

Einfache lineare Regression

Es wird nur eine Erklärende-Variable betrachtet

$$y_i = \alpha + \beta x_i$$

$$Y_i = \alpha + \beta x_i + E_i$$

1. α ist der Achsenabschnitt
2. β ist die Steigung
3. E_i ist der Zufallsfehler
4. y_i Ziel-Variablen
5. x_i Erklärende-Variablen
1. Fehler haben Erwartungswert 0
2. Fehler sind normalverteilt
3. Fehler haben alle dieselbe Varianz σ^2 (Konstant)
4. Fehler sind unabhängig voneinander

Vorteil der einfachen Regression:

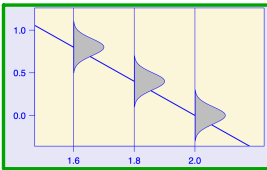
- Graphische Darstellung möglich
- Einfach zu Interpretieren
- Mathematisch einfacher als multiple Regression

Hauptziele:

1. Eine Gerade in eine Punktwolke legen, um die Beziehung zwischen einer Zielgröße Y und einer erklärenden Variablen X zu beschreiben.
2. Bestimmen, ob die Daten mit einem Modell mit vorgegebenen Koeffizienten verträglich sind (statistischer Hypothesentest).
3. Angeben, wie genau der Achsenabschnitt und die Steigung aus den Daten bestimmt werden können (Vertrauensintervall)

z.B.

$$Y_i = 4 - 2x_i + E_i$$



Die Parameter α und β werden mittels Kleinst-Quadrate geschätzt (Die Abstände aller Residuen in der Summe werden minimiert.)

R-Code

```
fit <- lm(Y ~ X) # Erstellung eines Modells
coef(fit) # Ausgabe der geschätzten Koeffizienten  $\hat{\alpha}, \hat{\beta}$ 
summary(fit) # Ausführliche Zusammenfassung
confint(fit) # Vertrauensintervalle (Plausible Bereiche für  $\hat{\beta}$ ...)
```

```
Spr.lm <- lm(lErsch ~ lDist, data=SprengS1)
coef(Spr.lm)
(Intercept) 1Dist
8.9791 -1.9235
```

$$y = \beta_0 + \beta_1 \cdot x_i$$
$$lErsch = 8.9791 + (-1.9235) \cdot lDist$$

- Interpretation:
- Achsenabschnitt (Intercept): Falls die Distanz zwischen Messort und Sprengort gleich 0 ist (d.h. lDist=0), dann ist die logarithmierte Erschütterung (lErsch) gleich 8.9791
 - Steigung: Falls die lDist um eine Einheit zunimmt, dann nimmt die lErsch um (-1.9235) Einheiten ab

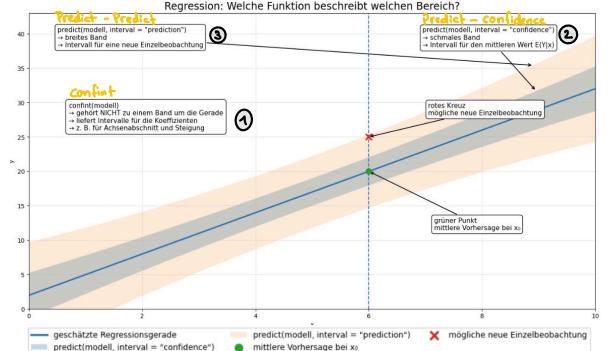
Geschätzt werden:
 $\hat{\alpha}, \hat{\beta}, \hat{\sigma}$

Vertrauensintervall

`confint(...)` ①

Wie gross ist ein Modellparameter wahrscheinlich?

liefert einen Bereich für den wahren Koeffizienten



`predict(..., interval="confidence")` ②

Wo liegt der wahre mittlere Funktionswert bei einem bestimmten x_0 ?

Wie gross ist der erwartete DURCHSCHNITT...

Vertrauensband (Mittelwert)

Enthält nur die Unsicherheit der Schätzung der Geraden

Prognoseintervall

`predict(..., interval="prediction")` ③

Wo liegt eine einzelne zukünftige Beobachtung bei einem bestimmten x_0 ?

Wie gross ist eine EINZELNE neue Beobachtung

Prognoseband (Einzelne Beobachtung)

Enthält sowohl die Unsicherheit der Schätzung als auch die zufälligen Fehler E_i

`predict(object, newdata = ..., interval = ..., level = ...)`

Newdata sind die spezifischen x_0 Werte, welche vorhergesagt werden

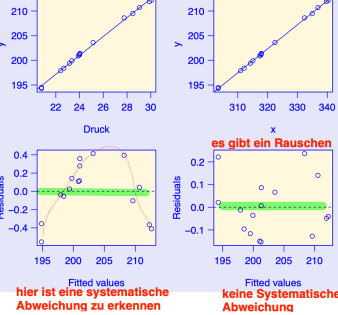
x_0 soll im Intervall der vorhandenen Daten liegen. Sonst wird Extrapoliert (Vorhersage ausserhalb Datenpunkte)

Tukey-Anscombe-Diagramm

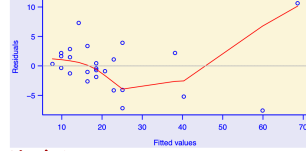
Es wird geprüft, ob $E(E_i) = 0$

die Datenpunkt sollten keine Struktur aufweisen und um die Null-Linie streuen

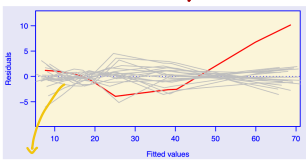
so sieht man fast keinen Unterschied bei diesen beiden Datensätzen



Mit Glättern (Mittelwerte)



Noch immer schwierig zu beurteilen daher mit Bootstrap



Rote Kurve ist atypisch. Liegt also ausserhalb der statistischen Fluktuation

Graue Pfade = statistische Fluktuation (Bootstrap)

Summary output lesen

```
> summary(Spr.lm)

Call:
lm(formula = lErsch ~ lDist, data = SprengS1)

Residuals:
    Min       1Q   Median       3Q      Max
-0.181131 -0.050099 -0.009148  0.076009  0.192842

Coefficients:
(Intercept) 8.9791 0.7267 12.36 8.61e-08 ***
1Dist      -1.9235  0.1783 10.79 3.45e-07 ***

Residual standard error: 0.2636 on 11 degrees of freedom
Multiple R-squared:  0.9130, Adjusted R-squared:  0.9056
F-statistic: 116.4 on 1 and 11 DF, p-value: 3.448e-07
```

Welches Modell wurde angepasst?

Differenzen zwischen beobachtetem Wert (Sind die Residuen gross) (Keine Aussage möglich =>) und vorhergesagtem Wert $r_i = y_i - \hat{y}_i$ (oder klein? Klein = Gut) (Später Residuen-Analyse)

Geschätzte Koeffizienten $\hat{\alpha}, \hat{\beta}$

Standardfehler der geschätzten Koeffizienten (also Genauigkeit) Wie stark streut der Schätzer? Klein = gut (Achtung wird in Einheit ausgedr.)

Ist der Koeffizient = 0?

Je weiter t-value weg von 0 ist, desto stärker spricht es gegen die Nullhypothese

p-value zum jeweiligen t-Test $H_0: \beta = 0$ p-value < 0,05 => Verwerfen => Signifikant p-value > 0,05 => nicht verwerfen => nicht Signifikant

$\hat{\sigma}$: Geschätzte Standardabweichung der Fehler

R^2 Wie gut wird die ZV durch EV ausgedr. Technik / Naturwissenschaften: Werte > 0,9 üblich Geistes / Sozialwissenschaften: Werte um 0,6 i.O. $0 < R^2 < 1$

(Korrelation quadriert (-1,1)) Misst die Stärke des linearen Zusammenhangs. Sagt NICHT darüber aus wie gut das Modell an sich ist.

adj R^2 : Ist ein Mass für die Kombination von Genauigkeit (σ^2) und Modellkomplexität (Anzahl EV)

Globaler Test für das Modell alle Koeffizienten werden auf einmal gegen die Nullhypothese getestet. Relevant bei: "Multiblen linearen Regression"

$$H_0: \beta_1 = \beta_2 = \dots = \beta_m = 0$$

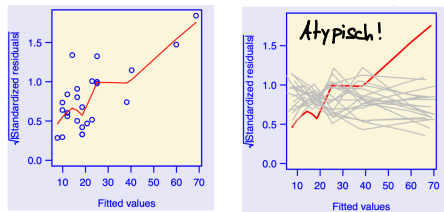
Mindestens ein Koeffizient ist ungleich 0, wenn verworfen wird.

ausser β_0

Alternativ-Hypothese $H_A: \beta \neq 0$

Scale-location plot

Es wird geprüft ob $\text{var}(E_i) = \sigma^2$ ist die Varianz konstant?



Was meistens hilft:

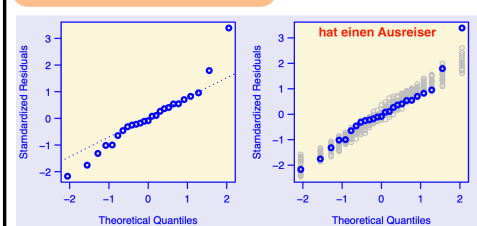
Transformation der Zielvariable

- $\log(y)$
- $\sqrt{y}$
- $\arcsin(p)$

- Liegt der Glättern innerhalb der Fluktuation?
- Weist er eine Struktur auf? z.B. linear zunehmend

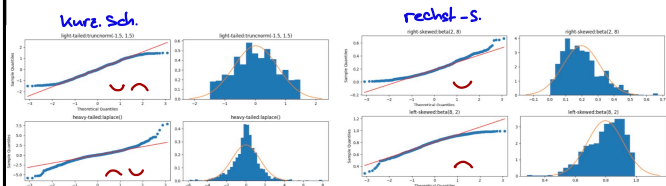
Normal Q-Q-Plot

Prüft auf Normalverteilung



Was oft hilft:

oft Y transformieren oder robuste Methoden



kurze Sch.

rechel-S.

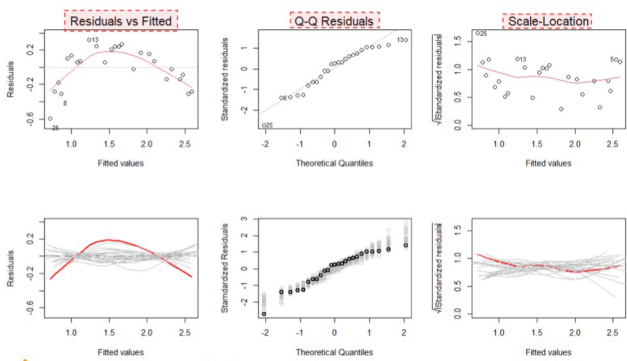
Bei Langschwänzigkeit helfen Transformationen oft NICHT ausreichend, dann sind robustere Methoden verlangt.

Residuen - Analyse

Wird verwendet um zu prüfen, ob das Regressionsmodell überhaupt zu den Daten passt.

```
par(mfrow = c(2,4))
plot(Synth.lm_2)
plot.lmSim(Synth.lm_2)
```

mit Which = 1:3 werden nur die unteren Plots ausgegeben.



- 1 Erwartungswert der Residuen = Null?
- 2 Fehler sind Normalverteilt?
- 3 Streuung konstant? (Varianz)

wird nicht behandelt ← 4

Modell: $Y_i = \alpha + \beta x_i + E_i$ mit E_i unabhängig $\sim N(0, \sigma^2)$

Transformationen Rück-Transformation gemacht?

Hat die Residuenanalyse ergeben, dass das Modell nicht passt, wird mittels Transformationen versucht dies zu verbessern.

Ziel: Linearität, Konstante Varianz.

First Aid Transformationen Häufig: Abhängigkeit der Streuung von $\hat{y}$ (nimmt im Diagramm zu, Normal-Plot zeigt eine rechtsschiefe Verteilung) -> Logarithmieren der Zielgröße hilft Logarithmus-Transformation Daten > 0 ! Für Konzentrationen und Beträge / Mengen summary(data) -> Daten anschauen, Differenz Min/Max < als Faktor 2 -> Log lohnt sich nicht Wurzeltransformation Für Zählvariablen Anzahl, Stückzahl... Arcus-Sinus-Wurzel-Transformation / Logit-Transformation $\hat{y} = \arcsin(\sqrt{y})$ resp. $\hat{y} = \log\left(\frac{y+0.005}{1.01-y}\right)$ Für Anteile / Prozentzahlen Werte zwischen (0,1) Transformation der Zielvariablen ändert Form der Verteilung der Fehler -> Potenzgesetz für die ursprünglichen Größen, somit ist der Fehler proportional Falls $\beta \approx 1$ ist die Zielvariable proportional zu x bis auf einen multiplikativen zufälligen Fehler. Rücktransformation ergibt nicht immer dasselbe Ergebnis wie zuvor	Erwartungswert ist nicht konstant 0 Systematische Abweichungen im Erwartungswert können oft durch: - Transformation der erklärenden Variablen x - Oder durch Hinzufügen eines zusätzlichen Terms x^2 (quadr. Regression) zum Verschwinden gebracht werden. Ausreisser - Fehler sind nicht normalverteilt Auf Richtigkeit der Daten prüfen! -> Transformationen können helfen Ausreisser entfernen: D.synt.lm2 <- lm(y ~ x1 + x2, data=D.synt, subset=c(7,17,27)) Langschwänzigkeit - Fehler sind nicht normalverteilt Extremste Beobachtungen weglassen, bis Langschwänzigkeit verschwindet (1 optimistisch) Kleinste-Quadrate-Methode nicht optimal -> nur robuste Methoden geeignet (geringere Gewichtung von Ausreißern) Unabhängigkeit zufälliger Fehler Bei Beobachtungen mit zeitlicher Reihenfolge, können Autokorrelationen vorhanden sein. -> Residuen in dieser Reihenfolge auftragen (keine Strukturen ersichtlich?) -> oder R_i gegen R_{i-1} in Streudiagramm auftragen - Punkte sollten frei streuen und keine Korrelation zeigen Wenn Autokorrelationen vorliegen: - P-Werte der üblichen Tests häufig grob falsch - Vertrauensintervalle üblicherweise zu kurz
--	--

Polynomiale Erweiterung (x^2)

Das Modell ist nicht linear in x, aber weiterhin linear in den Koeffizienten!

Daher immer noch lineare Regression!

Anstelle x zu log-Transformieren, reicht es oft x^2 ins Modell zu nehmen.

Wann? Z.B. Tukey-Anscombe weist Systematische Krümmung auf.

$$Y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + E_i$$

β_0 ist der Achsenabschnitt.

β_1 beschreibt den linearen Anteil.

β_2 beschreibt die Krümmung.

$\beta_2 > 0$, ist sie nach oben geöffnet.
 $\beta_2 < 0$, ist sie nach unten geöffnet.

Multiple lineare Regression

$$Y_i = \beta_0 + \beta_1 x_i^{(1)} + \beta_2 x_i^{(2)} + \dots + \beta_m x_i^{(m)} + E_i$$

mit E_i unabhängig $\sim N(0, \sigma^2)$

Gleicheitende Varianz

Multiple Regression ist NICHT einfach die Summe mehrerer einfacher Regressionen! Der jeweilige Koeffizient β_j beschreibt den Einfluss von $x^{(j)}$, wenn die anderen Variablen konstant gehalten werden

Einfache Regressionen:

```
> Sy.lm1 <- lm(y ~ x1, data=Sy)
> summary(Sy.lm1)
```

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	4.214	5.499	0.766	0.4781
x1	4.028	1.230	3.273	0.0221 *

Residual standard error: 6.507 on 5 degrees of freedom

steigung ist signifikant

Einfache Regressionen:

```
> Sy.lm2 <- lm(y ~ x2, data=Sy)
> summary(Sy.lm2)
```

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	14.741	9.717	1.517	0.190
x2	1.300	2.045	0.636	0.553

Residual standard error: 11.1 on 5 degrees of freedom

steigung ist NICHT signifikant

Multiple Regressionen:

```
> Sy.lm12 <- lm(y ~ x1 + x2, data=Sy)
> summary(Sy.lm12)
```

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	8.9464	0.8514	10.51	0.000464 ***
x1	7.2783	0.2763	26.34	1.23e-05 ***
x2	-4.1406	0.2695	-15.36	0.000105 ***

Residual standard error: 0.9392 on 4 degrees of freedom

multiple vs. einfache Regression kann zu erheblichen Interpretationsunterschieden führen

alle Werte sind signifikant

Interpretation

x_1 & x_2 einzeln modelliert ergibt jeweils:

- Positiv
- Signifikant

Nun beide Variablen in einem Modell ergeben jeweils beide sind noch einen signifikanten Einfluss auf y

x_2 hat nun jedoch einen **NEGATIVEN** Einfluss auf y

Nur eine multiple Regression zeigt den wahren Einfluss der einzelnen erklärenden Variablen! Aber nur wenn alle wichtigen Erklärenden-Variablen berücksichtigt wurden. (oft kennt man gar nicht alle)

Grunderidee

Die lineare Regression versucht, eine Zielvariable Y durch eine oder mehrere erklärende Variablen (EV) $x_1, x_2, \dots, x_m$ zu beschreiben oder vorherzusagen. Die Parameter β_j beschreiben den geschätzten Einfluss der EV auf Y. Wenn alle β_j linear sind, jedoch x_j nicht zwingend. Das Modell wird so gewählt, dass die vorhergesagten Werte $\hat{y}_j$ möglichst gut zu den beobachteten Werten y_j passen. Die Differenz $R_i = y_i - \hat{y}_i$ heißt Residuum und entspricht dem beobachteten Fehler. Daraus folgt dann die Residuenanalyse, welche beurteilen kann, ob die Modellannahmen vertrauenswürdig sind. Wird ein Muster oder eine Verteilung erkannt, so muss das Modell weiter angepasst werden. Z.B. Transformation, zusätzliche Variablen, robustere Methoden...

Intercept (Beschreibt NICHT mehr den Achsenabschnitt!)

$$\hat{Y} = \beta_0 \quad \hat{Y} = \beta_0 + \beta_1 \cdot \text{Distanz} + \beta_2 \cdot \text{Ladung}$$

Fall: Alle Erklärenden-Variablen sind 0
 In der Praxis eher unrealistisch Distanz=0, Ladung=0

Stetige erklärende Variable (β_1) $H_0: \beta_j = 0$ Wie bisher interpretiert

Wenn EV um 1 Einheit steigt, dann ändert sich Y im Mittel um β_1 . Während die anderen Variablen gleich bleiben

Binäre erklärende Variable $H_0: \beta_j = 0$ Kein Unterschied zwischen den beiden Gruppen

$$x_i = \begin{cases} 0 & \text{Gruppe A} & \text{Für } x_i = 0: Y_i = \beta_0 + E_i \\ 1 & \text{Gruppe B} & \text{Für } x_i = 1: Y_i = (\beta_0 + \beta_1) + E_i \end{cases}$$

β_0 ist das Niveau der Referenzgruppe
 β_1 ist die Verschiebung zur zweiten Gruppe

Faktorvariable $H_0: \beta_{2,2} = 0$ Stelle 2 unterscheidet sich nicht von Referenzstelle 1

Ist die Erweiterung der Binären Version
 Nun mehr als zwei Kategorien

Z.B. Messstellen

Eine Kategorie wird nicht als Koeffizient angezeigt, sondern als Referenz genommen

• StelleSt2 • StelleSt3 • StelleSt4 Dann ist StelleSt1 automatisch die Referenz.	• (Intercept) = Referenzstelle • StelleSt2 = Unterschied von Stelle 2 zur Referenzstelle 1 • StelleSt3 = Unterschied von Stelle 3 zur Referenzstelle 1 • StelleSt4 = Unterschied von Stelle 4 zur Referenzstelle 1
---	---

Vergleich von Regressionsmodellen

Kleines Modell: $lErsch \sim lDist + lLadung$ Spr2.lm3

Grosses Modell: $lErsch \sim lDist + lLadung + Stelle$ Spr2.lm2

Frage: Verbessert ein grösseres Modell die Datenbeschreibung gegenüber einem kleineren Modell?

Das normale Summary des jeweiligen Modells kann diese Frage nicht beantworten!

Besonders wichtige Frage bei Faktorvariablen

Einzelne Stufen können nicht signifikant sein, obwohl die Faktorvariable als ganzes relevant ist.

F-test: (anova/drop1)

`anova(modell_gross, modell_klein)`

Kleines Modell: $Y \sim x_1 + x_2$
Grosses Modell: $Y \sim x_1 + x_2 + x_3 + x_4$

} Geschachtelt da alle Variablen aus dem kleinen Modell im Grossen enthalten sind

anova testet auf:

$H_0: \beta_3 = \beta_4 = 0$
 $H_A: \text{mindestens einer von beiden ist ungleich } 0$

} also auf die erweiterten Variablen

Output:

- Res. Df = Verbleibende Freiheitsgrade (wie viele Parameter das Modell hat)
- RSS = Residual of Sum Squares
↳ Kleiner = besser Modell macht weniger Fehler
- Df = Anzahl dazugekommene/entfernte Freiheitsgrade (Parameter)
- Sum of Sq = Verbesserung der Fehlerquadratsumme $RSS_{\text{klein}} - RSS_{\text{gross}}$
↳ Grösser = besser (Grosses Modell ist besser als kleines Modell)
- $Pr(>F)$ = P-Wert
↳ Verwerfen $p < 0.05$ Das grössere Modell beschreibt die Daten signifikant besser

`drop1(modell_gross, test="F")` F-test

- Jede Variable wird einzeln geprüft
- Man sieht auf einem Bilde welche EV signifikant sind
↳ Für Koeffizienten trotzdem Summary
- Diese Funktion ist besonders relevant wenn Faktorvariablen vorkommen
Problem: Das Summary gibt in der Signifikanz an ob der Koeffizient sich signifikant zur Referenz unterscheidet aber nicht ob die Faktorvariable insgesamt signifikant ist
- Intercept = Referenzwert = StelleSt1 signifikant, StelleSt2 signifikant, Stelle3,4 = nicht signifikant
Stelle2,3,4 werden jedoch nur gegenüber Stelle 1 betrachtet also ob StelleSt4 sich signifikant zu StelleSt1 unterscheidet

Drop1 beantwortet die Frage:

Wird das Modell signifikant schlechter, wenn ich diese Variable weglasse?

Output:

- Df = wie viele Freiheitsgrade verloren gehen
- Sum of sq = wie stark sich die Fehlerquadratsumme erhöht, wenn du diese Variable weglässt
↳ Je grösser, desto wichtiger ist diese Variable für das Modell
- RSS = Fehlerquadratsumme nach dem Weglassen dieser Variable
↳ kleiner = besser
- F-Value = Die F Teststatistik
- $Pr(F)$ = P-Wert
↳ Verwerfen $p < 0.05$ Verbessert das Modell signifikant

Residuen-Analyse plot (lm-modell)

Gleiches Vorgehen wie bei einfacher linearen Regression!

Es wird geprüft ob die Varianz der Fehler konstant ist, Erwartungswert 0 und Normalverteilung.

Wichtig: Wir prüfen ob der Fehler diese Eigenschaften erfüllt mit Hilfe der Residuen, obwohl diese theoretisch nicht exakt konstante Varianz haben. Zudem sind Residuen auch oft korreliert, also nicht unabhängig.

Da wir aber den echten Fehler nicht kennen, müssen wir uns mit den Residuen zufriedensetzen.

Durch das Standardisieren der Residuen (skalieren) wird dieses Problem gemindert. Es wird jegliches versucht zu erkennen ob ein Muster auftritt das mir genügend Evidenz gibt zu sagen, dass dies so nicht sein darf.

Kleine Strukturen innerhalb der statistischen Fluktuationen sind i.O. Da Varianz in den Residuen

Ich möchte prüfen ob ein Tisch schief ist. Mein Messgerät (Residuen) ist aber nicht perfekt daher messe ich nicht exakt und habe eine nicht konstante Varianz.

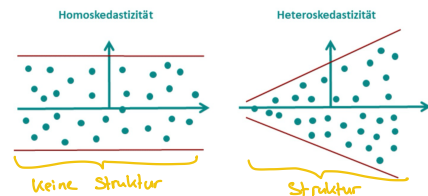
Allgemein frage ich mich also nicht: Ist jede Messung exakt identisch (Echter Fehler)

sondern:

Sehe ich mehr Abweichung als ich erwarten würde

Zusätzliche echte Struktur nennt man:

Heteroskedastizität



PRESS-Residuen (leave-one-out)

Wie gross wäre der Vorhersagefehler für Beobachtung i , wenn diese Beobachtung beim Fitten des Modells gar nicht verwendet worden wäre?

PRESS Residuen sind leave one out Vorhersagefehler. Je grösser ihr Betrag, desto schlechter sagt das Modell diesen Punkt vorher, wenn der Punkt beim Fitten weggelassen wird.

Einzelne EV prüfen

Wenn im gesamten Modell etwas nicht passt so muss ermittelt werden welche erklärende Variable verantwortlich ist, um zu wissen welche EV transformiert werden soll.

Allgemeines Schema:

Für eine bestimmte EV (z.B. x_1), einen Plot erstellen mit:

X-Achse: x_1

Y-Achse: das entsprechende Residuum

↳ Linearität prüfen $E(E_i) = 0$: } Tulkey-Anscombe-Diagramm
Unveränderte Residuen

↳ Normalverteilung prüfen Std.-Residuen : } Normal QQ Plot

↳ Varianz prüfen $\text{var}(E_i) = \sigma^2$: } Scale-location plot
Wurzel-abs-std.-Residuen

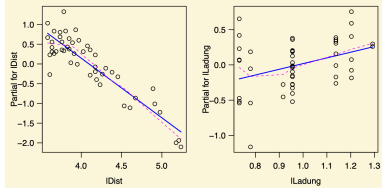
Diese Art eignet sich jedoch nur bedingt und soll nur als grobe Einschätzung dienen. Besser "Partial Residual Plots"

Partial Residual Plots `termplot()`

Hat die Residuen-Analyse ergeben dass das Modell nicht valide ist so möchte man durch z.B Transformation das Modell verbessern.
Doch welche EV muss transformiert werden?

Prüfung einer einzelnen EV ob sie zum Modell passt
Der Einfluss der anderen EV wird möglichst herausgerechnet

`termplot(D.sp.lm2, partial.resid=TRUE, smooth=panel.smooth, ylim="free", col.res="black", col.term="blue", col.smth="magenta")`



vom linearen Modell
berechnete Gerade

Glätter der Daten-
Punkte

Interpretation:

Glätter folgt ungefähr der Geraden $\Rightarrow$ Passende EV

Glätter ist gekrümmt etc. $\Rightarrow$ Transformieren

$lDist$ & $lLadung$ sind also passend durch ihre Transformation

Interaktion Prüfen

Wir gehen grundlegend davon aus dass unsere Modelle **additiv** sind. $Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + E$

Y = Umsatz
 x_1 = Werbung auf Youtube
 x_2 = Werbung auf Facebook
Der Effekt von x_1 hängt nicht von x_2 ab
Der Effekt von x_2 hängt nicht von x_1 ab

Eine zusätzliche Einheit Youtube verändert den Umsatz immer um den selben Betrag, egal wie hoch Facebook gerade ist

Oft kommen aber EV vor welche wechselwirkend sind.
Die eine Variable beeinflusst die andere

x_1 = Lernstunden
 x_2 = Schlafstunden

In der additivität würde man davon ausgehen, dass Lernen der Effekt einer zusätzlichen Lernstunde immer gleich gross ist

Ganz egal ob jemand 3 Stunden oder 8 Stunden schläft

Schnell wird jedem bewusst, dass dies keine korrekte Annahme ist
Viel eher: Wenn jemand genug schläft, bringt lernen viel
Wenn jemand fast nicht schläft, bringt lernen wenig

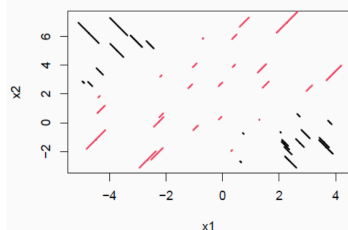
Ein Passendes Modell wäre dann $Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_1 x_2 + E$

Interaktionsterm

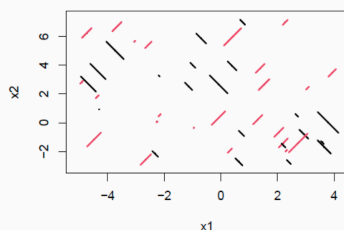
Plot zur Mustererkennung `p.res.2x()`

```
library(sfsmisc)
fit_add <- lm(y ~ x1 + x2, data = dat)
p.res.2x(x ~ x1 + x2, data = fit_add, scol = 2:1)
```

Beispiel für Interaktion



Beispiel für keine Interaktion



Ist ein Muster klares Muster zu erkennen?

- Die Striche sind die jeweiligen Residuen
- Die Steigung (Farbe) gibt dem Residuum das Vorzeichen
 $\hookrightarrow$ Hier Rot = Positiv, Schwarz = Negativ
- Die Länge des Striches ist die Grösse des Residuums

Modell nach Anpassung prüfen

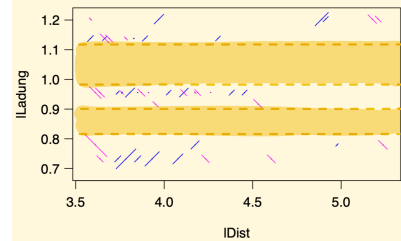
```
fit_add <- lm(y ~ x1 + x2, data = dat)
fit_int <- lm(y ~ x1 * x2, data = dat)
anova(fit_add, fit_int)
```

Modelle erstellen und mit der `anova` Funktion vergleichen, ob das neue Modell auch wirklich eine Verbesserung ist.

Weiteres Beispiel

```
library(sfsmisc)
fit_add <- lm(lErsch ~ lLadung + lDist, data = D.sp)
p.res.2x(
  x ~ lDist + lLadung,
  data = fit_add,
  size = 1.2,
  scol = c("blue", "magenta")
)
```

Keine (sichtbare) Evidenz,
 $\uparrow$ dass eine Wechselwirkung vorliegt!



Solche Lücken sind kein Problem. Sie sind entstanden, da es keine halben Ladungen gab.

z.B. Ladung Typ 1 = 2kg
Ladung Typ 2 = 4kg...

Gewichtete Regression

Variant im Fehler
nicht konstant?!

Hat z.B. die Residuenanalyse ergeben, dass die Streuung nicht konstant ist, dann kann eine gewichtete lineare Regression abhilfe schaffen.

Wenn viele Stichproben erhoben wurden, dann sinkt die Varianz (wird genauer). Diesen Stichproben werden dann mehr gewichtet.

- Mittelwert aus Fischart-1 mit 5 Messungen
- Mittelwert aus Fischart-2 mit 1 Messung

Da die Varianz und das Mittel mit der Anzahl Messungen genauer wird gehen wir dort mehr Gewicht

$\rightarrow$ mit Gewichtung

```
> FFat2.lm <- lm(Fat.mean ~ Species, data=FFat2.Agg, weights=NoS)
```

```
> summary(FFat2.lm)
```

```
...
Coefficients:
(Intercept) 17.1625 1.4917 11.506 2.95e-06 ***
Speciesb    -1.0339 2.1836 -0.474 0.64851
Speciesc    -0.1792 2.2786 -0.079 0.93926
Speciesd    -8.1482 2.1836 -3.732 0.00577 **
```

H_0 : Kein Unterschied zwischen den Fischarten

Residual standard error: 4.219 on 8 degrees of freedom
Multiple R-squared: 0.692, Adjusted R-squared: 0.5765
F-statistic: 5.991 on 3 and 8 DF, p-value: 0.01922

H_0 würde hier verworfen werden

Ohne Berücksichtigung der unterschiedlichen Genauigkeiten:
> FFat2.lm00 <- lm(Fat.mean ~ Species, data=FFat2.Agg)

```
> summary(FFat2.lm00)
```

```
...
Coefficients:
(Intercept) 15.7139 1.7390 9.036 1.8e-05 ***
Speciesb    -0.7556 2.4593 -0.307 0.7665
Speciesc    1.8083 2.4593 0.735 0.4831
Speciesd    -5.6889 2.4593 -2.313 0.0494 *
```

Residual standard error: 3.012 on 8 degrees of freedom
Multiple R-squared: 0.5603, Adjusted R-squared: 0.3955
F-statistic: 3.398 on 3 and 8 DF, p-value: 0.07394

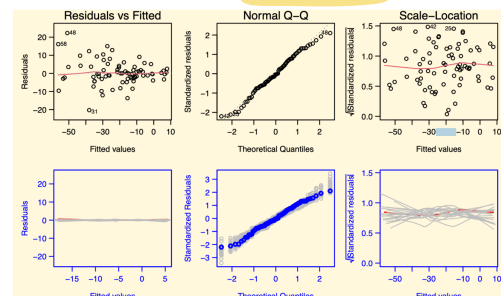
H_0 würde nicht verworfen werden

Die gewichtete Regression ist hier die korrekte Analyse!

Weiteres Beispiel

Die Gewichtungen wirken oft willkürlich
 $\rightarrow$ hat aber in Realität eine Bedeutung
diese zu kennen ist jedoch oft schwierig

```
> SyD2.lm2 <- lm(y ~ x1 + x2, weight=1/SyD2$x1^2, data=SyD2)
```



Die Gewichtung hat perfekt funktioniert

Die Residuen-Analyse ist perfekt!

Ausreisser

Residuals vs. Leverage

Welche einzelnen Datenpunkte, können mein Modell stark beeinflussen?

Nicht alle Ausreisser sind problematisch, wo ziehen wir die Reissleine?

Dazu betrachten wir den Hebelarm und Cook's Distance

Hebelarm: Ein Punkt der auf der x-Achse weit weg von den anderen Punkten liegt

↳ Hebelarm $> 2 \frac{P}{n}$

↳ Hebelarm $< 0,2$ unbedenklich

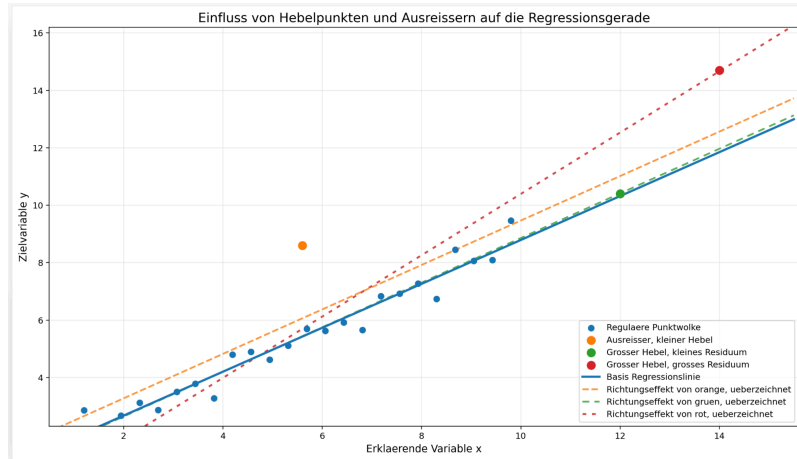
↳ Hebelarm $> 0,5$ sollten vermieden werden

Residuum: Es kann zwar auch Ausreisser auf y-Achsen Ebene geben diese sind oft nicht so gravierend.

Cook's: Das ist die Kombination von Hebel (H_{ii}) und Std. Residuum.

↳ Cook's Distance $> 1 \Rightarrow$ zu einflussreich

↳ Cook's Distance sagt also wie stark dieser eine Punkt das Modell beeinflusst.

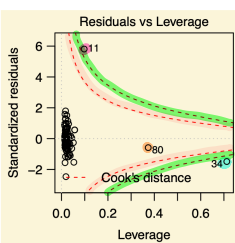


Diese Grafik veranschaulicht klar die 3 möglichen Kombinationen

1. Grosser Hebel kleines Residuum $\Rightarrow$ Kaum Einfluss auf Gerade
2. kleiner Hebel grosser Residuum $\Rightarrow$ Verschiebt die Gerade Vertikal (weniger dramatisch)
3. Grosser Hebel grosses Residuum $\Rightarrow$ Zieht die Gerade regelrecht an (sehr schnell dramatisch)

Einordnung mit Cook's Distance $\text{plot}(\dots, \text{which} = 5)$

Das ist der letzte Standardplot welcher bei der Plot-Funktion ausgegeben wird. Mit diesem Plot, kann man gut entscheiden wo die Reissleine gezogen wird. Nämlich dort wo Cook's Distance > 1 ist



Die Punkte welche ausserhalb dieser Grenzen liegen, sind Problematisch

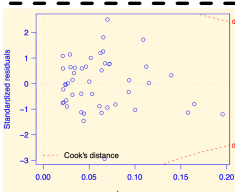
Die Punkte welche ausserhalb dieser Grenzen liegen, sind laut Faustregel der Hebel (Problematisch)

Interpretation der Ausreisser:

11 Hat zu grosses Residuum und leichten Hebel

34 Hat zu grossen Hebel und etwas Residuum

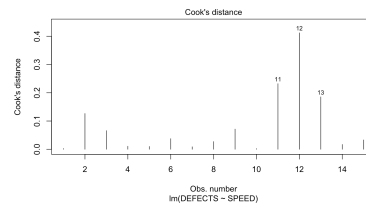
80 Hat zwar grossen Hebel aber kleines Residuum NICHT KRITISCH



Sieht man die Grenzen nur kaum oder gar nicht, ist das kein Fehler!

Im Gegenteil kein Ausreisser "Hura!"

Cook's Distance alleine $\text{plot}(\dots, \text{which} = 4)$



alles kleiner 1
alles super!

Dennoch werden die 3 grössten Werte markiert.

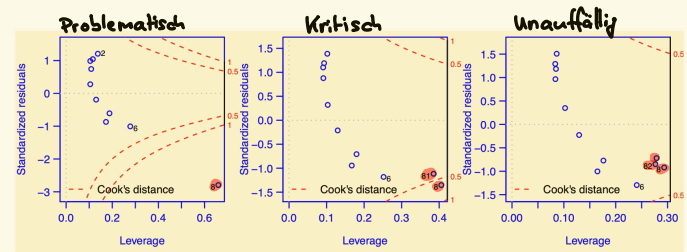
Werte der y-Achse berücksichtigen!

Robuste Anpassungsmethoden

Wurde z.B. mit Cook's Distance ein Ausreisser ... Hebelpunkt identifiziert darf man diesen nicht einfach weglassen. Dies hat mehrere Gründe wie z.B. Subjektivität, Inferenz wird falsch etc.

Aus diesem Grund werden robustere Methoden verwendet.

Zudem gibt es auch den Fall von maskierten Ausreisser. Mit Cook's Distance werden leicht einzelne Punkte sichtbar, sammeln sich jedoch mehrere Ausreisser zusammen an, werden sie oft nur schwer erkannt.



Das Grundproblem dabei ist, dass das kleinste Quadrat Verfahren sehr empfindlich auf Ausreisser ist. Grund: Residuen werden Quadratisch gewichtet. Wie oben erwähnt, dürfen Ausreisser jedoch nicht einfach blind gelöscht werden!

Robustheit messen: Einflussfunktion und Bruchpunkt

Die Einflussfunktion kann Aussagen machen wie stark die Wirkung einer einzelnen Beobachtung ist.

Einflussfunktion = Wirkung einer einzelnen Beobachtung auf die Schätzung

Der Bruchpunkt sagt dabei: Wie viele schlechte Datenpunkte es braucht, bis die ganze Schätzung zusammenbricht.

Ein sehr guter robuster Schätzer hat: $\epsilon^* \approx \frac{1}{2}$

Beispiel: Daten: 1, 2, 3, 4, 5 dann $\bar{x} = 3$

nur 1 Datenpunkt anders: 1, 2, 3, 4, 1.000.000 dann $\bar{x} = 200.002$

Der Mittelwert ($\bar{x}$) hat also einen sehr kleinen Bruchpunkt $\epsilon^* \approx 0$

Daten: 1, 2, 3, 4, 5 dann $\bar{x} = 3$

nur 1 Datenpunkt anders: 1, 2, 3, 4, 1.000.000 dann $\bar{x} = 3$

Der Median hat einen sehr hohen Bruchpunkt. Erst wenn ungefähr die Hälfte $\epsilon^* \approx \frac{1}{2}$ der Werte schlecht ist, kann der Median wirklich kaputtgehen.

Zusammenhang

Einflussfunktion misst den Schaden durch einen einzelnen schlechten Punkt. Der Bruchpunkt misst, wie viel schlechte Punkte nötig sind, bis die ganze Methode zusammenbricht.

Grundidee robuster Regression

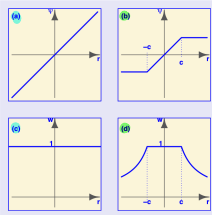
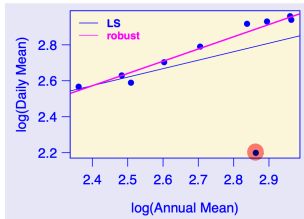
Im Gegensatz zur klassischen Methode (kleinste Quadrate) **LS** verwendet die Robuste-Regression die Idee, dass grosse Residuen weniger stark gewichtet werden. Auffällige Beobachtungen erhalten also weniger Gewicht

$$\sum_{i=1}^n w_i r_i^2$$

Regression M Schätzer

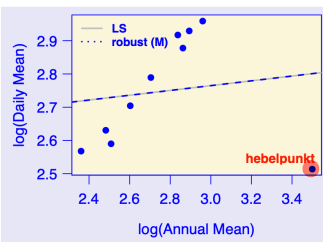
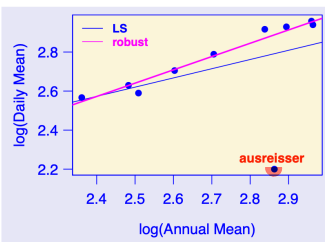
```
library(MASS)
fit.M <- rim(y ~ x1 + x2, data = daten, method = "M")
```

Das ist die erste Alternative zur klassischen Methode. Dabei wird die jeweilige Gewichtund der Residuen mittels einer definierten Funktion festgelegt.



- (a): Klassische Regression
(c): Klassische Gewichtung
(b): M-Schätzung
(d): M-Gewichtung

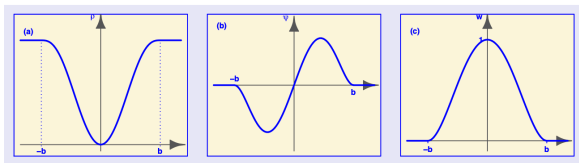
Bei Ausreissern funktioniert die M-Schätzer Methode sehr gut jedoch bei Hebelpunkten nicht (siehe Grafik)



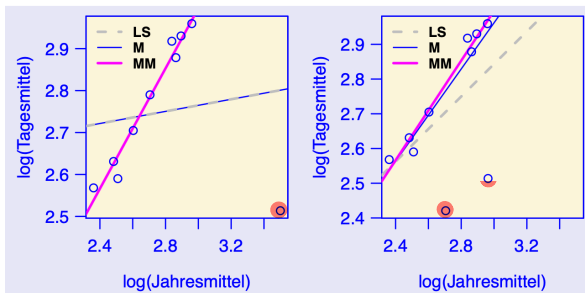
Regression MM Schätzer

```
library(robustbase)
fit.MM <- lmrob(y ~ x1 + x2, data = daten, setting = "KS2014")
```

Wie wir gesehen haben kann die M-Schätzer Methode versagen. Daher gibt es eine weitere Methode (MM) welche die Gewichtung robuster aber komplexer festlegt. Diese Methode hat einen sehr hohen Bruchpunkt von $\epsilon = \frac{1}{2}$



- (a): Wie stark wird ein Residuum bestraft.
(b): Einfluss der Residuen
(c): Gewichtung der Residuen



Robuste Methoden sind also wesentlich stabiler. Trotzdem dürfen diese Methoden nicht als Standard bei jeder Regression verwendet werden, weil bei sauberen Daten LS effizienter und genauer ist.

Modellvergleich

```
anova(fit.gross, kleineres_modell, test = "Deviance")
```

Auch bei der Robusten-Regression können die Geschachtelten Modelle miteinander verglichen werden. Es ist alles analog wie bisher: Hypothesen, Schachtelung... Jedoch ist die Test-Art nun nicht mehr ein F-Test, sondern "Deviance".

Output:

```
Robust Deviance Table

Model 1: y ~ x1 + x2 + x3 + x4
Model 2: y ~ x1 + x2

pseudoDf Test.Stat Df Pr(> chisq)
1      95         12.40 2 0.0020 **
2      97         12.40 2 0.0020 **
```

$H_0: \beta_3 = 0$ und $\beta_4 = 0$

↳ Das kleine Modell reicht aus

H_A : Mindestens einer der zusätzlichen Koeffizienten ist nicht 0

↳ Das grössere Modell ist nötig

Pr(>chisq) Fall 1: $p < 0,05 \Rightarrow H_0$ verwerfen!

↳ Das kleinere Modell reicht nicht aus
↳ Grösser Modell besser

Fall 2: $p \geq 0,05$

↳ Kleiner Modell reicht aus
↳ Kleiner Modell besser

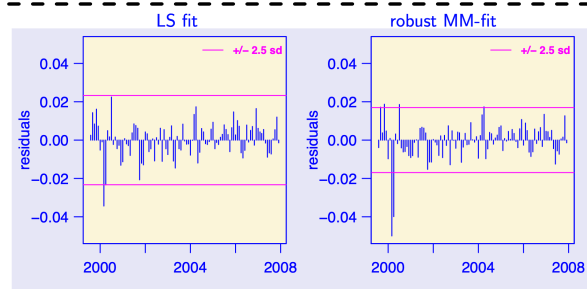
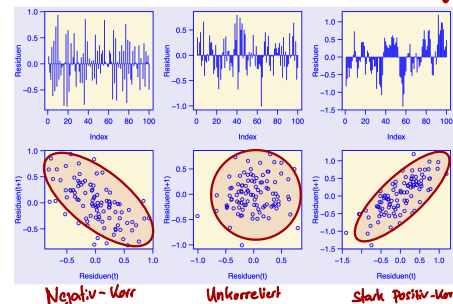
Methode	Was passiert mit grossen Residuen?
Klassische Regression	sie zählen voll und können stark ziehen
Robuste M Schätzung	ihr Einfluss wird begrenzt
Redescending Methode	sehr extreme Punkte bekommen fast Gewicht 0

Autokorrelierte Residuen (Residuen gegen die Zeit)

Residuen müssen über die Zeit um 0 streuen und dürfen keine zeitliche Struktur zeigen. z.B. keine Trends, Wellen, Cluster oder hauptsächlich positive / negative Residuen

Das heisst insgesamt wird geprüft ob der Fehler unabhängig ist.

Hier werden die Residuen gegen die Zeit geplottet. Sieht man ein Muster anstelle zufälliger Streuung?



Hier ist klar ersichtlich dass bei der klassischen kleinsten Quadrate Methode Effekte weniger sichtbar und somit übersehen werden können. Mit robusteren Methoden können diese Effekte sichtbar werden.

Was dagegen tun?

Fehlende EV welche Trend, Saison etc. erklärt hinzufügen.

oder: (Wurde nicht behandelt!)

Modell mit korrelierten Fehlern verwenden mit gls()

Variablenselektion

Bei der multiplen linearen Regression stellt sich die Grundlegende Frage: Welche EV sollen ins Modell aufgenommen werden? Dieses Problem entsteht vor allem oft wenn viele EV erhoben wurden, aber nicht klar ist, welche davon wirklich relevant sind.

Warum nicht einfach alle Variablen verwenden?

Einfachheit	Wenn mehrere Modelle ähnlich gut sind, sollte das einfachere Modell bevorzugt werden.
Reduktion der Schätzvariabilität	Unnötige Variablen machen die Schätzungen oft ungenauer.
Interpretierbarkeit	Weniger Variablen machen das Modell verständlicher.
Vorhersage	Weniger Variablen bedeuten weniger Aufwand bei neuen Daten.

P-Wert als Hilfsmittel Falscher Ansatz!

P-Werte werden in der Variablenselektion nicht als endgültiger Beweis verwendet, sondern nur als Hinweis auf die relative Wichtigkeit eine Variable.

- Grosser P-Wert = relativ unwichtig
- Kleiner P-Wert = relativ wichtig

Problem:

Es werden viele Tests durchgeführt. Dadurch steigt die Wahrscheinlichkeit, dass eine Variable nur zufällig signifikant erscheint.

Deshalb ist die Grenze $p < 0.05$ problematisch und nicht automatisch eine sinnvolle Stoppregel. Wird aber in der Praxis fälschlicherweise Weise oft gemacht!

Rückwärtsselektion drop1

Es wird mit dem vollen Modell (Alle EV) gestartet:

$$Y_i = \beta_0 + \beta_1 x_i^{(1)} + \dots + \beta_m x_i^{(m)} + E_i$$

Danach entfernt man Schritt für Schritt die jeweils unwichtigste Variable. Jene mit dem grössten P-Wert? Falsch!

Stoppregel NICHT P-Wert (ist sonst willkürlich)!!!

Vorwärtsselektion add1

Es wird mit dem kleinsten Modell (Achsenabschnitt) gestartet:

$$Y_i = \beta_0 + E_i$$

Danach wird Schritt für Schritt die jeweils beste Variable hinzugefügt. Jene mit kleinstem P-Wert? Falsch!

Stoppregel NICHT P-Wert (ist sonst willkürlich)!!!

Vor- und Rückwärtsselektion mit Faktorvariablen

Es muss anstelle einer t-Tests F-Test gemacht werden!

drop1(model, test = "F") add1(model, scope = ..., test = "F")

Modellwahlkriterium (Stoppkriterium)

Da wir wissen, dass der P-Wert nicht geeignet als Stoppkriterium ist benötigen wir andere Kriterien für unsere Entscheidung.

Dabei soll ein Modell gewählt werden, das die Daten gut beschreibt, aber nicht unnötig kompliziert ist.

Ein gutes Stoppkriterium berücksichtigt deshalb

- Modellgenauigkeit (Wie gut passt das Modell zu den Daten)
- Modellkomplexität (Wie viele Erklärende-Variablen enthält das Modell)

R^2_{adj}

grosser Wert = besseres Modell

Das normale Bestimmtheitsmass R^2 ist ungeeignet weil es nur die Modellgenauigkeit berücksichtigt.

Mehr EV können nur dazu führen das der Wert sich verbessert oder gleich bleibt

Adjusted- R^2 jedoch berücksichtigt die Anzahl an Parameter (EV) und bestraft somit unnötige Komplexität.

Mallow's C_p Kleiner Wert = besseres Modell

Modellwahlkriterium mit besonderem Blick auf den Vorhersagefehler. Ein gutes Modell hat ungefähr: $C_p \approx p$

Wobei P die Anzahl EV ist. Eignet sich eher für Modellvergleiche Wird für eine Graphische Aufbereitung verwendet. (Siehe später)

AIC

Kleiner Wert = besseres Modell

Modellwahlkriterium mit bestrafung für Modellkomplexität

Eignet sich eher als Modell für Prognosen

Tendenziell mehr EV als BIC

BIC

Kleiner Wert = besseres Modell

Modellwahlkriterium mit starker bestrafung für Modellkomplexität

Eignet sich eher als Modell für Interpretationen

Tendenziell weniger EV als AIC

Vorwärtsselektion

Step(direction = "forward")

```
model_null <- lm(Y ~ 1, data = daten)

model_step <- step(
  model_null,
  scope = list(
    lower = ~ 1,
    upper = ~ x1 + x2 + x3 + x4
  ),
  direction = "forward"
)

summary(model_step)
```

Mit minimalem Modell starten und EV hinzufügen.

Rückwärtsselektion

Step(direction = "backward")

```
model_full <- lm(Y ~ x1 + x2 + x3 + x4, data = daten)

model_step <- step(model_full)

summary(model_step)
```

Mit Vollen-Modell starten und EV entfernen

Vollsuche

Modellsuche regsubsets

Grafik subplots

Nicht anwenden bei Faktor-Variablen!

Bei den oberen Methoden ist nicht garantiert, dass das Globale Optimum gefunden wird!

Bei einer Vollsuche jedoch schon. Jedoch ist diese schnell sehr rechenintensiv, da wirklich alle Modell Kombinationen getestet werden.

```
library(leaps)
library(car)

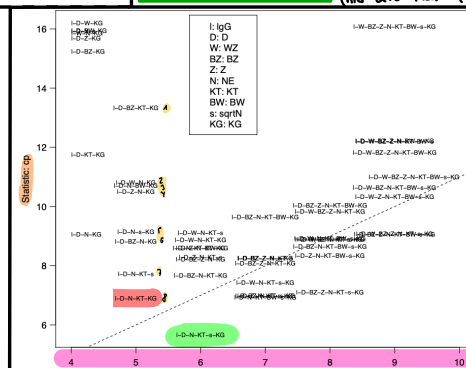
model_cp <- regsubsets(
  Y ~ x1 + x2 + x3 + x4,
  data = daten,
  nbest = 6,
  nvmax = 4
)
```

nbest

Pro Modellgrösze (n-EV) gespeicherte Modelle.

nvmax

Grösst mögliche Modellgrösze (hier also max 4 EV)



Subsets()

nbest = 6

Anzahl EV

Tiefster cp

Weniger Komplexität kommt auch in Frage

Step Output

Start: AIC=130.0	
Y ~ x1	
	Df Sum of Sq RSS AIC
+ x2	1 40.0 168.0 122.0
+ x3	1 20.0 180.0 126.0
<none>	200.0 130.0
+ x4	1 5.0 195.0 131.0
- x1	1 30.0 230.0 134.0
(Step: AIC=122.0)	
Y ~ x1 + x2	
	Df Sum of Sq RSS AIC
+ x3	1 25.0 135.0 119.0
<none>	160.0 122.0
- x1	1 35.0 195.0 125.0
+ x4	1 3.0 157.0 123.5
- x2	1 50.0 210.0 128.0
(Step: AIC=119.0)	
Y ~ x1 + x2 + x3	
	Df Sum of Sq RSS AIC
- x1	1 4.0 139.0 117.5
<none>	135.0 119.0
+ x4	1 2.0 133.0 120.2
- x3	1 25.0 160.0 122.0
- x2	1 45.0 180.0 126.0
(Step: AIC=117.5)	
Y ~ x2 + x3	
	Df Sum of Sq RSS AIC
+ x1	1 4.0 135.0 119.0
+ x4	1 1.5 137.5 119.2
- x3	1 30.0 169.0 123.0
- x2	1 60.0 199.0 128.0
Call:	
lm(formula = Y ~ x2 + x3, data = daten)	

Startmodell

Kleinsten AIC Wert

↳ + = EV aufnehmen

- = EV entfernen

<none> = so lassen

Verbessertes Modell (AIC wurde kleiner)

Finales Modell

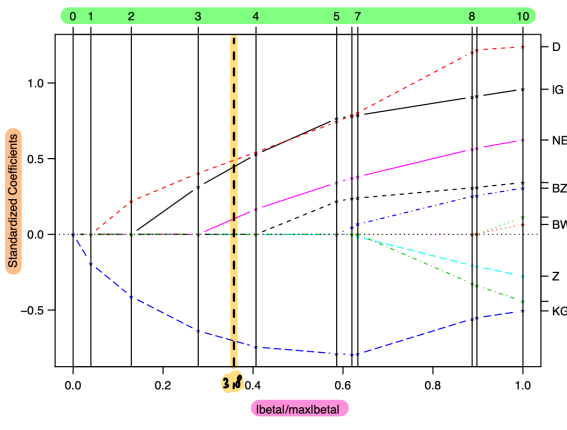
Overfitting

Wenn ein Modell sich zu stark an die vorhandenen Daten anpasst, dann spricht man von Overfitting. Der Fehler E_i ist dann nicht mehr korrekt. Somit sind Prognosen dann nicht mehr zuverlässig.

LASSO

Ist eine weitere Möglichkeit für Variablenselektion. Zudem kann die LASSO Methode mit dem Problem des Overfittings besser umgehen

```
> library(lars)
> h.x <- model.matrix(~ -1 + IG + D + WZ + BZ + Z + NE + KT + BW + wN + KG, data=AKW)
> AKW_lasso <- lars(x=h.x, y=AKW$IK)
> plot(AKW_lasso)
```



Y-Achse

Das sind die standardisierten EV. Je weiter der Pfad von der horizontalen Nulllinie weg geht, desto stärker ist der Einfluss dieser EV!

X-Achse

Zeigt wie stark LASSO bestraft. links = stärkere Bestrafung, rechts = schwächere Bestrafung

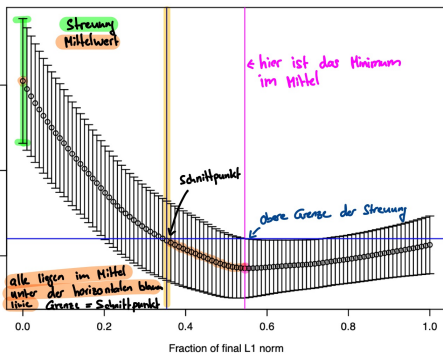
Obere Skala

Zeigt wie viele EV zum entsprechenden Zeitpunkt im Modell aktiv sind

Steigung

Zeigt vor allem ob das Vorzeichen positiv oder negativ ist und die Stärke des Einflusses

Aus dieser Grafik kann einfach abgelesen werden, welche EV was für einen Einfluss haben und wie stark diese sind. Es ist leicht ersichtlich, welche die wichtigen EV sind.



Diese Grafik wird verwendet um ein Optimum zu finden.

X-Achse ist wieder die Stärke der Bestrafung.

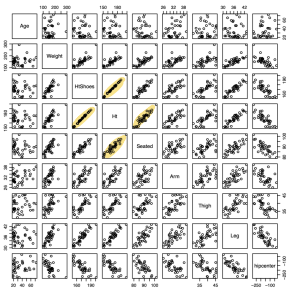
Dann wird wieder die obere Grenze verwendet und die Erkenntnis (gelbe vertikale Linie) in der oberen Grafik angewendet.

Kollinearität

Führt zu Problemen bei der Interpretation

Eine EV lässt sich annähernd durch eine andere EV darstellen.

Der Globale F-Test kann signifikant sein, einzelne t-Test aber nicht. (Das Modell erklärt also insgesamt etwas, aber erkennt nicht welche einzelnen EV verantwortlich sind.)



VIF

```
library(car)
fit <- lm(Y ~ x1 + x2 + x3 + x4, data = daten)
vif(fit)
```

Damit kann gut die Kollinearität erkannt werden

Faustregeln

VIF ≈ 1 : unproblematisch
VIF > 5 : mögliche Kollinearitätsprobleme
VIF > 10 : starke Kollinearitätsprobleme

Kann gut in Paris() erkannt werden

Was tun?

- Wenn immer möglich, soll man Beobachtungen so durchführen, dass das Problem gar nicht erst auftritt.
- Variablen linear transformieren

$$x_{neu} = x_1 - x_2, \quad x_{neu} = x_1 + x_2, \quad x_{neu} = \frac{x_1}{x_2}$$

z.B. Größe mit Schulen - Größe ohne Schulen
- Variablen mit dem höchsten VIF aus dem Modell entfernen (Amputation)

Additive Modelle

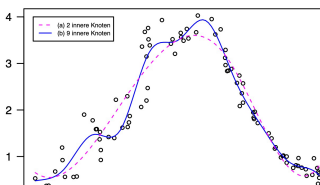
$$Y_i = f_1(x_{1i}) + f_2(x_{2i}) + \dots + f_m(x_{mi}) + E_i$$

Wenn keine passende Transformation, oder aus Fachwissen bekannte Funktion (Physikalische Gesetze) Vorliegen, kann der Zusammenhang datenbasiert als unbekannte glatte Funktion modelliert werden. $Y_i = h(x_i) + E_i$

Polynome sind die naheliegendste Möglichkeit eine Funktion durch die Daten zu legen. Dabei immer der Grad des Polynoms oft hoch sein um die Daten gut zu beschreiben. Dies führt aber oft zum Overfitting und hohen Schwingungen an den beiden Enden, welche dann nicht passen. Daher gibt es zwei andere Ansätze: Splines, lokale...

Splines

Sind stückweise Polynome. Die Übergangspunkte nennt man Knoten. Meistens sind kubische Splines (3. Ordnung) ausreichend.



```
library(splines)
fit_bs <- lm(y ~ bs(x, df = 7, degree = 3), data = daten)
```

Y = Zielvariable

X = Erklärende Variable

bs = Erzeugung der Splines Basisfunktionen

df = indirekte Angabe der Knoten

↳ Wo die Knoten sind wird automatisch über die Quantile von X gesteuert.

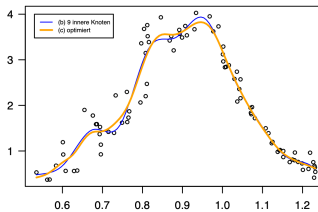
degree = Polynom-Grad

daraus folgt:

```
fit_bs <- lm(y ~ bs(x1, df = 7) + bs(x2, df = 5), data = daten)
```

Smoothing Splines

Es wird nicht primär die Anzahl der Knoten gewählt. Stattdessen wird eine zu wellige Kurve bestraft.



```
fit_smooth <- smooth.spline(x = daten$x, y = daten$y, cv = TRUE)
```

Dieser Ansatz ist nur eine EV anwendbar!

Bei mehreren EV verende:

library(mgcv)

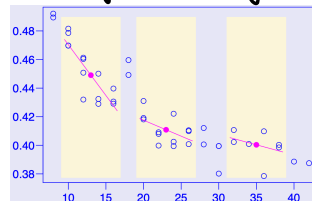
```
fit_gam <- gam(
  y ~ s(x1) + s(x2) + x3,
  data = daten
)
```

Lokale Regression

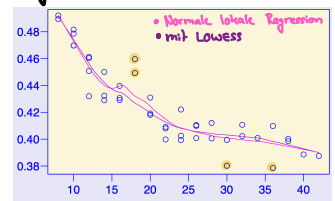
Anstelle eine einer globalen Funktion, wird an viele Stellen lokal eine kleine Regression angepasst. Dabei wird eine Fensterbreite festgelegt, welches die Glätte steuert. Faustregel: $d = \frac{2}{3}n$ Span = 0,66

LOWESS Ansatz LOcally WEighted Scatter Smoother

Ist eine gute Umsetzung der lokalen Regression



Im Zentrum (Punkte) erhalten für die Kurve mehr Gewichtung, als links und rechts vom Zentrum



LOWESS ist sichtlich robust gegenüber Ausreißern (gelbe Punkte)

LOESS ist die allgemeinere Version. Sie ist flexibler einsetzbar.

```
fit_loess <- loess(y ~ x, data = daten, span = 0.3, degree = 2, family = "symmetric")
```

Funktioniert auch für mehrere EV: $y \sim x_1 + x_2 + \dots + x_n$

Wie flexibel? Klein = sehr flexibel, groß = sehr glatt

robuste Anpassung

GAM

gam()

Additives Modell

$$Y_i = f_1(x_{1i}) + f_2(x_{2i}) + \dots + f_m(x_{mi}) + E_i$$

Mehrdimensionale Funktion

$$Y_i = h(x_{1i}, x_{2i}, \dots, x_{mi}) + E_i$$

Man könnte theoretisch bei einem Modell mit mehreren EV eine große Mehrdimensionale Funktion schätzen. Praktisch ist dies jedoch schwierig und daher wird auf das additive Modell zurückgegriffen. Das Ziel ist jedoch nicht das additive Modell so zu verwenden sondern die geeigneten Transformationen zu ermitteln. z.B. $\log, \sqrt{\cdot}$...

```
fit_gam <- gam(
  y ~ log(x1) + log(x2),
  data = daten
)
```

```
fit_gam <- gam(
  y ~ s(x1) + s(x2),
  data = daten
)
```

lo(x, span = 0.5, degree = 1)

s(x, df = 4)

lo(x, span = 0.5, degree = 1)

s(x, df = 4)

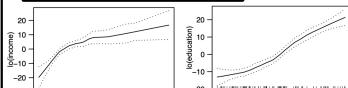
Summary output lesen

Anova for Nonparametric Effects				
	Npar	Df	Npar F	Pr(>F)
(Intercept)				
lo(income)	4.0	5.4878	0.0005086	***
lo(education)	2.4	3.3730	0.0309103	**

Nur dieser Teil ist interessant!

Wenn Signifikanz vorliegt, bedeutet dass das die EV nicht linear sind und daher transformiert werden müssen. Um herauszufinden welche nötig ist können die Partielle Residuen-Plots angeschaut werden.

Partielle Residuen-Plots



Diese Grafiken sollen dabei helfen zu erkennen welche Transformation verwendet werden soll

Faustregel: Kann ungefähr eine Gerade hineingelegt werden so ist eine Transformation nicht nötig. Wenn keine Gerade geht, werden tendenziell dass: Die Nichtlinearität ist statistisch relevant.

Verwendete Transformationen

```
> P$Income <- log(P$Income)
> P$Edu <- ifelse(P$Education < 9, P$Education)
## Ich nenne diese Transformation "Hockey-Stick Trsf."
```

```
P.lm <- lm(prestige ~ income + edu, data=P)
```

Schlussendlich landet man wieder bei einem linearen Modell

Modellbildung

Eine Modellentwicklung lässt sich in den meisten Fällen nicht auf eine automatisierte Variablenselektion reduzieren. Sie genügen oft nicht, weil:

- Auswahl der Variablen ist vom Zufall abhängig
☞ neben dem „besten“ Modell die „fast gleich guten“ auch in Betracht ziehen.
- Das „beste“ Modell muss nicht unbedingt alle Modellvoraussetzungen erfüllen.
☞ Residuenanalyse.
- Können nicht alle potentiell nützlichen Transformationen von Beginn weg berücksichtigen wegen übergrosser Anzahl an Möglichkeiten.
- Die Auswirkung von Multikollinearität auf die automatische Variablenselektion ist schwierig zu durchschauen.
- Manchmal liefern die Verfahren Modelle, die mit dem gesicherten Fachwissen nicht übereinstimmen.
- Die „besten“ Modelle können für einen Zweck sehr tauglich sein, jedoch für einen anderen unbrauchbar.

Variablenselektion ist folglich nur eine Phase in der Entwicklung eines datengestützten Regressionsmodells.

Drei wichtige Elemente bei der statistischen Regressions-Modellierung:

Residuen-Analyse: In der Residuen-Analyse sollen offensichtliche Abweichungen von den Voraussetzungen identifiziert und behoben werden.

Auch versuchen wir, einflussreiche Beobachtungen zu identifizieren.

Transformationen: Je nach Situation ist es sinnvoll,

- die Ziel- und/oder die erklärenden Variablen zu transformieren,
- aus mehreren erklärenden Variablen neue zu bilden oder
- mit Faktorvariablen die Modelle zu erweitern.

Variablenselektion: Welche erklärende Variablen sind sinnvoll und notwendig, um die Zielvariable erklären oder vorhersagen zu können

- In welcher Reihenfolge sollte man diese Modellierungsschritte vornehmen?
- Sollten einige oder alle Schritte wiederholt angewendet werden?
- Wann ist man fertig?

Schlüssige Antworten dazu gibt es nicht.

Grobe Skizze zum Vorgehen

- ❶ Problem verstehen – Gibt es schon Modellansätze? Effektschätzen oder Vorhersage?
- ❷ Daten beschaffen, kennenlernen und aufbereiten
 - Umgang mit fehlenden Werten
 - Bedeutung der Zahl 0 in den verschiedenen Variablen vereinheitlichen
 - Daten gemäss first-aid Transformationen behandeln, ausser es gibt triftige Gründe dagegen (z.B. bestehendes Modell); Interaktionen sinnvoll?
- ❸ Erste Anpassung (mit allen Variablen); vorzugsweise mit robusten Methoden
- ❹ Residuen-Analyse;
 - Tragen die Daten dazu bei, das Problem zu lösen?ev. zurück nach (ii) oder (i)
- ❺ Variablenselektion, allenfalls Kollinearitäten behandeln
- ❻ Modelleignung klären
 - Residuen-Analyse mit selektionierten Modellen
 - Modelle und geschätzte Parameterwerte mit Fachwissen abgleichen
 - 'out-of-sample' Validierung in Betracht ziehen (d.h. Validierung mit noch nicht verwendeten Daten)

Aus den Daten alleine kann man keine Ursache Wirkungs Aussage treffen

Bsp. Storch & Baby Ursache war Industrialisierung

1. Problem verstehen
Geht es um Erklärung, Effektschätzung oder Prognose?
2. Daten kennenlernen und aufbereiten
Fehlende Werte, Bedeutung von 0, erste Transformationen prüfen.
3. Erstes Modell mit allen sinnvollen Variablen anpassen
Wenn möglich auch robuste Methoden betrachten.
4. Residuenanalyse durchführen
Prüfen, ob das Modell grundsätzlich geeignet ist.
5. Variablenselektion durchführen
Dabei Kollinearität prüfen und behandeln.
6. Modell erneut prüfen
Residuenanalyse wiederholen, Fachwissen einbeziehen und falls möglich mit neuen Daten v

Summary vs. anova

Beispiel Sprengungen: R-Summary-Output - Nochmals

```
> Spr2.lm3 <- lm(lErsch ~ lDist + lLadung + Stelle, data=SprengS2)
> summary(Spr2.lm3)
...
Coefficients:
            Estimate Std. Error t value Pr(> |t|)
(Intercept)  5.78051    0.64967   8.898 3.25e-11 ***
lDist       -1.33779    0.14073  -9.506 4.97e-12 ***
lLadung       0.59179    0.29666   2.332  0.0246 *
StelleSt2     0.37832    0.17257   2.192  0.0340 *
StelleSt3     0.04996    0.14657   0.341  0.7349
StelleSt4     0.25511    0.17216   1.482  0.1459
---
Residual standard error: 0.3379 on 42 degrees of freedom
Multiple R-squared:  0.8322, Adjusted R-squared:  0.8122
F-statistic: 41.66 on 5 and 42 DF, p-value: 3.194e-15
```

Intercept ist Stelle 1
also die Referenz Stelle
für Stelle 2, 3, 4

daher muss Referenzstelle
mit anderer Stelle addiert werden

Vs.

```
> anova(Spr2.lm3, Spr2.lm2)
Analysis of Variance Table

Model 1: lErsch ~ lDist + lLadung + Stelle
Model 2: lErsch ~ lDist + lLadung

    Res.Df  RSS Df Sum of Sq  F Pr(>F)
1        42 4.7964          2.2827 0.09297
2        45 5.5785      -3    -0.78205 2.2827 0.09297
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Stelle St 1 = 5,78
Stelle St 2 = 5,78 + 0,378 ⇒ 6,158
im out ist also Stelle St 2 die Differenz
zur Stelle St 1. also parallele Verschiebung

Obwohl im oberen output Stelle St 2
signifikant ist hat die Faktorvariable
"Stelle" keine signifikante Verbesserung
des Modells ergeben! also $p_L = 0$

Erweiterter Entscheidungsbaum für Modellanpassungen

Tukey-Anscombe-Plot

Ja Systematische Struktur im Plot? Nein

Formproblem

- Residuen gegen erklärende Variablen
- Transformation: x^2 , $\log(x)$ oder $1/x$
- Polynomterm: x^2 , $\log(x)$ oder $1/x$
- Interaktion: Faktor x Faktor / $X_1 X_2$

Genügt Anpassung?

- Residuen gegen erklärende Variablen
- Transformation: x^2 , $\log(x)$ oder $1/x$
- Polynomterm: x^2 , $\log(x)$ oder $1/x$
- Interaktion: Faktor x Faktor / $X_1 X_2$

Minutene Anpassung

Scale-Location-Plot

Ja Ändert sich die Streuung systematisch? Nein

Varianzproblem

→ $\sqrt{Y}$ / $\log(Y)$

Gewichtete Regression

→ $\log(Y)$

Erneute Anpassung

- Residuen gegen erklärende Variablen
- Teil-Residuendiagramme
- Interaktion: Faktor x Faktor / $X_1 X_2$

Modell plausibel

QQ-Plot

Schiefe oder auffällige Tails?

Nicht-Normalverteilung

- evtl. → $\log(Y)$
- Einflussanalyse / robustes Verfahren

Einzelne auffällige Punkte?

- Einflussanalyse
- Cook, Hebel
- Studentisierte Residuen