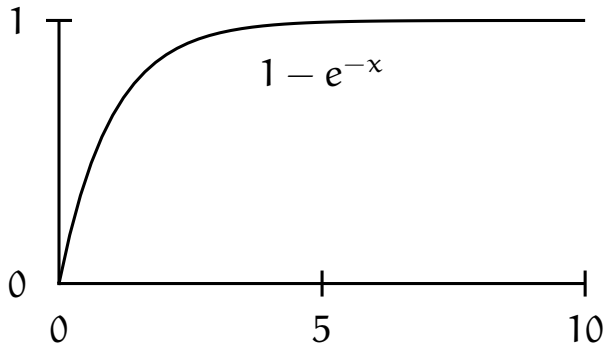
5.1.8 Andere nichtlineare Funktionen

Probleme mit den obigen Modellen:

- Polynomiales Modell: eventuell nicht monoton.
- linear-log: `testscr` steigt monoton mit `avginc`, ist aber nach oben nicht beschränkt.
- Gibt es keine Spezifikation die beides erfüllt: Monotonie und Beschränktheit?

$$Y = \beta_0 - \alpha e^{-\beta_1 X}$$

(negative exponentielle Wachstumskurve)



Schätze die Parameter von

$$Y_i = \beta_0 - \alpha e^{-\beta_1 X_i} + u_i$$

oder (mit $\alpha = \beta_0 e^{\beta_2}$)

$$Y_i = \beta_0 \left(1 - e^{-\beta_1 (X_i - \beta_2)}\right) + u_i$$

Vergleiche dieses Modell mit linear-log oder polynomialen Modell:

$$\begin{aligned} Y_i &= \beta_0 + \beta_1 \cdot \ln X_i + u_i \\ Y_i &= \beta_0 + \beta_1 \cdot X_i + \beta_2 \cdot X_i^2 + \beta_3 \cdot X_i^3 + u_i \end{aligned}$$

Linearisierung von $Y_i = \beta_0 \left(1 - e^{-\beta_1 (X_i - \beta_2)}\right) + u_i$ ist nicht mehr möglich.

5.1.9 Nichtlineare kleinste Quadrate

- Modelle die linear in ihren Parametern sind, können mit OLS geschätzt werden.
- Modelle die nichtlinear in einem oder mehreren Paramteren sind, können mit nichtlinearen Methoden geschätzt werden (aber nicht mit OLS).

$$\min_{\beta_0,\beta_1,\beta_2} \sum_{i=1}^{n} \left(Y_i - \beta_0 \left(1 - e^{-\beta_1 (X_i - \beta_2)} \right) \right)^2$$

```
(nest <- nls(testscr ~ b0 * (1 - exp(-1 * b1 * (avginc -
    b2))), start = c(b0 = 730, b1 = 0.1, b2 = 0), trace = TRUE))
```

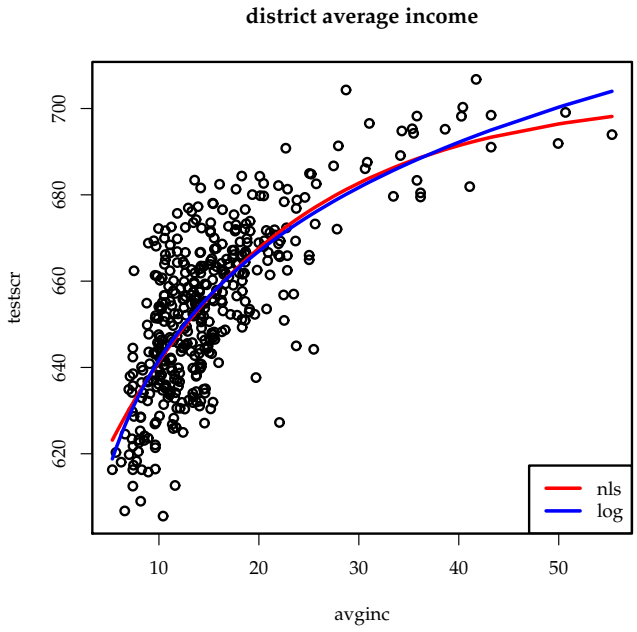
```
7485378 :  730.0   0.1   0.0
1392009 :  695.41660291   0.09260118  -8.24400488
233046.7 :  696.82401653   0.07926959 -17.01056360
98541.8 :  699.44122774   0.06495657 -25.58741504
69931.37 :  702.08095623   0.05753085 -31.61313562
67013.75 :  703.05046932   0.05552977 -33.72481577
66988.4 :  703.20480076   0.05525916 -33.98665933
66988.4 :  703.22080355   0.05523593 -34.00237947
66988.4 :  703.22210308   0.05523406 -34.00353771
Nonlinear regression model
  model:  testscr ~ b0 * (1 - exp(-1 * b1 * (avginc - b2)))
   data:  parent.frame()
       b0        b1        b2
703.22210   0.05523 -34.00354
 residual sum-of-squares: 66988

Number of iterations to convergence: 8
Achieved convergence tolerance: 0.000000701
```

```
summary(nest)
```

```
Formula: testscr ~ b0 * (1 - exp(-1 * b1 * (avginc - b2)))

Parameters:
     Estimate Std. Error t value      Pr(>|t|)
```

```
b0 703.222103   6.697451 104.998       < 2e-16 ***
b1   0.055234   0.009101   6.069 0.00000000289 ***
b2 -34.003538   5.676787  -5.990 0.00000000454 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 12.67 on 417 degrees of freedom

Number of iterations to convergence: 8
Achieved convergence tolerance: 0.000000701
```

```
plot(testscr ~ avginc, main = "district average income")
lines(avginc[or], fitted(nest)[or], col = "red", lwd = 3)
lines(avginc[or], fitted(estL)[or], col = "blue", lwd = 3)
legend("bottomright", c("nls", "log"), lwd = 3, col = c("red",
    "blue"))
```



5.2 Interaktionen

- Vielleicht hängt der Effekt der Klassengröße auf den Testscore von weiteren Umständen ab?

- Vielleicht sind kleine Klassen gerade bei vielen Ausländern in der Klasse hilfreich, sonst aber nicht?
- $\frac{\partial \texttt{testscr}}{\partial \texttt{str}}$ hängt von `elpct` ab.
- Allgemein: $\frac{\partial Y}{\partial X_1}$ hängt von X_2 ab.
- Wie kann man diese "Interaktion" modellieren?
- Betrachte zunachst binäre X, dann stetige.

Beispiel 1:

$$\texttt{testsrc} = \beta_1 \texttt{str} + \beta_2 \texttt{elpct} + \beta_0 + u$$

in diesem Modell ist der Einfluss von `str` unabhängig von `elpct`
Beispiel 2:

$$\texttt{lwage} = \beta_1 \texttt{ed} + \beta_0 + u$$

```
library(lattice)
attach(Wages)
summary(ed)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   4.00   12.00   12.00   12.85   16.00   17.00
```

```
lm(lwage ~ ed)
```

```
Call:
lm(formula = lwage ~ ed)

Coefficients:
(Intercept)           ed
     5.8388       0.0652
```

Beispiel 3:

$$\texttt{lwage} = \beta_1 \texttt{college} + \beta_2 \texttt{sex} + \beta_0 + u$$

```
college = ed > 16
lm(lwage ~ college + sex)
```

```
Call:
lm(formula = lwage ~ college + sex)

Coefficients:
(Intercept)  collegeTRUE      sexmale
     6.2254       0.3340       0.4626
```

5.2.1 Interaktion zwischen binären Variablen

$$\texttt{lwage} = \underbrace{\beta_0}_{6.21} + \underbrace{\beta_1}_{0.55}\texttt{college} + \underbrace{\beta_2}_{0.49}\texttt{sex} + \underbrace{\beta_3}_{-0.24}\texttt{sex}\cdot\texttt{college} + u$$

```
(est <- lm(lwage ~ college + sex + sex:college))
```

```
Call:
lm(formula = lwage ~ college + sex + sex:college)

Coefficients:
        (Intercept)          collegeTRUE              sexmale
             6.2057               0.5543               0.4850
collegeTRUE:sexmale
            -0.2412
```

Anstelle der Koeffizienten der Regression können wir auch Mittelwerte der einzelnen Kategorien berechnen:

```
mean(lwage[college == FALSE & sex == "female"])
```

```
[1] 6.205665
```

```
mean(lwage[college == TRUE & sex == "female"])
```

```
[1] 6.760007
```

```
mean(lwage[college == FALSE & sex == "male"])
```

```
[1] 6.690634
```

```
mean(lwage[college == TRUE & sex == "male"])
```

```
[1] 7.00375
```

```
detach(Wages)
```

mean(lwage)

college	sex: female	sex: male
FALSE	6.21 β_0	6.69 $\beta_0 + \beta_2$
TRUE	6.76 $\beta_0 + \beta_1$	7.00 $\beta_0 + \beta_1 + \beta_2 + \beta_3$

Effekt von College bei Frauen: β_1 Effekt von College bei Männern: $\beta_1 + \beta_3$

```
Histr = str >= 20
Hiel = elpct >= 10
table(Histr, Hiel)
```

```
       Hiel
Histr   FALSE TRUE
  FALSE   149   89
  TRUE     79  103
```

```
(est <- lm(testscr ~ Histr * Hiel))
```

```
Call:
lm(formula = testscr ~ Histr * Hiel)

Coefficients:
       (Intercept)           HistrTRUE            HielTRUE
           664.143              -1.908             -18.163
HistrTRUE:HielTRUE
            -3.494
```

```
mean(testscr[Hiel == FALSE & Histr == FALSE])
```

```
[1] 664.1433
```

```
mean(testscr[Hiel == TRUE & Histr == FALSE])
```

```
[1] 645.9803
```

```
coef(est) %*% c(1, 0, 1, 0)
```

```
         [,1]
[1,] 645.9803
```

```
mean(testscr[Hiel == FALSE & Histr == TRUE])
```

```
[1] 662.2354
```

```
coef(est) %*% c(1, 1, 0, 0)
```

```
         [,1]
[1,] 662.2354
```

```
mean(testscr[Hiel == TRUE & Histr == TRUE])
```

```
[1] 640.5782
```

```
coef(est) %*% c(1, 1, 1, 1)
```

```
         [,1]
[1,] 640.5782
```

Wenn wir keine Lust haben, für jede einzelne Kategorie den Mittelwert auszurechnen, dann können wir das auch R überlassen:

```
library(memisc)
aggregate(mean(testscr) ~ Hiel + Histr)
```

```
   Hiel Histr mean(testscr)
1 FALSE FALSE      664.1433
3  TRUE FALSE      645.9803
2 FALSE  TRUE      662.2354
6  TRUE  TRUE      640.5782
```

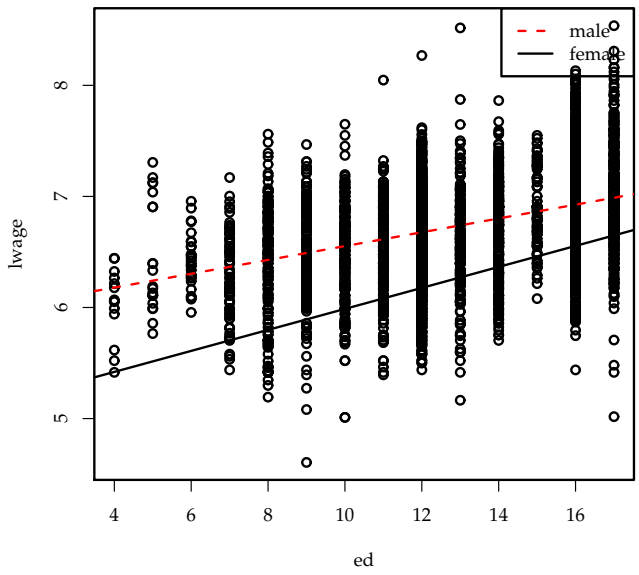
$$\texttt{testscr} = \underbrace{\beta_0}_{664.14} + \underbrace{\beta_1}_{-18.16}\,\texttt{Hiel} + \underbrace{\beta_2}_{-1.91}\,\texttt{Histr} + \underbrace{\beta_3}_{-3.49}\,\texttt{Histr}\cdot\texttt{Hiel} + u$$

mean(testscr)

Hiel \ Histr	FALSE	TRUE
FALSE	664.1433 β_0	662.2354 $\beta_0+\beta_2$
TRUE	645.9803 $\beta_0+\beta_1$	640.5782 $\beta_0+\beta_1+\beta_2+\beta_3$

5.2.2 Interaktion zwischen einer binären und einer stetigen Variablen

```
attach(Wages)
plot(lwage ~ ed)
abline(lm(lwage ~ ed, subset = (sex == "female")))
abline(lm(lwage ~ ed, subset = (sex == "male")), col = "red",
    lty = 2)
legend("topright", c("male", "female"), lty = 2:1, col = c("red",
    "black"))
```




```
(lm(lwage ~ ed, subset = (sex == "female")))
```

```
Call:
lm(formula = lwage ~ ed, subset = (sex == "female"))
```

```
Coefficients:
(Intercept)           ed
    5.04207      0.09452
```

```
(lm(lwage ~ ed, subset = (sex == "male")))
```

```
Call:
lm(formula = lwage ~ ed, subset = (sex == "male"))

Coefficients:
(Intercept)           ed
    5.93060      0.06221
```

```
(est <- lm(lwage ~ sex * ed))
```

```
Call:
lm(formula = lwage ~ sex * ed)

Coefficients:
(Intercept)      sexmale           ed   sexmale:ed
    5.04207      0.88854      0.09452     -0.03231
```

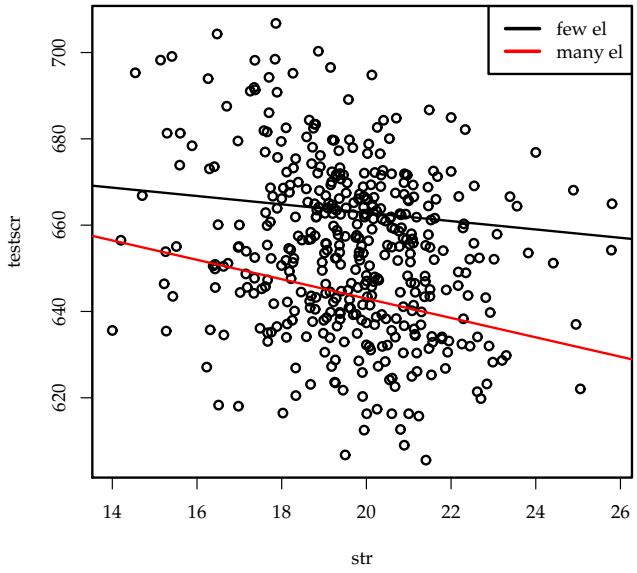
```
detach(Wages)
```

$$\texttt{lwage} = \beta_0 + \beta_1 \texttt{ed} + \beta_2 \texttt{sex} + \beta_3 \texttt{sex} \cdot \texttt{ed} + u$$

- $\beta_3 = 0$: Regressionsgeraden sind parallel
- $\beta_2 = 0$: Regressionsgeraden haben den gleichen Achsenabschnitt

Hängt der Effekt der Klassengröße auf Testscores vom Anteil der Mutterspachler ab?

```
Hiel = elpct >= 10
plot(testscr ~ str)
abline(lm(testscr ~ str, subset = (Hiel == FALSE)))
abline(lm(testscr ~ str, subset = (Hiel == TRUE)), col = "red")
legend("topright", c("few el", "many el"), lwd = 3, col = c("black",
    "red"))
```




```
(lm(testscr ~ str, subset = (Hiel == FALSE)))
```

```
Call:
lm(formula = testscr ~ str, subset = (Hiel == FALSE))

Coefficients:
(Intercept)          str
   682.2458      -0.9685
```

```
(lm(testscr ~ str, subset = (Hiel == TRUE)))
```

```
Call:
lm(formula = testscr ~ str, subset = (Hiel == TRUE))

Coefficients:
(Intercept)          str
    687.885       -2.245
```

```
lm(testscr ~ str * Hiel)
```

```
Call:
lm(formula = testscr ~ str * Hiel)
```

```
Coefficients:
 (Intercept)           str      HielTRUE  str:HielTRUE
    682.2458       -0.9685        5.6391       -1.2766
```

```
(est <- lm(testscr ~ str * Hiel))
```

```
Call:
lm(formula = testscr ~ str * Hiel)

Coefficients:
 (Intercept)           str      HielTRUE  str:HielTRUE
    682.2458       -0.9685        5.6391       -1.2766
```

$$\texttt{testscr} = \underbrace{\beta_0}_{682.2458} + \underbrace{\beta_1}_{5.6391}\,\texttt{Hiel} + \underbrace{\beta_2}_{-0.9685}\,\texttt{str} + \underbrace{\beta_3}_{-1.2766}\,\texttt{str}\cdot\texttt{Hiel} + u$$

Effekt einer Änderung der Klassengröße `str`:

- wenn `elpct`<10: -0.9685
- wenn `elpct`≥ 10: -2.245

Sind die beiden Geraden parallel?

```
linear.hypothesis(est, "str:HielTRUE=0", vcov = hccm)
```

```
Linear hypothesis test

Hypothesis:
str:HielTRUE = 0

Model 1: testscr ~ str * Hiel
Model 2: restricted model

Note: Coefficient covariance matrix supplied.

  Res.Df  Df      F Pr(>F)
1    416
2    417  -1 1.6778 0.1959
```

Sind beide Geraden identisch?

```
linear.hypothesis(est, c("str:HielTRUE=0", "HielTRUE=0"),
    vcov = hccm)
```

```
Linear hypothesis test

Hypothesis:
str:HielTRUE = 0
HielTRUE = 0

Model 1: testscr ~ str * Hiel
Model 2: restricted model

Note: Coefficient covariance matrix supplied.

  Res.Df  Df      F    Pr(>F)
1    416
2    418  -2 88.806 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Haben beide Geraden den gleichen Achsenabschnitt?

```
linear.hypothesis(est, "HielTRUE=0", vcov = hccm)
```

```
Linear hypothesis test

Hypothesis:
HielTRUE = 0

Model 1: testscr ~ str * Hiel
Model 2: restricted model

Note: Coefficient covariance matrix supplied.

  Res.Df  Df      F Pr(>F)
1    416
2    417  -1 0.0804 0.7769
```

5.2.3 Anwendung: Gender gap

exp	years of full-time work experience
wks	weeks worked
bluecol	blue collar ?
ind	works in a manufacturing industry ?
south	resides in the south ?
smsa	resides in a standard metropolitan statistical area ?
married	married ?
sex	a factor with levels (male,female)
union	individual's wage set by a union contract ?
ed	years of education
black	is the individual black ?
lwage	logarithm of wage

`ifelse` gibt, abhängig vom ersten Argument, entweder das zweite oder dritte Argument zurück. `as.data.frame` wandelt das Argument, z.B. eine Matrix, in einen Dataframe um. Das ist hier nützlich, weil die ausgegebene Struktur Zahlen und Zeichenketten mischt. `colnames` erlaubt es, auf Spaltennamen zuzugreifen (und diese hier zu verändern).

```
attach(Wages)
library(lmtest)
lmr <- function(...) {
    est <- lm(...)
    print(coeftest(est, vcov = hccm))
    cat("R2=         ", round(summary(est)$r.squared,
        2), "\n")
}
est1 <- lm(lwage ~ ed)
est2 <- lm(lwage ~ ed + sex)
est3 <- lm(lwage ~ ed * sex)
est4 <- lm(lwage ~ ed * sex + exp + black + union + south +
    wks + married + smsa + ind)
mtable(`(1)` = est1, `(2)` = est2, `(3)` = est3, `(4)` = est4,
    summary.stats = c("R-squared", "N"))
```

```
Calls:
(1): lm(formula = lwage ~ ed)
(2): lm(formula = lwage ~ ed + sex)
(3): lm(formula = lwage ~ ed * sex)
(4): lm(formula = lwage ~ ed * sex + exp + black + union + south +
    wks + married + smsa + ind)

=============================================================
                            (1)       (2)       (3)       (4)
-------------------------------------------------------------
(Intercept)              5.839***  5.419***  5.042***  4.666***
                        (0.032)   (0.034)   (0.087)   (0.107)
ed                       0.065***  0.065***  0.095***  0.086***
                        (0.002)   (0.002)   (0.007)   (0.006)
sex: male/female                   0.474***  0.889***  0.552***
                                  (0.018)   (0.093)   (0.092)
ed x sex: male/female                       -0.032*** -0.016*
                                            (0.007)   (0.007)
exp                                                    0.011***
                                                      (0.001)
black: yes/no                                         -0.168***
                                                      (0.022)
union: yes/no                                          0.063***
                                                      (0.012)
south: yes/no                                         -0.055***
                                                      (0.013)
wks                                                    0.005***
                                                      (0.001)
married: yes/no                                        0.066**
                                                      (0.022)
smsa: yes/no                                           0.161***
                                                      (0.012)
ind                                                    0.043***
                                                      (0.012)
-------------------------------------------------------------
R-squared                   0.155     0.260     0.264     0.387
N                        4165      4165      4165      4165
=============================================================
```

5.2.4 Interaktion zwischen zwei stetigen Variablen

Beispiel

$$\texttt{lwage} = \beta_0 + \beta_1 \texttt{ed} + \beta_2 \texttt{exp} + \beta_3 \texttt{ed} \cdot \texttt{exp} + u$$

```
est1 <- lm(lwage ~ ed + exp)
est2 <- lm(lwage ~ ed * exp)
mtable(`(1)` = est1, `(2)` = est2, summary.stats = c("R-squared",
    "N"))
```

```
Calls:
(1): lm(formula = lwage ~ ed + exp)
(2): lm(formula = lwage ~ ed * exp)

================================
                (1)       (2)
--------------------------------
(Intercept)   5.436***  5.446***
             (0.037)   (0.075)
ed            0.076***  0.076***
             (0.002)   (0.005)
exp           0.013***  0.013***
             (0.001)   (0.003)
ed x exp                0.000
                         (0.000)
--------------------------------
R-squared       0.247     0.247
N            4165      4165
================================
```

$$Y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \beta_3 (X_{1i} \cdot X_{2i}) + u_i$$

Marginale Effekte:

$$\frac{\partial Y_i}{\partial X_1} = \beta_1 + \beta_3 X_2 \qquad \frac{\partial Y_i}{\partial X_2} = \beta_2 + \beta_3 X_1$$

Was passiert, wenn X_1 sich um ΔX_1 ändert und X_2 ändert sich um ΔX_2?

$$\begin{aligned}
\Delta Y &= \beta_0 + \beta_1(X_1 + \Delta X_1) + \beta_2(X_2 + \Delta X_2) + \beta_3\left((X_1 + \Delta X_1)\cdot(X_2 + \Delta X_2)\right) - \\
&\quad - \left(\beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3(X_1 \cdot X_2)\right) \\
&= \beta_1\Delta X_1 + \beta_2\Delta X_2 + \beta_3(X_1 \cdot X_2 + \Delta X_1 \cdot X_2 + \Delta X_2 \cdot X_1 + \Delta X_1 \cdot \Delta X_2) - \\
&\quad - \beta_3 X_1 \cdot X_2 \\
&= \beta_1\Delta X_1 + \beta_2\Delta X_2 + \beta_3\Delta X_1 \cdot X_2 + \beta_3\Delta X_2 \cdot X_1 + \beta_3\Delta X_1 \cdot \Delta X_2 \\
&= (\beta_1 + \beta_3 X_2)\Delta X_1 + (\beta_2 + \beta_3 X_1)\Delta X_2 + \beta_3\Delta X_1\Delta X_2
\end{aligned}$$

$$\texttt{testscr} = \underbrace{\beta_0}_{686.34} + \underbrace{\beta_1}_{-1.1170}\,\texttt{str} + \underbrace{\beta_2}_{-0.6729}\,\texttt{elpct} + \underbrace{\beta_3}_{0.001162}\,\texttt{str}\cdot\texttt{elpct} + u$$

```
attach(Caschool)
lmr(testscr ~ str * elpct)
```

```
t test of coefficients:

              Estimate  Std. Error t value Pr(>|t|)
(Intercept) 686.3385246  11.9378546 57.4926  < 2e-16 ***
str          -1.1170183   0.5965150 -1.8726  0.06183 .
elpct        -0.6729114   0.3865378 -1.7409  0.08245 .
str:elpct     0.0011618   0.0191576  0.0606  0.95167
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

R2=         0.43
```

Was ist der Effekt der Klassengröße `str` für eine Klasse mit Medianausländeranteil?

`median` bestimmt den Median eines Vektors. `quantile` bestimmt Quantile eines Vektors. Die kleinste Beobachtung entspricht einem Quantil von 0, die größte Beobachtung einem Quantil von 1. `mean` bestimmt den arithmetischen Mittelwert.

```
median(elpct)
```

```
[1] 8.777634
```

```
est <- lm(testscr ~ str * elpct)
coef(est)
```

```
  (Intercept)           str         elpct     str:elpct
686.338524629  -1.117018345  -0.672911392   0.001161752
```

```
(eff1 = coef(est)["str"] + coef(est)["str:elpct"] * median(elpct))
```

```
     str
-1.106821
```

Wie ändert sich dieser Effekt für eine Klasse mit Ausländeranteil im 75%-Quantil?

```
quantile(elpct, 0.5)
```

```
    50%
8.777634
```

```
quantile(elpct, 0.75)
```

```
    75%
22.97000
```

```
(eff2 = coef(est)["str"] + coef(est)["str:elpct"] * quantile(elpct,
    0.75))
```

```
     str
-1.090333
```

Ist der Interaktionsterm signifikant?

```
linear.hypothesis(est, "str:elpct=0", vcov = hccm)
```

```
Linear hypothesis test

Hypothesis:
str:elpct = 0

Model 1: testscr ~ str * elpct
Model 2: restricted model

Note: Coefficient covariance matrix supplied.

  Res.Df  Df      F Pr(>F)
1    416
2    417  -1 0.0037 0.9517
```

5.3 Nichtlineare Interaktionsterme

```
Hiel = elpct >= 10
est1 <- lm(testscr ~ str + elpct + mealpct)
est2 <- lm(testscr ~ str + elpct + mealpct + log(avginc))
est3 <- lm(testscr ~ str * Hiel)
est4 <- lm(testscr ~ str * Hiel + mealpct + log(avginc))
est5 <- lm(testscr ~ str + I(str^2) + I(str^3) + Hiel +
    mealpct + log(avginc))
est6 <- lm(testscr ~ (str + I(str^2) + I(str^3)) * Hiel +
    mealpct + log(avginc))
est7 <- lm(testscr ~ str + I(str^2) + I(str^3) + elpct +
    mealpct + log(avginc))
```

```
mtable(`(1)` = est1, `(2)` = est2, `(3)` = est3, `(4)` = est4,
    `(5)` = est5, `(6)` = est6, `(7)` = est7, summary.stats = c("R-squared",
        "N"))
```

	(1)	(2)	(3)	(4)	(5)	(6)	(7)
(Intercept)	700.150***	658.552***	682.246***	653.666***	252.051	122.354	244.809
	(5.641)	(8.749)	(12.071)	(10.053)	(179.724)	(205.050)	(181.899)
str	−0.998***	−0.734**	−0.968	−0.531	64.339*	83.701**	65.285*
	(0.274)	(0.261)	(0.599)	(0.350)	(27.295)	(31.506)	(27.708)
elpct	−0.122***	−0.176***					−0.166***
	(0.033)	(0.034)					(0.035)
mealpct	−0.547***	−0.398***		−0.411***	−0.420***	−0.418***	−0.402***
	(0.024)	(0.034)		(0.029)	(0.029)	(0.029)	(0.034)
log(avginc)		11.569***		12.124***	11.748***	11.800***	11.509***
		(1.841)		(1.823)	(1.799)	(1.809)	(1.834)
Hiel			5.639	5.498	−5.474***	816.075*	
			(19.889)	(10.012)	(1.046)	(354.100)	
str × Hiel			−1.277	−0.578		−123.282*	
			(0.986)	(0.507)		(54.290)	
str^2					−3.424*	−4.381**	−3.466*
					(1.373)	(1.597)	(1.395)
str^3					0.059**	0.075**	0.060**
					(0.023)	(0.027)	(0.023)
str^2×Hiel						6.121*	
						(2.752)	
str^3×Hiel						−0.101*	
						(0.046)	
R-squared	0.775	0.796	0.310	0.797	0.801	0.803	0.801
N	420	420	420	420	420	420	420

Es sieht so aus, als gäbe es einen nichtlinearen Effect von `str` auf `testscr`. Betrachten wir nochmals Modell 6 und bestimmen den marginalen Effekt von `str`.

```
estC <- coef(est6)
mEffstr <- function(str, Hiel) {
    estC %*% c(0, 1, 2 * str, 3 * str^2, 0, 0, 0, Hiel,
        Hiel * 2 * str, Hiel * 3 * str^2)
}
mEffstr(20, 0)
```

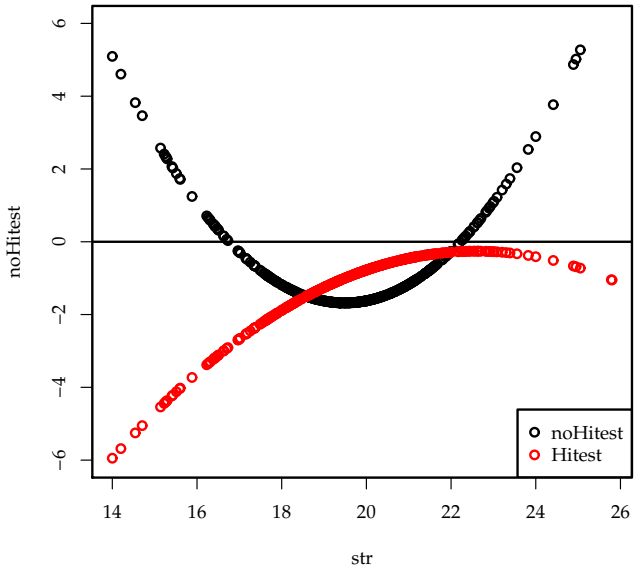
```
          [,1]
[1,] -1.622543
```

```
mEffstr(20, 1)
```

```
           [,1]
[1,] -0.7771982
```

sapply wendet eine Funktion auf jedes Element eines Vektors an. In einer Formel sorgt I() dafür, dass ein Ausdruck nicht als Interaktion o.ä. nterpretiert wird.

```
noHitest = sapply(str, function(x) {
    mEffstr(x, 0)
})
Hitest = sapply(str, function(x) {
    mEffstr(x, 1)
})
plot(noHitest ~ str, ylim = c(-6, 6))
points(Hitest ~ str, col = "red")
abline(h = 0)
legend("bottomright", c("noHitest", "Hitest"), pch = 1,
    col = c("black", "red"))
```



5.3.1 Nichtlineare Interaktionsterme

- lineare Interaktion zwischen `str` und `elpct` oder `str` und `Hiel`: kein signifikanter Effekt
- signifikanter nichtlinearer Effekt von `str` auf `testscr`
- signifikante nichtlinearer Interaktion von `str` und `Hiel`
- Effekt einer Änderung der Klassengröße: „es hängt davon ab“

5.3.2 Zusammenfassung

- nichtlineare Transformationen (log, Polynome) erlauben uns nichtlineare Modelle als multiple Regressionen aufzuschreiben.
- Schätzung läuft wie bei OLS ab.
- Interpretation der Koeffizienten muss die Transformation berücksichtigen.
- Sehr viele nichtlineare Spezifikationen sind möglich. Berücksichtige...
  - Welche nichtlinearen Effekte interessieren uns?
  - Was ist in unserem Problem sinnvoll?

6 Bewertung von Multiplen Regressionsanalysen

6.1 Einführung

↑ Stärken von multiplen Regressionsmodellen

↓ Probleme

- z.B.: Was können wir wirklich über den Effekt von Klassengröße auf Testscores sagen?

6.1.1 Können wir multiple Regressionsanalysen systematisch bewerten?

Vorteile (im Gegensatz zur einfachen Regression):

- Marginale Effekte $\Delta X \to \Delta Y$ können geschätzt werden.
- Omitted Variable Bias kann eventuell verhindert werden (falls die Variable gemessen werden kann)
- Nichlineare Effekte (die von X abhängen) können untersucht werden.

$\to$ dennoch: OLS kann ein verzerrter Schätzer des wahren Effektes sein.

6.1.2 Interne und Externe Validität

Interne Validität statistische Schlüsse über kausale Zusammenhänge gelten für die Population die wir untersuchen.

- Der Schätzer ist unverzerrt und konsistent.
- Hypothesentests haben die gewünschten Signifikanzniveaus und Konfidenzintervalle haben die gewünschten Konfizenzniveaus.

Externe Validität statistische Schlüsse über kausale Zusammenhänge können für andere Population und andere Rahmenbedingungen verallgemeinert werden.

Wie weit können unsere Ergebnisse über Kalifornische Schulen verallgemeinert werden?

- Unterschiedliche Populationen
  - Kalifornien 1998/99
  - Massachusetts 1997/98
  - Mexiko 1997/98

  z.B. beim Test von Arzneimitteln: Verallgemeinerung von Ratten auf Menschen.
- Unterschiedliche Rahmenbedingungen
  - Gesetzliche Rahmenbedingungen von Fördermaßnahmen
  - Unterschiedlicher Umgang mit Zweisprachiger Ausbildung
  - Unterschiedliche Lehrercharacteristika

→ Test von externer Validität durch Vergleich verschiedener Populationen und Rahmenbedingungen.

6.2 Probleme für interne Validität

- Omitted variable bias
- Falsche funktionale Form
- Fehler in den Variablen
- Selection bias
- Simultane Kausalität
- Heteroskedastizität und Korrelation der Fehlerterme:

→ $E(u_i|X_i) \neq 0$, OLS ist verzerrt und nicht konsistent.

6.2.1 Omitted Variable Bias

- Eine Variable beeinflusst Y
- Eine Variable ist korreliert mit einer erklärenden Variablen X

$$\mathrm{E}(b_1) = \beta_1 + (X_1'X_1)^{-1}X_1'X_2\beta_2$$

Abhilfe:

- Wenn die Variable gemessen werden kann $\rightarrow$ in die Regression aufnehmen.
  - Entscheidende Koeffizienten identifizieren (a priori)
  - Aktive Suche nach Quellen von omitted variable bias (a priori)

  $\rightarrow$ Basisspezifikation
  - Erweiterung der Basisspezifikation durch weitere Variablen
    - Test ob die geschätzten Koeffizienten Null sind.
    - Änderen sich die bereits geschätzten Koeffizienten wenn man eine Variable hinzufügt?
  - Überblick über die verschiedenen geschätzten Spezifikationen.

```
data(Caschool)
attach(Caschool)
est1 <- lm(testscr ~ str)
est2 <- lm(testscr ~ str + elpct)
est3 <- lm(testscr ~ str + elpct + mealpct)
est4 <- lm(testscr ~ str + elpct + calwpct)
est5 <- lm(testscr ~ str + elpct + mealpct + calwpct)
```

```
mtable(`(1)` = est1, `(2)` = est2, `(3)` = est3, `(4)` = est4,
    `(5)` = est5, summary.stats = c("R-squared", "N"))
```

	(1)	(2)	(3)	(4)	(5)
(Intercept)	698.933***	686.032***	700.150***	697.999***	700.392***
	(10.461)	(8.812)	(5.641)	(7.006)	(5.615)
str	−2.280***	−1.101*	−0.998***	−1.308***	−1.014***
	(0.524)	(0.437)	(0.274)	(0.343)	(0.273)
elpct		−0.650***	−0.122***	−0.488***	−0.130***
		(0.031)	(0.033)	(0.030)	(0.037)
mealpct			−0.547***		−0.529***
			(0.024)		(0.039)
calwpct				−0.790***	−0.048
				(0.070)	(0.062)
R-squared	0.051	0.426	0.775	0.629	0.775
N	420	420	420	420	420

- Wenn die Variable nicht gemessen werden kann:
  - Falls sich die Variable im Zeitverlauf nicht ändert → Regression mit Paneldaten
  - Falls die Variable mit einer anderen Variable, die man messen kann, korreliert ist → Regression mit Instrumenten.
  - Randomisiertes kontrolliertes Experiment um den Effekt der Variable im Mittel auszuschließen (wenn X zufällig ist, dann ist X insbesondere unabhängig von u, also $E(u|X = x) = 0$)

6.2.2 Misspezifikation der funktionalen Form

- Interaktionsterme einschließen.
- Logarithmische/Polynomiale Spezifikation
- Bei einer diskreten (z.B. binären) abhängigen Variablen: Erweiterung von multiplen Regressionsmodellen (probit, logit)

6.2.3 Fehler in den Variablen

Was ist, wenn wir unser X nicht genau messen können:

- Eingabefehler

- Ungenaue Erinnerung (wann haben Sie mit Ihrem aktuellen Job angefangen?)
- Unklare Fragen (was war Ihr Einkommen im vergangenen Jahr?)
- Bewusst falsche Antworten (Alkoholkonsum / Sexualverhalten)

Beispiel: Sei die wahre Spezifikation

$$Y_i = \beta_0 + \beta_1 X_i + u_i$$

für diese Spezifikation gilt $E(u_i|X_i) = 0$.
Sei X_i der wahre Wert von X und $\tilde{X}_i$ der ungenau gemessene Wert von X.
wir schätzen nun

$$\begin{aligned} Y_i &= \beta_0 + \beta_1 X_i + u_i \\ &= \beta_0 + \beta_1 \tilde{X}_i + \left(\beta_1 (X_i - \tilde{X}_i) + u_i\right) \\ &= \beta_0 + \beta_1 \tilde{X}_i + v_i \end{aligned}$$

wobei $v_i = \beta_1 (X_i - \tilde{X}_i) + u_i$. Wenn nun $(X_i - \tilde{X}_i)$ korreliert ist mit $\tilde{X}_i$, dann ist $\tilde{X}_i$ korreliert mit v_i, und $\hat{\beta}_1$ ist verzerrt und inkonsistent.
Beispiel: Sei $\tilde{X}_i = X_i + w_i$ wobei w_i eine Zufallsvariable mit Mittelwert 0 und Varianz σ_w^2 und unkorreliert mit X_i und u_i.

$$\begin{aligned} v_i &= \beta_1 (X_i - \tilde{X}_i) + u_i \\ &= \beta_1 (X_i - X_i - w_i) + u_i \\ &= -\beta_1 w_i + u_i \end{aligned}$$

Nach Annahme

$$\begin{aligned} \operatorname{cov}(X_i, u_i) &= 0 \\ \operatorname{cov}(\tilde{X}_i, w_i) &= \operatorname{cov}(X_i + w_i, w_i) \\ &= \sigma_w^2 \\ \text{also } \operatorname{cov}(\tilde{X}_i, v_i) &= -\beta_1 \operatorname{cov}(\tilde{X}_i, w_i) + \operatorname{cov}(\tilde{X}_i, u_i) \\ &= -\beta_1 \sigma_w^2 \end{aligned}$$

Wir erinnern uns an

$$\begin{aligned}
\hat{\beta}_1 &= \beta_1 + \frac{\frac{1}{n}\sum_{i=1}^{n}(X_i - \bar{X})u_i}{\frac{1}{n}\sum_{i=1}^{n}(X_i - \bar{X})^2} \\
&\xrightarrow{p} \beta_1 + \frac{\operatorname{cov}(u_i, X_i)}{\sigma_X^2} \\
&= \beta_1 + \frac{\rho_{Xu}\sigma_u\sigma_X}{\sigma_X^2} \\
&= \beta_1 + \rho_{Xu}\frac{\sigma_u}{\sigma_X}
\end{aligned}$$

In unserem Beispiel

$$\begin{aligned}
\hat{\beta}_1 &= \beta_1 + \frac{\operatorname{cov}(u_i, X_i)}{\sigma_{\tilde{X}}^2} \\
&= \beta_1 - \frac{\beta_1\sigma_w^2}{\sigma_X^2 + \sigma_w^2} = \beta_1\frac{\sigma_X^2 + \sigma_w^2 - \sigma_w^2}{\sigma_X^2 + \sigma_w^2} \\
&= \beta_1\frac{\sigma_X^2}{\sigma_X^2 + \sigma_w^2}
\end{aligned}$$

Da $\frac{\sigma_X^2}{\sigma_X^2+\sigma_w^2} < 1$ wird $\hat{\beta}_1$ in Richtung Null verzerrt sein.
Extremfall 1: w_i ist so groß, dass $\tilde{X}_i$ praktisch keine Information mehr enthält: $\hat{\beta}_1 \to 0$ Extremfall 2: $w_i = 0$: $\hat{\beta}_1 \to \beta_1$
Abhilfe bei Fehlern in den Variablen:

- X genauer messen.
- Falls das nicht möglich ist, aber eine andere Variable existiert, die mit X_i korreliert ist und nicht mit u_i korreliert ist (Instrument), dann kann man eine Regression mit Instrumenten schätzen.
- Alternativ: Modell des Messfehlers entwickeln, und dieses zur Korrektur verwenden (z.B. auf Basis von σ_w^2 und σ_X^2)

6.2.4 Sample selection bias

Was ist, wenn die Auswahl der Daten (sampling) beeinflusst wird von der abhängigen Variablen?

- Zufällige Ziehungen vermeiden sampling bias.
- Beispiel: Regression von Lohn (von Beschäftigten) auf Ausbildung. Arbeitslosen werden nicht gezogen. $E(u_i) > 0$
- Beispiel: Performance von Aktienfonds.
  - Ziehe heute zufällig 100 Aktienfonds und betrachte die durchschnittliche Performance über die vergangenen 10 Jahre.

  $\rightarrow$ Performance wird überschätzt.
  - Ziehe vor 10 Jahren zufällig 100 Aktienfonds und betrachte die durchschnittliche Performance über die vergangenen 10 Jahre.
    * Aktienfonds schlagen nicht den Markt.
    * Vergangene gute Performance erklärt nicht zukünftige Performance.

6.2.5 Simultane Kausalität

Kausalität geht in beide Richtungen, von $X \rightarrow Y$, aber auch von $Y \rightarrow X$. Beispiel: Wie wäre es, wenn im `testscr`-Beispiel die Regierung Schulen mit schlechten Scores unterstützt, so dass diese mehr Lehrer einstellen können.

$$\begin{aligned} \texttt{testscr} &\overset{-}{\leftarrow} \texttt{str} \\ \texttt{testscr} &\overset{+}{\rightarrow} \texttt{str} \end{aligned}$$

$$\begin{aligned} Y_i &= \beta_0 + \beta_1 X_i + u_i \\ X_i &= \gamma_0 + \gamma_1 Y_i + \nu_i \end{aligned}$$

- Problem: X_i ist jetzt korreliert mit dem Störterm u_i

Abhilfe:

- Regression mit Instrumenten
- Randomisiertes kontrolliertes Experiment

6.2.6 Heteroskedastizität und Korrelation der Fehlerterme

- Heteroskedastizitat-robuste Standardfehler
- Korrelation der Fehlerterme über Beobachtungen
  - Passiert nicht, wenn Beobachtungen zufällig gezogen werden.
  - Passiert aber im Panel (die selbe Beobachtungseinheit wird im Zeitverlauf mehrmals gezogen) und in Zeitreihen $\rightarrow$ serielle Korrelation
  - Geographischer Einfluss

$\rightarrow$ OLS ist immer noch konsistent, aber die Schätzer der OLS Standardfehler sind nicht mehr konsistent.

$\rightarrow$ alternative Formel für Standardfehler für Panel Daten, Zeitreihendaten, Daten die in Gruppen korreliert sind.

6.3 OLS und Vorhersage

- Unverzerrte Schätzung von $\hat{\beta}$. Beispiel: Was passiert mit `testscr` wenn `str` um 2 Einheiten verringert wird.
- Unverzerrte Schätzung von $\widehat{Y}$. Wie groß ist `testscr` etwa in einem Distrikt mit `str`=20?

$$\texttt{testscr} = 698.933 - 2.2798 \cdot \texttt{str}$$

Wir wissen, der Koeffizient von `str` ist verzerrt. Für die Prognose ist das aber belanglos.

  - R^2 ist jetzt wichtig
  - Omitted variable bias ist kein Problem mehr

- Interpretation der Koeffizienten ist nicht wichtig, was zählt ist nur ein guter „fit“.
- Externe Validität ist wichtig: Das mit vergangenen Daten geschätzte Modell muss auch in der Zukunft gelten.

6.4 Vergleich von Caschool mit MCAS

distcod	disctric code
county	county
district	district
grspan	grade span of district
enrltot	total enrollment
teachers	number of teachers
calwpct	percent qualifying for CalWorks
mealpct	percent qualifying for reduced-price lunch
computer	number of computers
testscr	average test score (read.scr+math.scr)/2
compstu	computer per student
expnstu	expenditure per student
str	student teacher ratio
avginc	district average income
elpct	percent of English learners
readscr	average reading score
mathscr	average math score

Source: California Department of Education

code	district code (numerical)
municipa	municipality (name)
district	district name
regday	spending per pupil, regular
specneed	spending per pupil, special needs
bilingua	spending per pupil, bilingual
occupday	spending per pupil, occupational
totday	spending per pupil, total
spc	students per computer
speced	special education students
lnchpct	eligible for free or reduced price lunch
tchratio	students per teacher
percap	per capita income
totsc4	4th grade score (math+english+science)
totsc8	8th grade score (math+english+science)
avgsalary	average teacher salary
pctel	percent english learners

Source: Massachusetts Comprehensive Assessment System (MCAS), Massachusetts Department of Education, 1990 U.S. Census

Die Schreibweise `Datensatz$variable` bezeichnet eine Variable (Spalte) aus einem Datensatz. Alternativ kann man auch `Datensatz[,"variable"]` sagen, bzw. für mehrere Spalten `Datensatz[,c("variable1","variable2")]`.

```
data(MCAS)
Caschool$type = "CA"
Caschool
```

```
    distcod          county                                    district
1     75119         Alameda                        Sunol Glen Unified
    grspan enrltot  teachers calwpct  mealpct computer testscr
1    KK-08     195   10.9000  0.5102   2.0408       67  690.80
       compstu  expnstu      str    avginc       elpct readscr
1   0.34358975 6384.911 17.88991 22.690001  0.00000000   691.6
    mathscr type
1     690.0   CA
 [ reached getOption("max.print") -- omitted 419 rows ]]
```

```
MCAS$type = "MA"
MCAS$str = MCAS$tchratio
MCAS$testscr = MCAS$totsc4
MCAS$elpct = MCAS$pctel
MCAS$avginc = MCAS$percap
MCAS$mealpct = MCAS$lnchpct
Caschool[, c("type", "str", "testscr", "elpct", "avginc",
    "mealpct")]
```

```
    type      str testscr       elpct    avginc  mealpct
1     CA 17.88991  690.80  0.00000000 22.690001   2.0408
2     CA 21.52466  661.20  4.58333349  9.824000  47.9167
3     CA 18.69723  643.60 30.00000191  8.978000  76.3226
4     CA 17.35714  647.70  0.00000000  8.978000  77.0492
 [ reached getOption("max.print") -- omitted 416 rows ]]
```

```
MCAS[, c("type", "str", "testscr", "elpct", "avginc",
    "mealpct")]
```

```
    type  str testscr      elpct avginc mealpct
1     MA 19.0     714  0.0000000 16.379    11.8
2     MA 22.6     731  1.2461059 25.792     2.5
3     MA 19.3     704  0.0000000 14.040    14.1
4     MA 17.9     704  0.3225806 16.111    12.1
 [ reached getOption("max.print") -- omitted 216 rows ]]
```

Das Kommando merge vereinigt zwei Datensätze. Z.B. enthalte ein Datensatz für jeden Studenten eine Matrikelnummer und die Note, ein anderer Datensatz die Matrikelnummer und den Namen. Ein merge sorgt dafür, dass neben jeder Matrikelnummer die passende Note und der passende Name steht. Wenn es nicht gemeinsames gibt, dann kann merge auch die nicht zusammenpassenden Datensätze hintereinanderfügen.

```
cama = merge(Caschool[, c("type", "str", "testscr", "elpct",
    "avginc", "mealpct")], MCAS[, c("type", "str", "testscr",
    "elpct", "avginc", "mealpct")], all = TRUE)
cama[1:10, ]
```

```
   type      str testscr     elpct avginc mealpct
1    CA 14.00000  635.60  0.000000 10.656 68.8235
2    CA 14.20176  656.50  0.000000 13.712 20.0000
3    CA 14.54214  695.30  3.765690 35.342  0.0000
4    CA 14.70588  666.85  2.500000 11.826 53.5032
 [ reached getOption("max.print") -- omitted 6 rows ]]
```

```
cama[500:510, ]
```

```
    type  str testscr     elpct avginc mealpct
500   MA 16.6     735 0.0000000 16.714     9.4
501   MA 16.6     738 0.0000000 26.103     1.7
502   MA 16.7     682 3.6803365 10.966    51.4
503   MA 16.7     692 0.0000000 15.100    13.8
 [ reached getOption("max.print") -- omitted 7 rows ]]
```

aggregate hilft uns, einen Datensatz in Teilmengen zu zerlegen und für diese Teilmengen jeweils eine Funktion auszuführen.

```
aggregate(cama[, 2:6], list(cama$type), mean)
```

```
  Group.1      str  testscr     elpct   avginc  mealpct
1      CA 19.64043 654.1565 15.768155 15.31659 44.70524
2      MA 17.34409 709.8273  1.117676 18.74676 15.31591
```

```
aggregate(cama[, 2:6], list(cama$type), sd)
```

```
  Group.1      str  testscr    elpct   avginc  mealpct
1      CA 1.891812 19.05335 18.28593 7.225890 27.12338
2      MA 2.276666 15.12647  2.90094 5.807637 15.06007
```

```
aggregate(cama[, 2:6], list(cama$type), length)
```

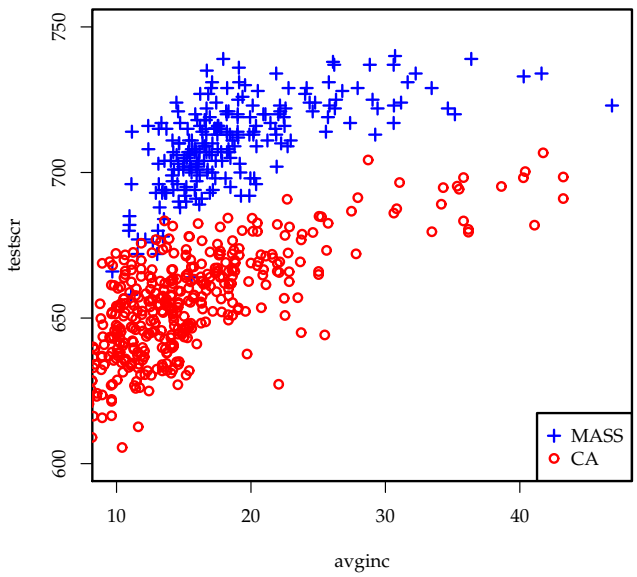
```
  Group.1 str testscr elpct avginc mealpct
1      CA 420     420   420    420     420
2      MA 220     220   220    220     220
```

Die Funktion `subset` wählt eine Teilmenge aus einem Datensatz aus. Wenn eine Funktion den Parameter `data` unterstützt, kann man dort eine passende Teilmenge übergeben. Viele Funktionen haben außerdem einen Parameter `subset` der gleich eine Teilmenge auswählt. `ylim` definiert den Maßstab der y-Achse, und `pch` definiert, welches Symbol benutzt wird, um einen Punkt darzustellen.

```
attach(cama)
```

```
       The following object(s) are masked from Caschool :

        avginc elpct mealpct str testscr
```

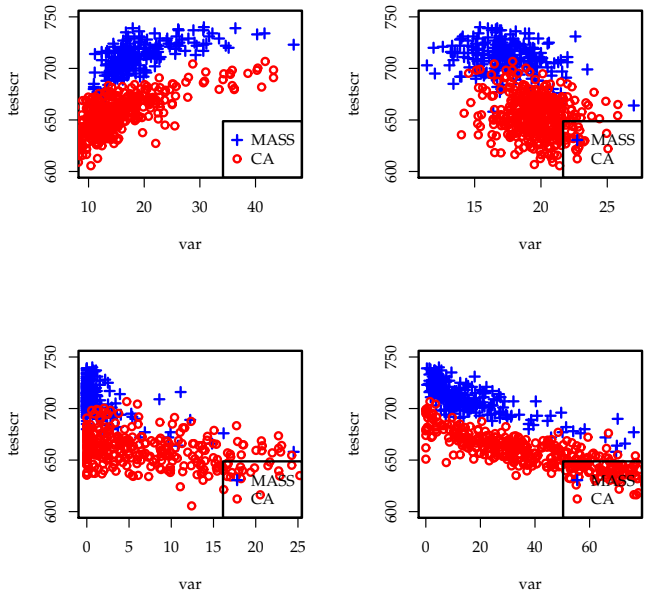
```
plot(testscr ~ avginc, subset = (type == "MA"), col = "blue",
    pch = 3, ylim = c(600, 750))
points(testscr ~ avginc, subset = (type == "CA"), col = "red")
legend("bottomright", c("MASS", "CA"), pch = c(3, 1),
    col = c("blue", "red"))
```



Um mehrere Plots dieser Art zu machen schreiben wir eine kleine Funktion.

```
myPlot <- function(var) {
    plot(testscr ~ var, subset = (type == "MA"), ylim = c(600,
        750), col = "blue", pch = 3)
    points(testscr ~ var, subset = (type == "CA"), col = "red",
        pch = 1)
    legend("bottomright", c("MASS", "CA"), pch = c(3,
        1), col = c("blue", "red"))
}
```

```
par(mfrow = c(2, 2))
myPlot(avginc)
myPlot(str)
myPlot(elpct)
myPlot(mealpct)
```



Testscores und Einkommen in Massachusetts

```
(estC <- lm(testscr ~ avginc, data = subset(cama, type ==
    "CA")))
```

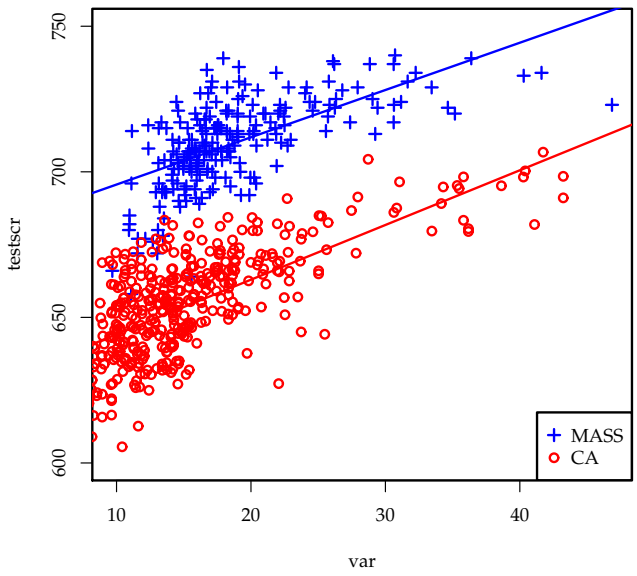
```
Call:
lm(formula = testscr ~ avginc, data = subset(cama, type == "CA"))

Coefficients:
```

```
(Intercept)       avginc
    625.384        1.879
```

```
(estM <- lm(testscr ~ avginc, data = subset(cama, type ==
    "MA")))
```

```
Call:
lm(formula = testscr ~ avginc, data = subset(cama, type == "MA"))

Coefficients:
(Intercept)       avginc
    679.387        1.624
```

```
myPlot(avginc)
abline(estC, col = "red")
abline(estM, col = "blue")
```



Wenn eine Funktion bei der Definition den Parameter ... hat, kann man beim Funktionaufruf weitere Parameter übergeben, die dann, wann immer man in der Funktion ... sagt, eingesetzt werden.

```
ePlot <- function(model, data, ...) {
    est <- lm(model, data)
    stdev <- sqrt(diag(hccm(est)))
    pvalue <- round(2 * pnorm(-abs(coef(est)/stdev)),
        4)
    stern <- ifelse(pvalue < 0.001, "***", ifelse(pvalue <
        0.01, "**", ifelse(pvalue < 0.05, "*", ifelse(pvalue <
        0.1, ".", ""))))
    a <- as.data.frame(cbind(coef(est), stdev, pvalue))
    a$stern = stern
    colnames(a)[1] = "beta"
    print(a, digits = 3)
    sum <- summary(est)
    cat("R2=        ", round(sum$r.squared, 2), "\n")
    or <- order(data$avginc)
    if (substr(model[2], 1, 4) == "log(") {
        lines(data$avginc[or], exp(fitted(est)[or]),
            ...)
    }
    else lines(data$avginc[or], fitted(est)[or], ...)
    est
}
```

```
myPlot(avginc)
est <- ePlot(testscr ~ avginc + I(avginc^2), data = subset(cama,
    type == "CA"))
```

```
               beta   stdev pvalue stern
(Intercept) 607.3017 2.92422      0   ***
avginc        3.8510 0.27110      0   ***
I(avginc^2)  -0.0423 0.00488      0   ***
R2=         0.56
```

```
est <- ePlot(testscr ~ avginc + I(avginc^2), data = subset(cama,
    type == "MA"))
```

```
               beta  stdev pvalue stern
(Intercept) 638.3711 8.0401      0   ***
avginc        5.4703 0.6893      0   ***
```

```
I(avginc^2)  -0.0808 0.0136      0   ***
R2=          0.48
```

```
est <- ePlot(testscr ~ avginc + I(avginc^2) + I(avginc^3),
    data = subset(cama, type == "CA"), col = "red")
```

```
                 beta    stdev pvalue stern
(Intercept) 600.078985 5.462310 0.0000   ***
avginc        5.018677 0.787290 0.0000   ***
I(avginc^2)  -0.095805 0.034052 0.0049    **
I(avginc^3)   0.000685 0.000437 0.1167
R2=          0.56
```

```
est <- ePlot(testscr ~ avginc + I(avginc^2) + I(avginc^3),
    data = subset(cama, type == "MA"), col = "red")
```

```
                beta    stdev pvalue stern
(Intercept) 600.39853 26.96057 0.0000   ***
avginc       10.63538  3.63075 0.0034    **
I(avginc^2)  -0.29689  0.15614 0.0572     .
I(avginc^3)   0.00276  0.00214 0.1968
R2=          0.49
```

```
est <- ePlot(testscr ~ log(avginc), data = subset(cama,
    type == "CA"), col = "blue")
```

```
            beta stdev pvalue stern
(Intercept) 557.8  3.86      0   ***
log(avginc)  36.4  1.41      0   ***
R2=          0.56
```

```
est <- ePlot(testscr ~ log(avginc), data = subset(cama,
    type == "MA"), col = "blue")
```

```
            beta stdev pvalue stern
(Intercept) 600.8  9.36      0   ***
log(avginc)  37.7  3.17      0   ***
R2=          0.46
```

```
est <- ePlot(log(testscr) ~ log(avginc), data = subset(cama,
    type == "CA"), col = "green")
```

```
              beta   stdev pvalue stern
(Intercept) 6.3363 0.00596      0   ***
log(avginc) 0.0554 0.00216      0   ***
R2=         0.56
```

```
est <- ePlot(log(testscr) ~ log(avginc), data = subset(cama,
    type == "MA"), col = "green")
```

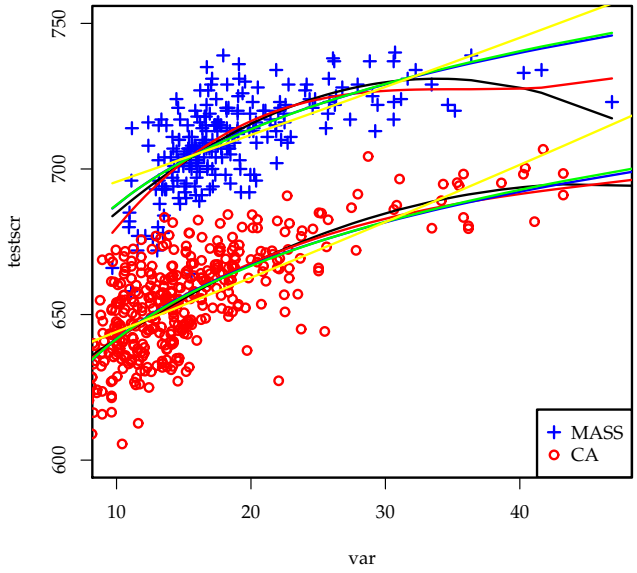
```
              beta   stdev pvalue stern
(Intercept) 6.4107 0.01343      0   ***
log(avginc) 0.0533 0.00454      0   ***
R2=         0.46
```

```
est <- ePlot(log(testscr) ~ avginc, data = subset(cama,
    type == "CA"), col = "yellow")
```

```
              beta    stdev pvalue stern
(Intercept) 6.43936 0.002987      0   ***
avginc      0.00284 0.000183      0   ***
R2=         0.5
```

```
options(scipen = 5)
est <- ePlot(log(testscr) ~ avginc, data = subset(cama,
    type == "MA"), col = "yellow")
```

```
              beta    stdev pvalue stern
(Intercept) 6.52186 0.005388      0   ***
avginc      0.00229 0.000276      0   ***
R2=         0.38
```



Multiple regression:

```
par(mfrow = c(1, 2))
myPlot(avginc)
est <- ePlot(testscr ~ str, data = subset(cama, type ==
    "CA"))
```

```
              beta  stdev pvalue stern
(Intercept) 698.93 10.461      0   ***
str          -2.28  0.524      0   ***
R2=          0.05
```

```
est <- ePlot(testscr ~ str, data = subset(cama, type ==
    "MA"))
```

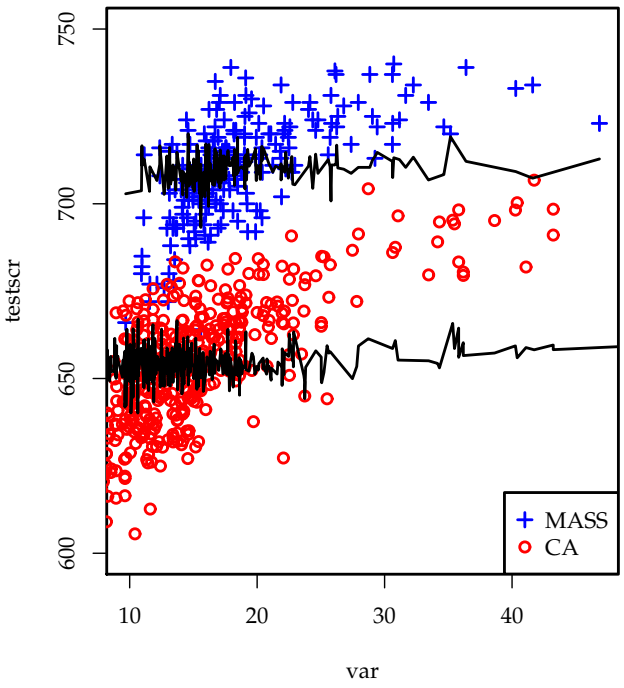
```
              beta stdev pvalue stern
(Intercept) 739.62 8.882 0.0000   ***
str          -1.72 0.516 0.0009   ***
R2=          0.07
```

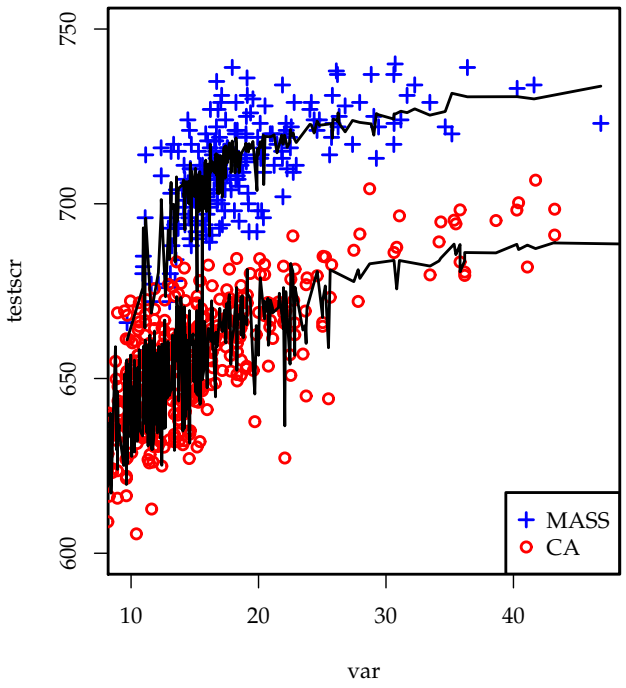
```
myPlot(avginc)
est <- ePlot(testscr ~ str + elpct + mealpct + log(avginc),
    data = subset(cama, type == "CA"))
```

```
               beta  stdev pvalue stern
(Intercept) 658.552 8.7489 0.0000   ***
str          -0.734 0.2606 0.0048    **
elpct        -0.176 0.0342 0.0000   ***
mealpct      -0.398 0.0336 0.0000   ***
log(avginc)  11.569 1.8413 0.0000   ***
R2=          0.8
```

```
est <- ePlot(testscr ~ str + elpct + mealpct + log(avginc),
    data = subset(cama, type == "MA"))
```

```
               beta   stdev pvalue stern
(Intercept) 682.432 12.0943 0.0000   ***
str          -0.689  0.2779 0.0131     *
elpct        -0.411  0.3512 0.2422
mealpct      -0.521  0.0834 0.0000   ***
log(avginc)  16.529  3.3010 0.0000   ***
R2=          0.68
```





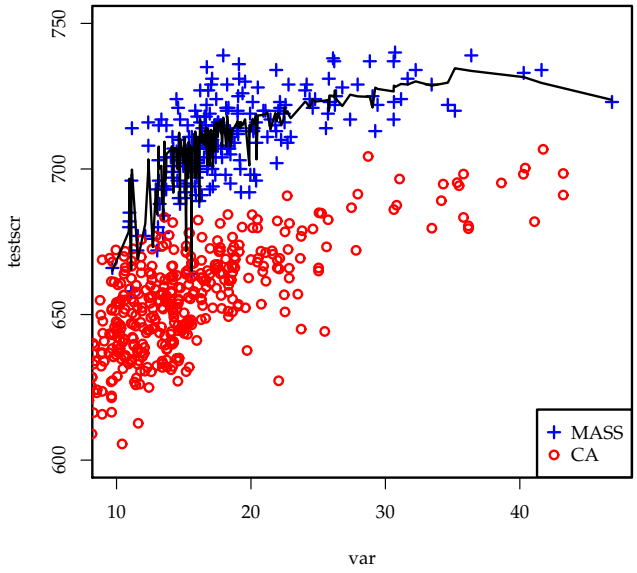
- Der Effekt von str ist signifikant.
- Zusätzliche Variablen reduzieren den Koeffizienten von str
- Der Effekt von avginc ist signifikant.

```
myPlot(avginc)
est <- ePlot(testscr ~ str + elpct + mealpct + avginc +
    I(avginc^2) + I(avginc^3), data = subset(cama, type ==
    "MA"))
```

```
                beta    stdev pvalue stern
(Intercept) 744.02504 23.18585 0.0000   ***
str          -0.64091  0.27642 0.0204     *
elpct        -0.43712  0.35908 0.2235
mealpct      -0.58182  0.10781 0.0000   ***
avginc       -3.06669  2.53398 0.2262
I(avginc^2)   0.16369  0.09172 0.0743     .
 [ reached getOption("max.print") -- omitted last row ]]
R2=         0.69
```

```
linear.hypothesis(est, c("I(avginc^2)=0", "I(avginc^3)=0"),
    vcov = hccm)
```

```
Linear hypothesis test

Hypothesis:
I(avginc^2) = 0
I(avginc^3) = 0

Model 1: testscr ~ str + elpct + mealpct + avginc + I(avginc^2) + I(avginc^3)
Model 2: restricted model

Note: Coefficient covariance matrix supplied.

  Res.Df  Df     F   Pr(>F)
1    213
2    215  -2 6.227 0.002354 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```



- Der Effekt von `str` ist signifikant.

```
myPlot(avginc)
est <- ePlot(testscr ~ str + I(str^2) + I(str^3) + elpct +
    mealpct + avginc + I(avginc^2) + I(avginc^3), data = subset(cama,
    type == "CA"))
```

```
                beta      stdev pvalue stern
(Intercept) 330.079169 173.293815 0.0568     .
str          55.618325  26.486559 0.0357     *
I(str^2)     -2.914810   1.340721 0.0297     *
I(str^3)      0.049866   0.022437 0.0262     *
elpct        -0.196440   0.035054 0.0000   ***
mealpct      -0.411538   0.033874 0.0000   ***
 [ reached getOption("max.print") -- omitted 3 rows ]]
R2=          0.81
```

```
est <- ePlot(testscr ~ str + I(str^2) + I(str^3) + elpct +
    mealpct + avginc + I(avginc^2) + I(avginc^3), data = subset(cama,
    type == "MA"))
```

```
                beta     stdev pvalue stern
(Intercept) 665.49605 116.07834 0.0000   ***
```

```
str           12.42598  20.27945 0.5401
I(str^2)     -0.68030   1.12956 0.5470
I(str^3)      0.01147   0.02081 0.5814
elpct        -0.43417   0.36722 0.2371
mealpct      -0.58722   0.11724 0.0000   ***
 [ reached getOption("max.print") -- omitted 3 rows ]]
R2=         0.69
```

```
linear.hypothesis(est, c("str=0", "I(str^2)", "I(str^3)"),
    vcov = hccm)
```

```
Linear hypothesis test

Hypothesis:
str = 0
I(str^2) = 0
I(str^3) = 0

Model 1: testscr ~ str + I(str^2) + I(str^3) + elpct + mealpct + avginc +
    I(avginc^2) + I(avginc^3)
Model 2: restricted model

Note: Coefficient covariance matrix supplied.

  Res.Df  Df      F  Pr(>F)
1    211
2    214  -3 2.3364 0.07478 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
linear.hypothesis(est, c("I(str^2)=0", "I(str^3)=0"),
    vcov = hccm)
```

```
Linear hypothesis test

Hypothesis:
I(str^2) = 0
I(str^3) = 0
```

```
Model 1: testscr ~ str + I(str^2) + I(str^3) + elpct + mealpct + avginc +
    I(avginc^2) + I(avginc^3)
Model 2: restricted model

Note: Coefficient covariance matrix supplied.

  Res.Df  Df      F Pr(>F)
1    211
2    213  -2 0.3396 0.7124
```

```
linear.hypothesis(est, c("I(avginc^2)=0", "I(avginc^3)=0"),
    vcov = hccm)
```

```
Linear hypothesis test

Hypothesis:
I(avginc^2) = 0
I(avginc^3) = 0

Model 1: testscr ~ str + I(str^2) + I(str^3) + elpct + mealpct + avginc +
    I(avginc^2) + I(avginc^3)
Model 2: restricted model

Note: Coefficient covariance matrix supplied.

  Res.Df  Df      F   Pr(>F)
1    211
2    213  -2 5.7043 0.003866 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- Der Effekt von `str` ist signifikant.
- Es gibt einen signifikanten nichtlinearen Effekt von `str` in Kalifornien, nicht aber in Massachusetts.

```
aggregate(elpct, list(type), median)
```

```
  Group.1        x
1      CA 8.777634
```

```
2      MA 0.000000
```

```
cama$HiEL = cama$elpct > 0
est <- ePlot(testscr ~ str + HiEL + HiEL:str + mealpct +
    avginc + I(avginc^2) + I(avginc^3), data = subset(cama,
    type == "CA"))
```

```
                beta     stdev pvalue stern
(Intercept)  658.16110 16.404309 0.0000   ***
str            1.35471  0.810345 0.0946     .
HiELTRUE      36.11583 16.262280 0.0264     *
mealpct       -0.50670  0.027027 0.0000   ***
avginc        -0.90724  0.616350 0.1410
I(avginc^2)    0.05912  0.023531 0.0120     *
 [ reached getOption("max.print") -- omitted 2 rows ]]
R2=         0.8
```

```
est <- ePlot(testscr ~ str + HiEL + HiEL:str + mealpct +
    avginc + I(avginc^2) + I(avginc^3), data = subset(cama,
    type == "MA"))
```

```
                beta    stdev pvalue stern
(Intercept)  759.91422 25.28938 0.0000   ***
str           -1.01768  0.38182 0.0077    **
HiELTRUE     -12.56073 10.22789 0.2194
mealpct       -0.70851  0.09894 0.0000   ***
avginc        -3.86651  2.71955 0.1551
I(avginc^2)    0.18412  0.09930 0.0637     .
 [ reached getOption("max.print") -- omitted 2 rows ]]
R2=         0.69
```

```
linear.hypothesis(est, c("str=0", "str:HiELTRUE=0"),
    vcov = hccm)
```

```
Linear hypothesis test

Hypothesis:
str = 0
```

```
str:HiELTRUE = 0

Model 1: testscr ~ str + HiEL + HiEL:str + mealpct + avginc + I(avginc^2) +
    I(avginc^3)
Model 2: restricted model

Note: Coefficient covariance matrix supplied.

  Res.Df  Df      F Pr(>F)
1    212
2    214  -2 3.7663 0.0247 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
linear.hypothesis(est, c("I(avginc^2)=0", "I(avginc^3)=0"),
    vcov = hccm)
```

```
Linear hypothesis test

Hypothesis:
I(avginc^2) = 0
I(avginc^3) = 0

Model 1: testscr ~ str + HiEL + HiEL:str + mealpct + avginc + I(avginc^2) +
    I(avginc^3)
Model 2: restricted model

Note: Coefficient covariance matrix supplied.

  Res.Df  Df      F  Pr(>F)
1    212
2    214  -2 3.2201 0.04191 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- Der Effekt von `str` ist signifikant.
- Keine nennenswerte Interaktion zwischen `HiEL` und `str` in Massachusetts, wohl aber in Kalifornien.

```
est <- ePlot(testscr ~ str + mealpct + avginc + I(avginc^2) +
    I(avginc^3), data = subset(cama, type == "MA"))
```

```
                beta    stdev pvalue stern
(Intercept) 747.36389 21.67952 0.0000   ***
str          -0.67188  0.27679 0.0152     *
mealpct      -0.65308  0.07859 0.0000   ***
avginc       -3.21795  2.46635 0.1920
I(avginc^2)   0.16479  0.09113 0.0706     .
I(avginc^3)  -0.00216  0.00106 0.0415     *
R2=         0.68
```

```
linear.hypothesis(est, c("I(avginc^2)=0", "I(avginc^3)=0"),
    vcov = hccm)
```

```
Linear hypothesis test

Hypothesis:
I(avginc^2) = 0
I(avginc^3) = 0

Model 1: testscr ~ str + mealpct + avginc + I(avginc^2) + I(avginc^3)
Model 2: restricted model

Note: Coefficient covariance matrix supplied.

  Res.Df  Df      F  Pr(>F)
1    214
2    216  -2 4.2776 0.01508 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- Der Effekt von `str` ist signifikant.

Diskussion:

- Der Datensatz aus Kalifornien ist größer → leichter, signifikante Ergebnisse zu finden.

Vergleich der Mittelwerte und Standardabweichungen für `str` in Kalifornien und Massachusetts:

```
(mmean <- aggregate(cama$testscr, list(type), mean))
```

```
  Group.1        x
1      CA 654.1565
2      MA 709.8273
```

```
colnames(mmean) <- c("type", "testmean")
(msd = aggregate(cama$testscr, list(type), sd))
```

```
  Group.1        x
1      CA 19.05335
2      MA 15.12647
```

```
colnames(msd) <- c("type", "testsd")
merge(cama, mmean)[500:502, ]
```

```
    type  str testscr    elpct avginc mealpct  HiEL testmean
500   MA 16.6     735 0.000000 16.714     9.4 FALSE 709.8273
501   MA 16.6     738 0.000000 26.103     1.7 FALSE 709.8273
502   MA 16.7     682 3.680336 10.966    51.4  TRUE 709.8273
```

```
merge(cama, mmean)[1:3, ]
```

```
  type      str testscr   elpct avginc mealpct  HiEL testmean
1   CA 14.00000   635.6 0.00000 10.656 68.8235 FALSE 654.1565
2   CA 14.20176   656.5 0.00000 13.712 20.0000 FALSE 654.1565
3   CA 14.54214   695.3 3.76569 35.342  0.0000  TRUE 654.1565
```

```
cama2 = merge(merge(cama, mmean), msd)
cama2$testnorm = (cama2$testscr - cama2$testmean)/cama2$testsd
detach(cama)
```

```
        The following object(s) are masked from cama ( position 3 ) :

         avginc elpct mealpct str testscr type
```

```
The following object(s) are masked from cama ( position 4 ) :

 avginc elpct mealpct str testscr type


The following object(s) are masked from Caschool :

 avginc elpct mealpct str testscr
```

```
est <- ePlot(testnorm ~ str * type, data = cama2)
```

```
               beta  stdev pvalue stern
(Intercept)  2.35005 0.5490  0.000   ***
str         -0.11965 0.0275  0.000   ***
typeMA      -0.38040 0.8038  0.636
str:typeMA   0.00609 0.0438  0.889
R2=         0.06
```

```
est <- ePlot(testscr ~ str * type + mealpct + avginc +
    I(avginc^2) + I(avginc^3), data = cama2)
```

```
                beta    stdev pvalue stern
(Intercept) 697.17282 7.309650 0.0000   ***
str          -0.74765 0.264140 0.0046    **
typeMA       38.00572 7.229002 0.0000   ***
mealpct      -0.54206 0.024022 0.0000   ***
avginc       -1.27930 0.587298 0.0294     *
I(avginc^2)   0.07567 0.022051 0.0006   ***
 [ reached getOption("max.print") -- omitted 2 rows ]]
R2=         0.92
```

6.4.1 Interne Validität

- Omitted variables: wir kontrollieren für
  - Einkommen

  - einige Eigenschaften von Studenten (Sprache)

  Was fehlt? z.B. `str` könnte korreliert sein mit

  - Qualität der Lehrer ☹
  - Außerschulische Angebote ☹
  - Aufmerksamkeit der Eltern ☹

  → Alternative: Experiment. Schüler werden zufällig unterschiedlichen Klassengrößen zugelost.

- Funktionale Form:
  - unterschiedliche nicht-lineare Spezifikationen führen im Beispiel zu ähnlichen Ergebnissen. ☑
  - Nichtlinearitäten sind nicht sehr groß ☑
- Fehler in den Variablen:
  - `str` ist ein Durchschnitt über den gesamten Distrikt.
  - Die tatsächliche Varianz des student-teacher-ratios wird möglichweise mit `str` unterschätzt. Dann aber wird auch die Schätzung für den Koeffizienten von `str` unterschätzt. ☹
  - Idealerweise hätten wir gerne Daten für individuelle Studenten.
- Selection bias:
  - Beide Datensätze beruhen auf einer Vollerhebung. ☑
- Simultane Kausalität:
  - `testscr` $\overset{\leftarrow}{\rightarrow}$ `str` (z.B. ausgleichende Maßnahmen)
    * Massachusettes: keine Maßnahmen. ☑
    * Kalifornien: Ausgleichende Finanzierung, aber unabhängig von Erfolgen der Studenten. ☑
- Heteroskedastizität und Korrelation der Fehlerterme:
  - Heteroskedastizität: Robuste Varianz-Kovarianz Matrizen. ☑
  - Korrelation der Fehlerterme: Kein zufälliges Ziehen der Beobachtungen. ☹

6.4.2 Externe Validität

Vergleich Kalifornien $\leftrightarrow$ Massachusetts ☺

6.4.3 Ergebnis

Reduzierung der Klassengröße um eine Einheit $\rightarrow$

- $+$ `testscr` steigt um ca. 0.08 Standardabweichungen
- $-$ Kosten (Klassenräume, Arbeitsstunden für Lehrer)