Mehrfache und polynomiale Regression

Kriteria für die Durchführung einer Regression

Jonathan Harrington

Bitte datasets.zip (unter 5.5, Tabellarische Daten) neu herunterladen und in pfad auspacken

Einfache Regression

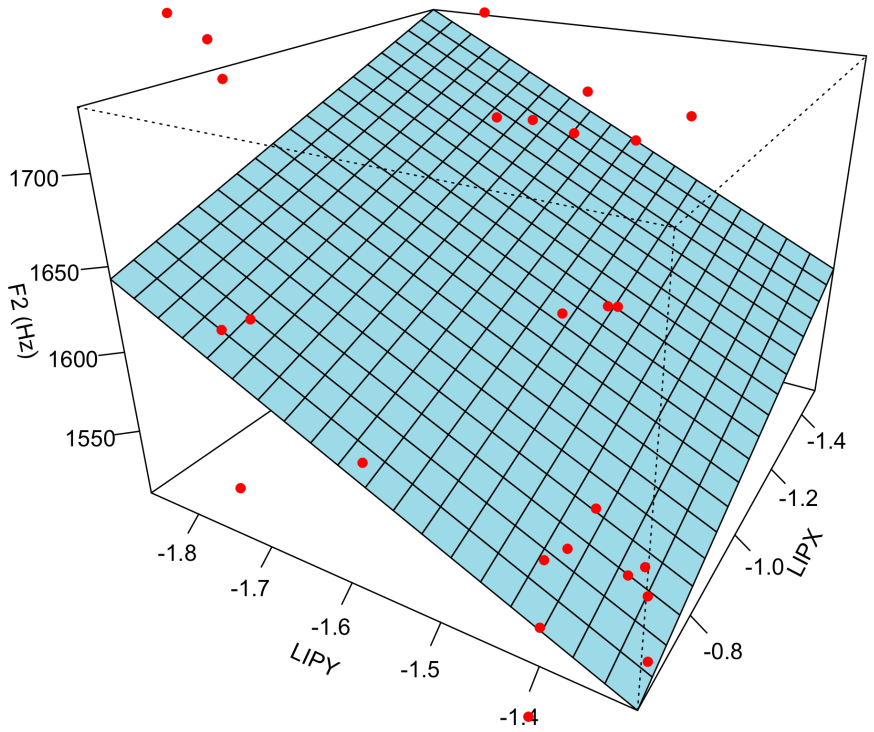
$$\hat{y} = bx + k$$

Mehrfache Regression

$$\hat{y} = b_1 x_1 + b_2 x_2 + k$$

In diesem Fall: 2 Regressoren (x_1, x_2) ,
2 Neigungen (b_1, b_2),
ein Intercept, k

und eine Ebene im 3D-Raum



Mehrfache Regression

Es können auch mehrere Regressoren sein…

$$\hat{y} = b_1 x_1 + b_2 x_2 + b_3 x_3 + \ldots b_n x_n + k$$

n verschiedene Neigungen, ein Intercept

Eine Hyper-Ebene in einem *n*-dimensionalen Raum

Einige Daten

```
ydata = read.table(file.path(pfad, "ydata.txt"))
```

```
names(ydata)
```

Zungendorsum Untere Lippe

```
[1] "F2"    "DORSY" "DORSX" "LIPY"  "LIPX"
```

[y] Vokale, alle Werte zum zeitlichen Mittelpunkt

DORSX, DORSY (horizontale und vertikale Position des Zungendorsums)

LIPX, LIPY (horizontale Verlagerung und vertikale Position der Unterlippe)

Ein mehrfaches Regressionsmodell

$$\hat{F2} = b_1 DORSX + b_2 DORSY + b_3 LIPX + b_4 LIPY + k$$

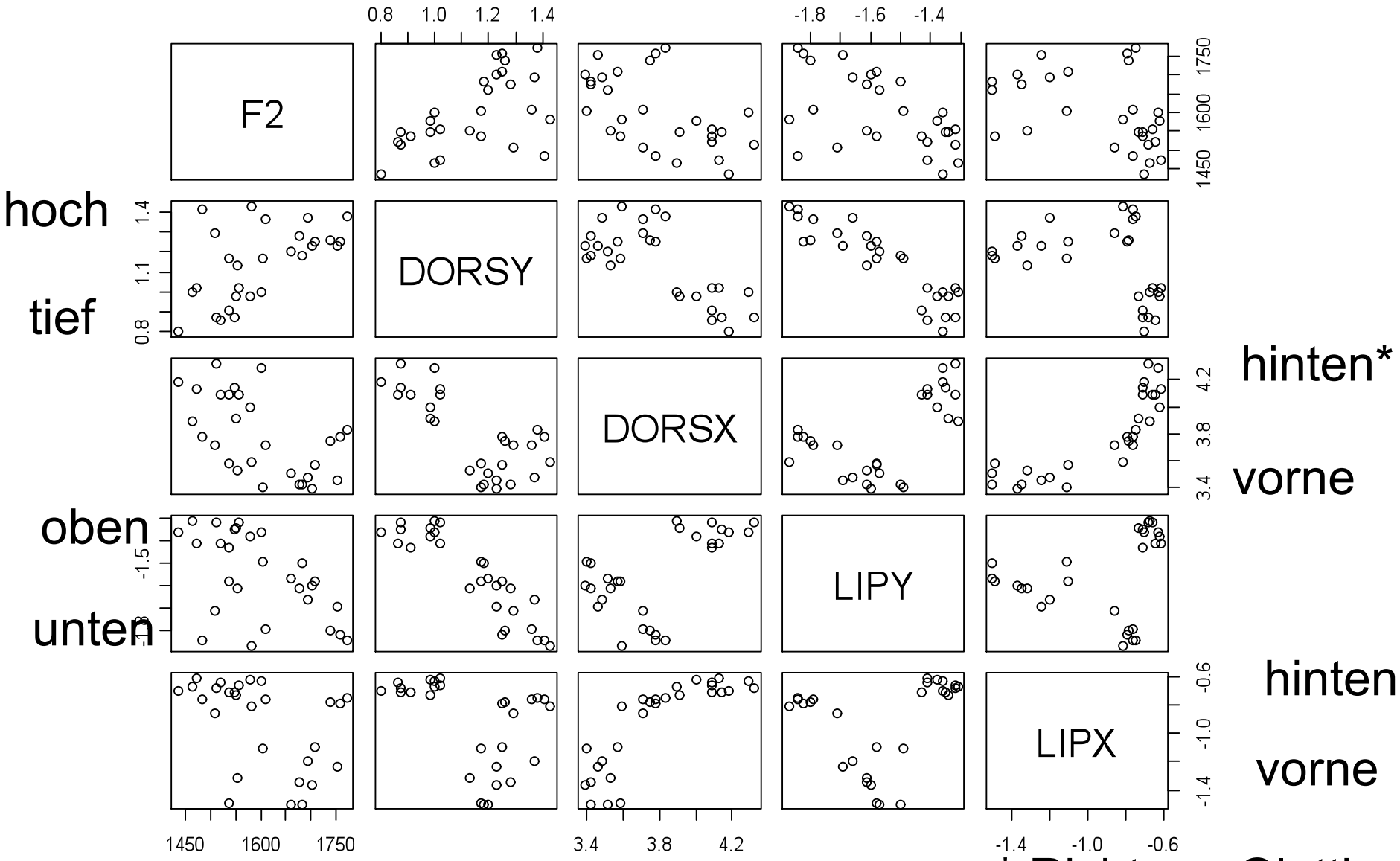
hat die Bedeutung

$\hat{F2}$ = ein Gewicht Mal die horizontale Position der Zunge + ein anderes Gewicht Mal die vertikale Position der Zunge + …. + ein Intercept

Mehrfache Regression in R

- Festlegung von b_1, b_2, b_3, b_4, k
- Sind alle diese Parameter notwendig? Wenn nicht, welches Parameter oder Parameterkombination hat die deutlichste lineare Beziehung zu F2?

pairs(ydata)


F2
DORSY
DORSX
LIPY
LIPX
0.8
1.0
1.2
1.4
-1.8
-1.6
-1.4
1450
1600
1750
1.4
1.1
0.8
4.2
3.8
3.4
-1.5
-0.6
-1.0
-1.4
hoch
tief
hinten*
vorne
oben
unten
hinten
vorne
* Richtung Glottis

$$\hat{F2} = b_1 DORSX + b_2 DORSY + b_3 LIPX + b_4 LIPY + k$$

```
regm = lm(F2 ~ DORSX+DORSY+LIPX+LIPY, data=ydata)
```

Koeffiziente

```
coef(regm)
```

summary(regm)

```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  1355.05     822.63   1.647    0.113
DORSX         -38.33     157.21  -0.244    0.810
DORSY          63.08     254.27   0.248    0.806
LIPX          -67.36     110.90  -0.607    0.550
LIPY         -164.08     205.04  -0.800    0.432

Residual standard error: 83 on 23 degrees of freedom
Multiple R-Squared: 0.3939,     Adjusted R-squared: 0.2884
F-statistic: 3.736 on 4 and 23 DF,  p-value: 0.01747
```

F2 kann mit einer multidimensionalen Regression aus diesen artikulatorischen Parametern modelliert werden:

Adjusted R^2 = 0.29, F[4, 23] = 3.7, $p < 0.05$.

Adjusted R^2

R^2: (siehe vorige Vorlesung) ist die Proportion der Varianz, die durch die Regression erklärt werden kann (variiert zwischen 0 und 1)

> R^2 wird mit einer zunehmenden Anzahl von Regressoren größer.

Daher muss in der Berechnung von R^2 für die Anzahl der Regressoren kompensiert werden, wenn wir - wie in diesem Fall - Regressionslinien mit unterschiedlichen Anzahlen von Regressoren miteinander vergleichen wollen.

Adjusted R^2

- kann auch negativ sein
- ist weniger als R^2

Adjusted R^2 = $$1 - (1 - R^2)\frac{n-1}{n-k-1}$$

n ist die Anzahl der Stichproben, k ist die Anzahl der Regressoren.

Für diesen Fall

```
n = with(ydata, length(F2))
1-(1-0.3939) * (  (n-1)/(n-4-1) )
[1] 0.2884913
```

Modell-Prüfung durch AIC (Akaike's Information Criterion)

Mit der stepAIC() Funktion in library(MASS) wird geprüft, ob für die Regression wirklich alle (in diesem Fall 4) Regressoren benötigt werden.

Je kleiner AIC, umso nützlicher die Kombination für die Regression (umso höher adjusted R^2)

```
library(MASS)

stepAIC(regm)
```

```
Start:  AIC= 251.95
 F2 ~ DORSX + DORSY + LIPX + LIPY
        Df Sum of Sq    RSS    AIC
- DORSX  1       410 158861    250
- DORSY  1       424 158875    250
- LIPX   1      2541 160993    250
- LIPY   1      4412 162863    251
<none>               158451    252
```

sortiert nach AIC. Dies ist der AIC-Wert, wenn aus diesem Modell DORSX weggelassen wäre.

- Vor allem ist dieser AIC Wert weniger als AIC mit allen Parametern (= 251.95).
- Daher wird im nächsten Modell DORSX weggelassen.

```
Step:  AIC= 250.02
 F2 ~ LIPX + LIPY + DORSY

         Df Sum of Sq    RSS    AIC
- DORSY   1      1311 160172    248
- LIPY    1      4241 163102    249
<none>                158861    250
- LIPX    1     16377 175238    251
```

AIC ist am kleinsten, wenn aus diesem Modell DORSY weggelassen wird. Und dieser Wert ohne DORSY ist kleiner als derjenige mit LIPX+LIPY+DORSY zusammen.

Daher wird DORSY weggelassen…

```
Step:  AIC= 248.25
 F2 ~ LIPX + LIPY
       Df Sum of Sq    RSS    AIC
<none>               160172    248
- LIPX  1     25225 185397    250
- LIPY  1     50955 211127    254
```

Wenn wir entweder LIPX oder LIPY weggelassen, dann wird AIC **höher** im Vergleich zu AIC mit beiden Parametern zusammen.

Daher bleiben wir bei **F2 ~ LIPX + LIPY**

Dieses Modell F2 ~ LIPX + LIPY müsste auch den höchsten adjusted R^2 haben. Prüfen, zB:

```
lip.lm = lm(F2 ~ LIPX+
LIPY, data= ydata)
```

```
summary(regm)
```

```
summary(lip.lm)
```

Adjusted R-squared: 0.2884

Adjusted R-squared: 0.3383

Also wird die Variation in F2 in [y] am meisten durch die horizontale und vertikale Position der Unterlippe erklärt.

Polynomiale Regression

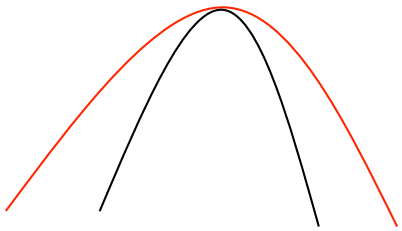
ein Regressor ein Koeffizient

$$\hat{y} = bx + k$$

ein Regressor, 2 Koeffiziente

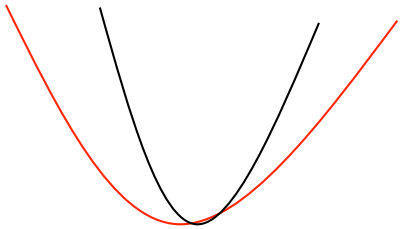
$$\hat{y} = b_1 x + b_2 x^2 + k$$

bestimmt die Krümmung;



b_2 ist negativ

b_2 ist näher an 0



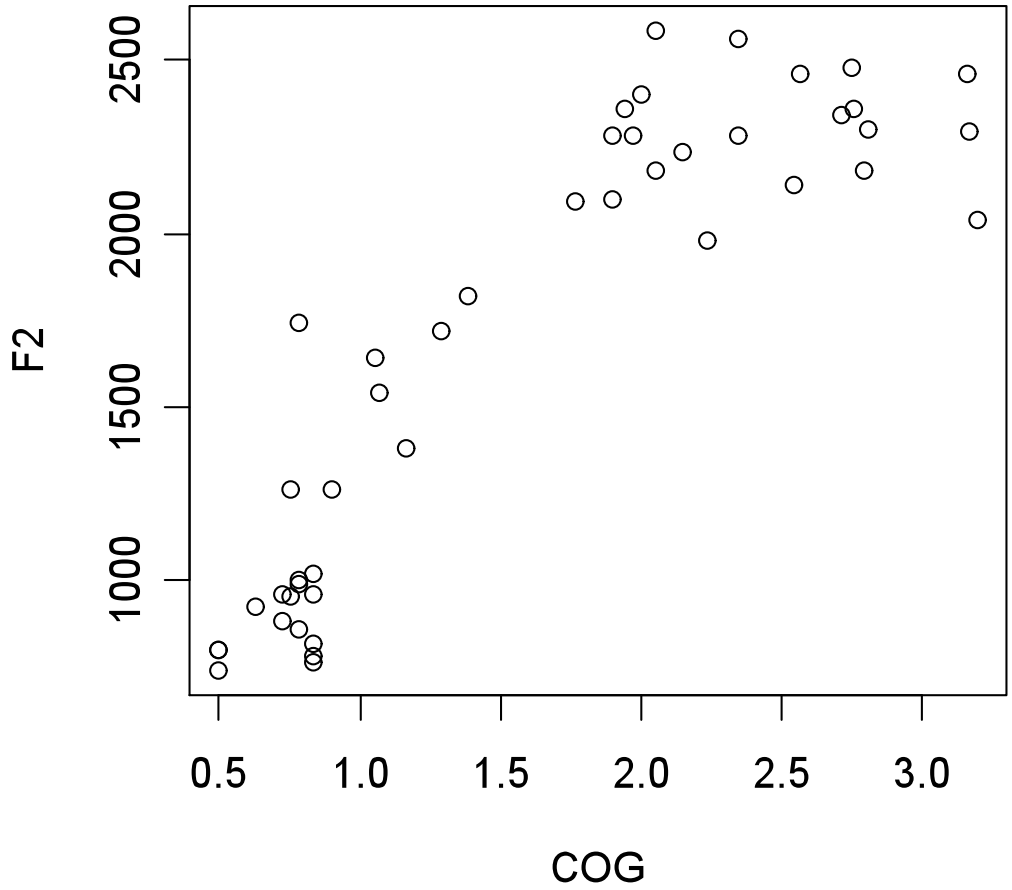
b_2 ist positiv

ein Regressor, n Koeffiziente

$$\hat{y} = b_1 x + b_2 x^2 + b_3 x^3 + b_4 x^4 + \ldots b_{n-1} x^n + k$$

In allen Fällen handelt es sich um Abbildung/ Beziehungen im **2D**-Raum (da wir mit **einem** Regressor zu tun haben).

```
epg = read.table(paste(pfad, "epg.txt", sep="/"))
with(epg, plot(COG, F2))
```



$$\hat{F2} = b_1 COG + b_2 COG^2 + k$$

```
regp = lm(F2 ~ COG + I(COG^2), data = epg)

k = coef(regp)
```

```
(Intercept)         COG    I(COG^2)
  -294.3732   2047.8403   -393.5154
```

$$\hat{F2} = 2047.8\, COG - 393.5\, COG^2 - 294.3732$$

```
with(epg, plot(COG, F2))
```

Die Parabel überlagern

```
curve(k[1] + k[2]*x + k[3]*x^2, add=T)
```

summary(regp)

Beide Komponente, COG und COG^2 der Parabel scheinen notwendig zu sein, um die F2-COG Beziehungen zu modellieren.

```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  -294.37     139.95  -2.103   0.0415 *
COG           2047.84     192.83  10.620 1.81e-13 ***
I(COG^2)      -393.52      54.17  -7.264 6.10e-09 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 '
' 1

Residual standard error: 202.1 on 42 degrees of freedom
Multiple R-Squared: 0.9092,     Adjusted R-squared: 0.9049
F-statistic: 210.4 on 2 and 42 DF,  p-value: < 2.2e-16
```

mit stepAIC() kann wieder festgestellt werden, ob wir den COG^2 Parameter wirklich benötigen:

stepAIC(regp)

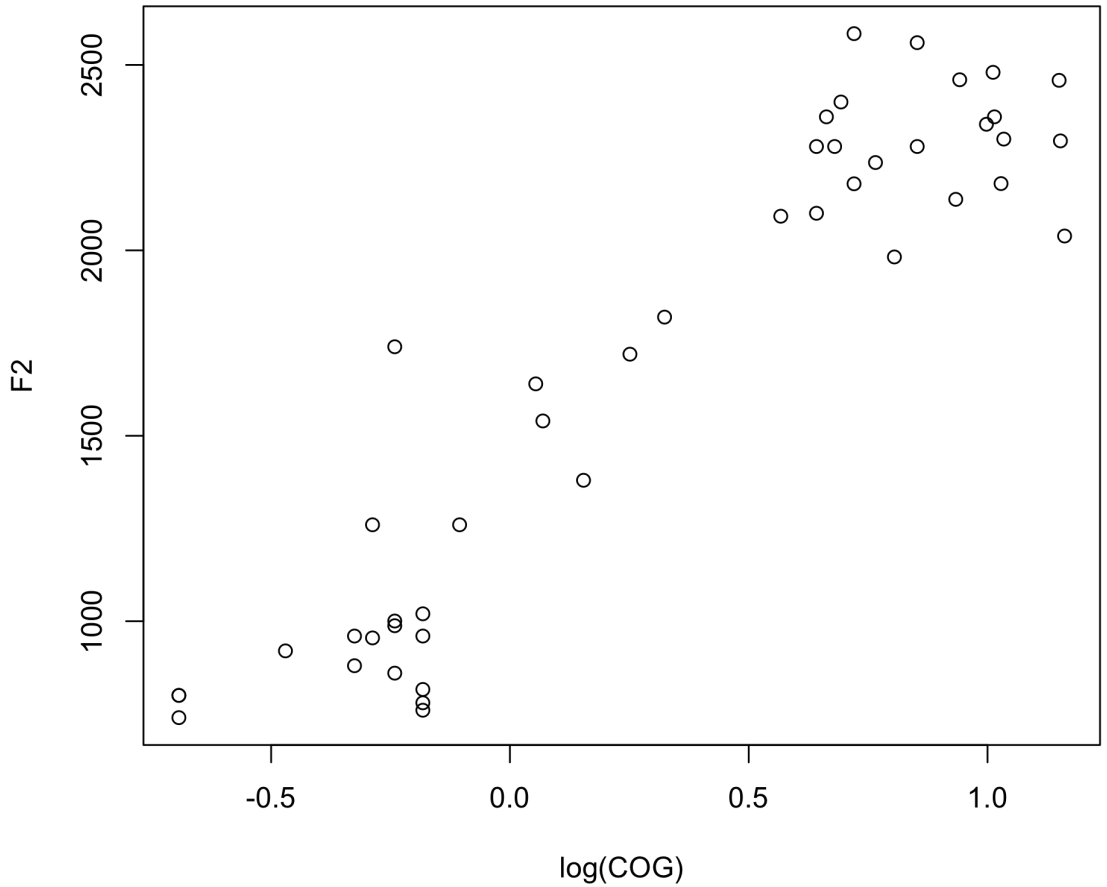
```
Start:  AIC= 480.69
 F2 ~ COG + I(COG^2)

            Df Sum of Sq      RSS      AIC
<none>                     1715550      481
- I(COG^2)   1   2155469 3871019      515
- COG        1   4606873 6322423      537

Call:
lm(formula = F2 ~ COG + I(COG^2))

Coefficients:
(Intercept)          COG     I(COG^2)
     -294.4        2047.8        -393.5
```

Scheinbar ja (da AIC höher wird, wenn entweder COG oder COG^2 aus dem Modell weggelassen werden).

Prüfen ob das Verhältnis logarithmisch ist, zB:

with(epg, plot(log(COG), F2))

Inwiefern kann man das Verhältnis zwischen den Variablen mit einer geraden Linie modellieren?



Bedingungen für die Durchführung einer Regression

siehe vor allem http://www.duke.edu/~rnau/testing.htm und Verzani Kap. 10)

Die wichtigsten Kriterien:

Die Residuals sollen:

(a) von einer Normalverteilung nicht signifikant abweichen.

(b) 'homoscedasticity' oder eine konstante Varianz aufweisen.

(c) keine Autokorrelation aufweisen.

(d) ggf. Ausreißer entfernen

(a) Sind die Residuals normalverteilt?

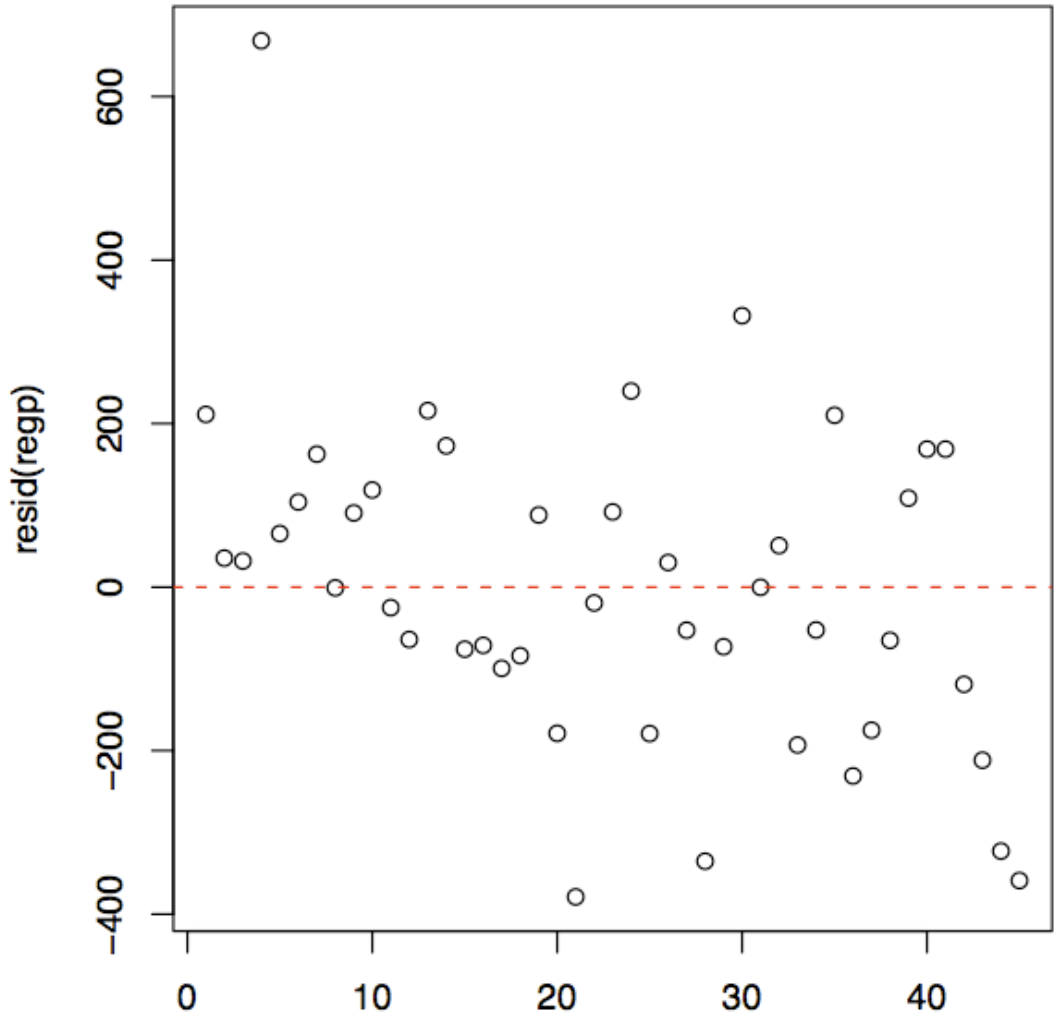
```
regp = lm(F2 ~ COG + I(COG^2), data = epg)

shapiro.test(resid(regp))
```

```
data:  resid(regp)
W = 0.9606, p-value = 0.1285
```

(b) Haben die Residuals eine konstante Varianz?

Insbesondere sollten die residuals nicht wesentlich größer am Anfang/Ende sein, sondern auf eine randomisierte Weise um die 0 Linie verteilt sein.

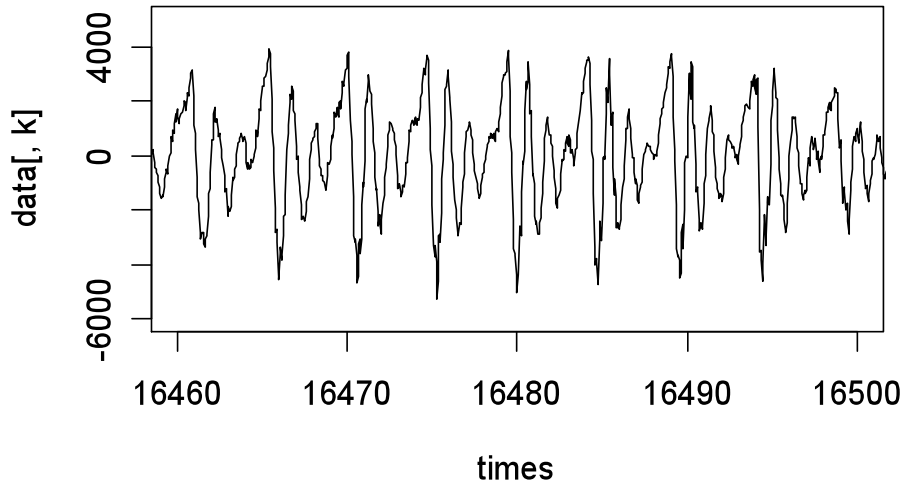
```
plot(resid(regp))

abline(h=0, lty=2)
```



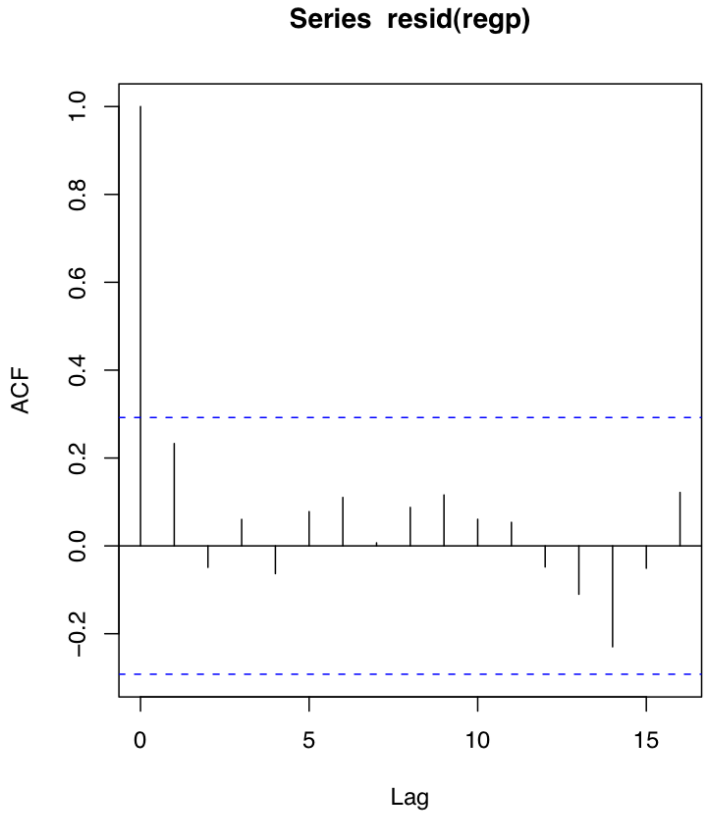
(c) Keine Autokorrelation

Autokorrelation ist wenn die Werte eines Signals mit sich selbst korreliert sind

Ein gutes Beispiel von autokorrelierten Daten: ein stimmfahtes Sprachsignal



Die Residuals sollen keine Autokorrelation aufweisen



acf(resid(regp))

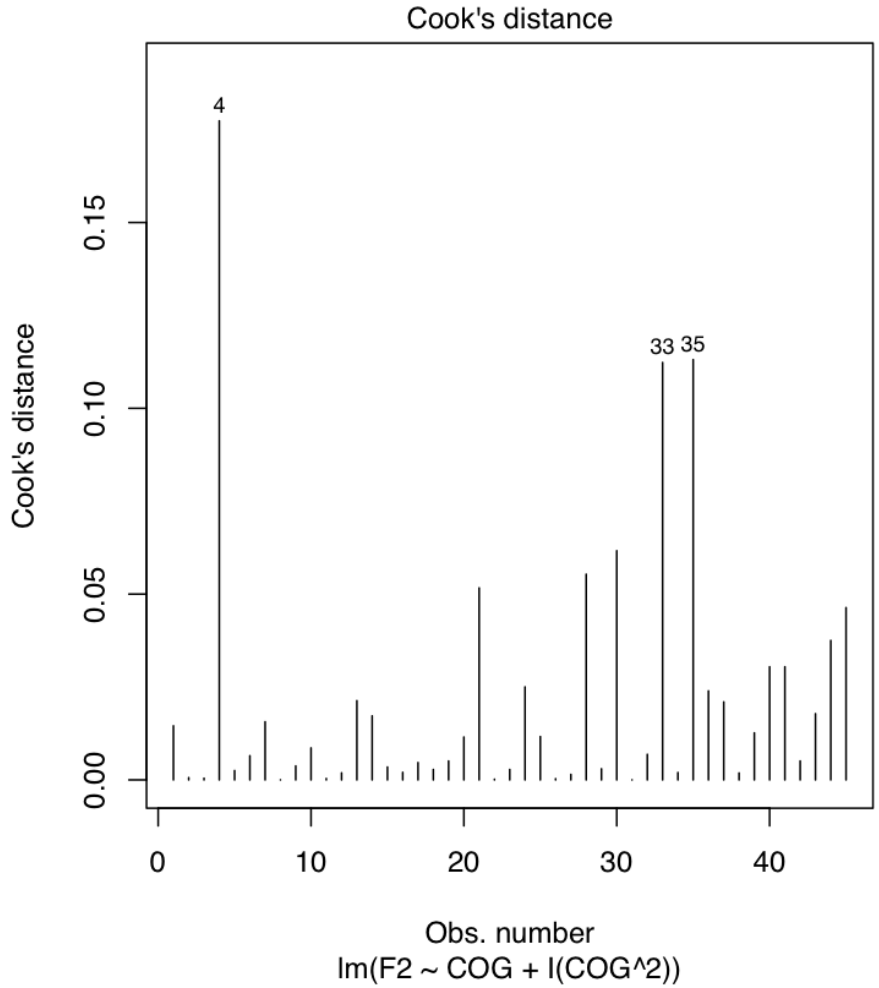
95% Konfidenzintervall um 0

Wenn die meisten ACF-Werte innerhalb der blauen Linien liegen, gibt es **keine Autokorrelation.**

Insbesondere die Werte bei lag 1 und 2 beobachten: diese sollten vor allem innerhalb des Vertauensintervalls liegen.

(d) Ausreißer

Ein einziger Ausreißer vor allem vom Mittelpunkt weit entfernt kann die Regressionslinie deutlich beeinflussen.



plot(regp, 4)

Ausreißer die eventuell (aber nicht unbedingt) die Regressionslinie stark beeinflussen, können mit dem sog. **Cookes Distance** aufgedeckt werden

Die Cookes-Entfernungen und daher Ausreißer könnten auch mit einem sogenannten 'bubble plot' (siehe Verzani) abgebildet werden

```
with(epg, plot(COG, F2, cex = 10*sqrt(cooks.distance(regp))))
with(epg, text(COG, F2, as.character(1:length(F2))))
```