

INTRO DATENANALYSE MIT R - DRITTER TEIL

Jan-Philipp Kolb

09 Mai, 2019

DIE LINEARE REGRESSION

DIE LINEARE REGRESSION

JOHN H. MAINDONALD AND W. JOHN BRAUN - **Data Analysis and Graphics Data and Functions**

- ▶ Einführung in R
- ▶ Datenanalyse
- ▶ Statistische Modelle
- ▶ Inferenzkonzepte
- ▶ Regression mit einem Prädiktor
- ▶ Multiple lineare Regression
- ▶ Ausweitung des linearen Modells
- ▶ ...

LINEARE REGRESSION IN R - BEISPIELDATENSATZ

```
data(mtcars)
```

HILFE FÜR DEN MTCARS DATENSATZ:

```
?mtcars
```

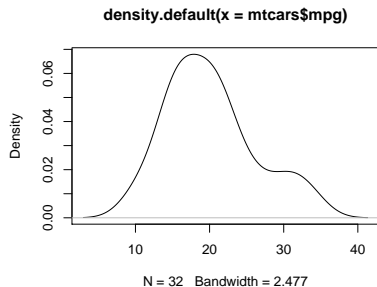
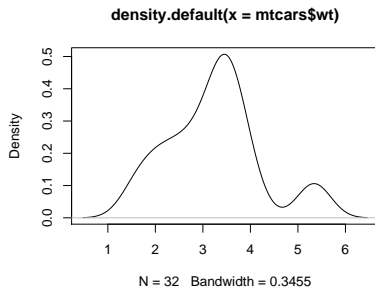
	mpg	cyl	disp	hp	drat	wt	model
21.0	6	160	110	3.90	2.620	Mazda RX4	
21.0	6	160	110	3.90	2.875	Mazda RX4 Wag	
22.8	4	108	93	3.85	2.320	Datsun 710	
21.4	6	258	110	3.08	3.215	Hornet 4 Drive	
18.7	8	360	175	3.15	3.440	Hornet Sportabout	
18.1	6	225	105	2.76	3.460	Valiant	

VARIABLEN DES MTCARS DATENSATZES

- ▶ mpg - Miles/(US) gallon
- ▶ cyl - Number of cylinders
- ▶ disp - Displacement (cu.in.)
- ▶ hp - Gross horsepower
- ▶ drat - Rear axle ratio
- ▶ wt - Weight (1000 lbs)
- ▶ qsec - 1/4 mile time
- ▶ vs - Engine (0 = V-shaped, 1 = straight)
- ▶ am - Transmission (0 = automatic, 1 = manual)
- ▶ gear - Number of forward gears
- ▶ carb - Number of carburetors

VERTEILUNGEN FÜR ZWEI VARIABLEN VON MTCARS

```
par(mfrow=c(1,2))  
plot(density(mtcars$wt)); plot(density(mtcars$mpg))
```



EIN EINFACHES REGRESSIONSMODELL

ABHÄNGIGE VARIABLE - MEILEN PRO GALLONE (MPG)

UNABHÄNGIGE VARIABLE - GEWICHT (WT)

```
m1 <- lm(mpg ~ wt,data=mtcars)
```

```
m1
```

```
##
```

```
## Call:
```

```
## lm(formula = mpg ~ wt, data = mtcars)
```

```
##
```

```
## Coefficients:
```

```
## (Intercept)          wt
```

```
##      37.285      -5.344
```

DIE MODELLFORMEL

MODELL OHNE ACHSENABSCHNITT

```
m2 <- lm(mpg ~ - 1 + wt,data=mtcars)
summary(m2)$coefficients
```

```
##      Estimate Std. Error  t value    Pr(>|t|)
## wt  5.291624   0.5931801  8.920771 4.55314e-10
```

WEITERE VARIABLEN HINZUFÜGEN

```
m3 <- lm(mpg ~ wt + cyl,data=mtcars)
summary(m3)$coefficients
```

```
##              Estimate Std. Error  t value    Pr(>|t|)
## (Intercept) 39.686261   1.7149840 23.140893 3.043182e-20
## wt          -3.190972   0.7569065 -4.215808 2.220200e-04
## cyl          -1.507795   0.4146883 -3.635972 1.064282e-03
```


SUMMARY DES MODELLS

```
summary(m3)
```

```
##
## Call:
## lm(formula = mpg ~ wt + cyl, data = mtcars)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -4.2893 -1.5512 -0.4684  1.5743  6.1004
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   39.6863     1.7150  23.141  < 2e-16 ***
## wt           -3.1910     0.7569  -4.216  0.000222 ***
## cyl          -1.5078     0.4147  -3.636  0.001064 **
## ---
```

R ARBEITET MIT OBJEKTEN

- ▶ m3 ist nun ein spezielles Regressions-Objekt
- ▶ Auf dieses Objekt können nun verschiedene Funktionen angewendet werden

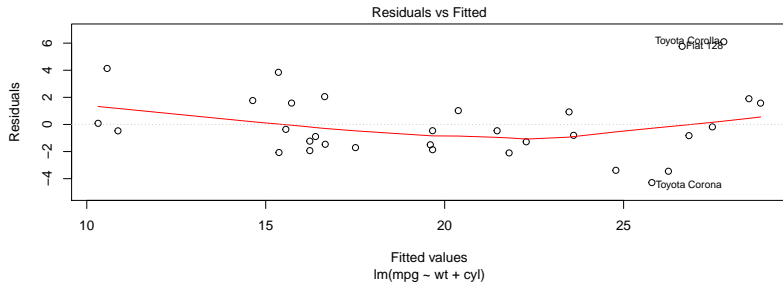
```
predict(m3) # Vorhersage
```

##	Mazda RX4	Mazda RX4 Wag	Datsun
##	22.27914	21.46545	26.25
##	Hornet 4 Drive	Hornet Sportabout	Vali
##	20.38052	16.64696	19.59
##	Duster 360	Merc 240D	Merc
##	16.23213	23.47588	23.60
##	Merc 280	Merc 280C	Merc 45
##	19.66255	19.66255	14.63
##	Merc 450SL	Merc 450SLC	Cadillac Fleetw
##	15.72158	15.56203	10.87

RESIDUENPLOT

- ▶ Sind Annahmen des linearen Regressionsmodells verletzt?
- ▶ Dies ist der Fall, wenn ein Muster abweichend von einer Linie zu erkennen ist. (Hier ist der Datensatz sehr klein)

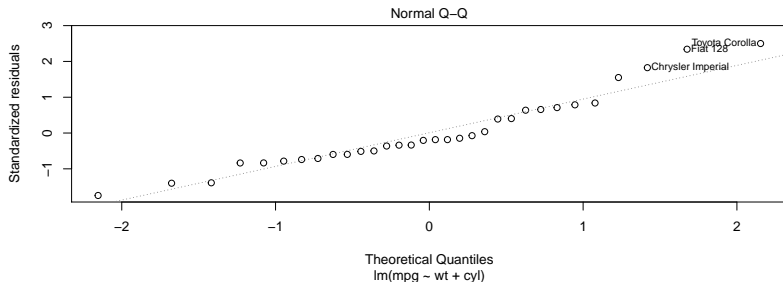
```
plot(m3,1)
```



RESIDUENPLOT

- ▶ Wenn Residuen normalverteilt sind sollten sie auf Linie sein.

```
plot(m3,2)
```



WEITERE MÖGLICHKEITEN DIE FORMEL ZU SPEZIFIZIEREN

INTERAKTIONSEFFEKT

```
# effect of cyl and interaction effect:  
m3a<-lm(mpg~wt*cyl,data=mtcars)  
  
# only interaction effect:  
m3b<-lm(mpg~wt:cyl,data=mtcars)
```

DEN LOGARITHMUS NEHMEN

```
m3d<-lm(mpg~log(wt),data=mtcars)
```

EIN MODELL MIT INTERAKTIONSEFFEKT

DISP - HUBRAUM

```
m3d<-lm(mpg~wt*disp,data=mtcars)
m3dsum <- summary(m3d)
m3dsum$coefficients
```

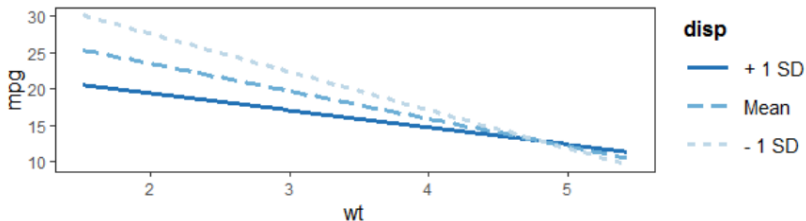
	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	44.08199770	3.123062627	14.114990	2.955567e-14
wt	-6.49567966	1.313382622	-4.945763	3.216705e-05
disp	-0.05635816	0.013238696	-4.257078	2.101721e-05
wt:disp	0.01170542	0.003255102	3.596022	1.226988e-03

INTERAKTIONEN UNTERSUCHEN

```
install.packages("jtools")
```

```
library(jtools)  
interact_plot(m3d, pred = "wt", modx = "disp")
```

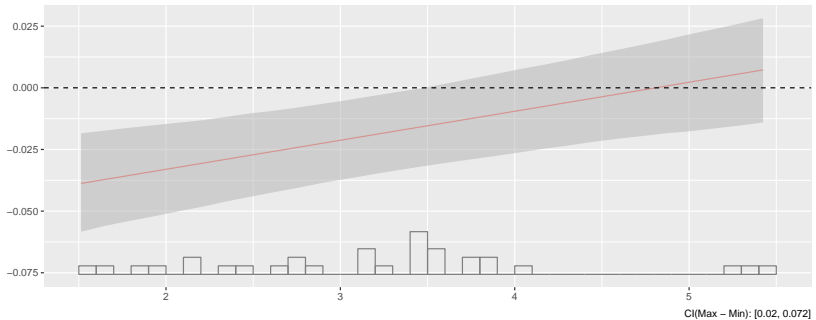
- ▶ Mit einem kontinuierlichen Moderator (in unserem Fall Disp) erhält man drei Zeilen - 1 Standardabweichung über und unter dem Mittelwert und der Mittelwert selbst.



DAS PAKET INTERPLOT

```
library(interplot)
```

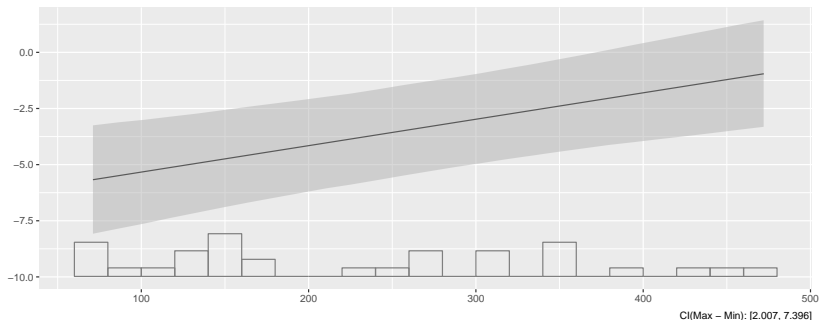
```
interplot(m = m3d, var1 = "disp", var2 = "wt", hist = TRUE,  
  aes(color = "pink") + theme(legend.position="none") +  
  geom_hline(yintercept = 0, linetype = "dashed"))
```



NOCH EIN INTERPLOT

- Effekt wird auf die y-Achse geplottet - wt auf der x-Achse

```
interplot(m = m3d, var1 = "wt", var2 = "disp", hist = TRUE)
```



- Eine detailliertere Beschreibung ist in der **interplot Vignette** zu bekommen.

BEISPIEL: OBJEKTORIENTIERUNG

- ▶ m3 ist nun ein spezielles Regressionsobjekt
- ▶ Verschiedene Funktionen können auf dieses Objekt angewendet werden

```
predict(m3) # Prediction  
resid(m3)  # Residuals
```

##	Mazda RX4	Mazda RX4 Wag	Datsun 710
##	22.27914	21.46545	26.25203
##	Hornet Sportabout	Valiant	
##	16.64696	19.59873	
##	Mazda RX4	Mazda RX4 Wag	Datsun 710
##	-1.2791447	-0.4654468	-3.4520262
##	Hornet Sportabout	Valiant	
##	2.0530424	-1.4987281	

EINE MODELLVORHERSAGE MACHEN

```
pre <- predict(m1)
head(mtcars$mpg)
```

```
## [1] 21.0 21.0 22.8 21.4 18.7 18.1
```

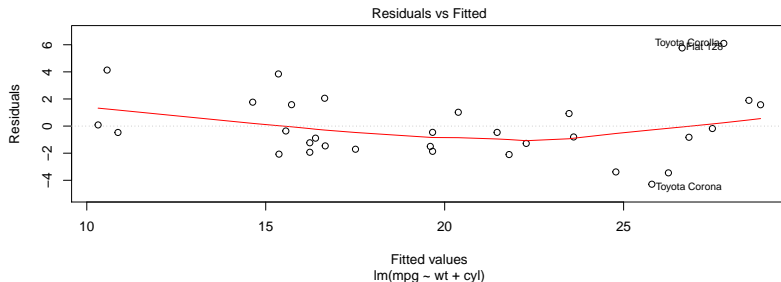
```
head(pre)
```

##	Mazda RX4	Mazda RX4 Wag	Datsun 710
##	23.28261	21.91977	24.88595
##	Hornet Sportabout	Valiant	
##	18.90014	18.79325	

RESIDUENPLOT - MODELLANNAHMEN VERLETZT?

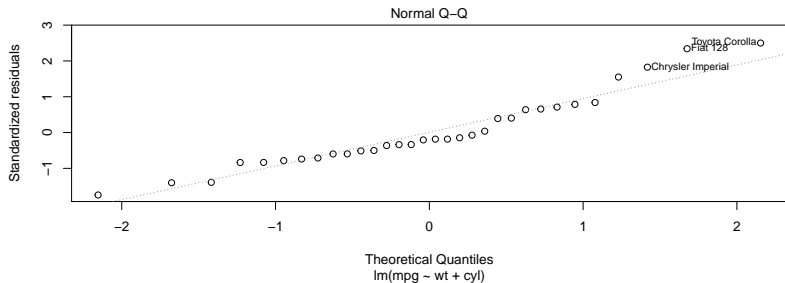
- Gibt es ein Muster in der Abweichung von der Linie

```
plot(m3,1)
```



RESIDUENPLOT

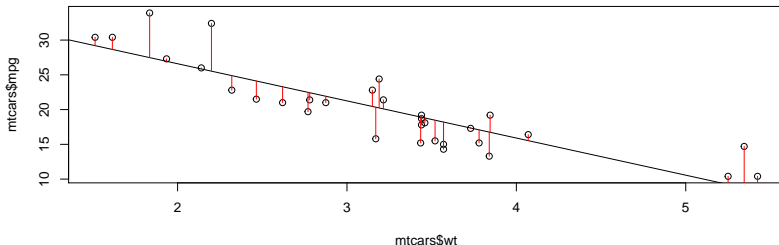
```
plot(m3,2)
```



- Bei Normalverteilung liegen Residuen auf gleicher Linie

REGRESSIONSDIAGNOSTIK MIT BASIS-R

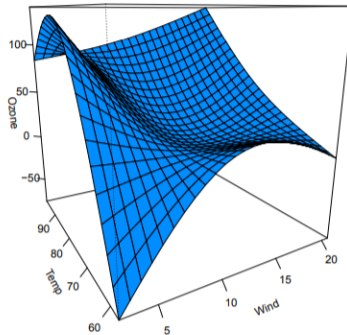
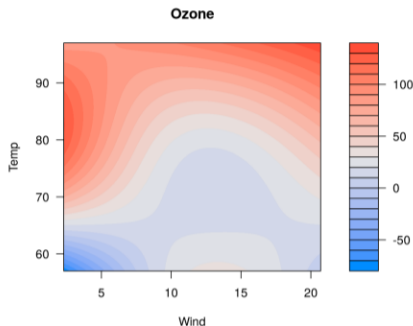
```
plot(mtcars$wt,mtcars$mpg)
abline(m1)
segments(mtcars$wt, mtcars$mpg, mtcars$wt, pre, col="red")
```



DAS VISREG-PAKET

```
install.packages("visreg")
```

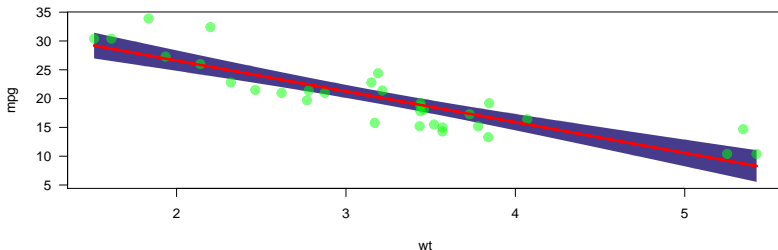
```
library(visreg)
```



DAS VISREG-PAKET

- ▶ Das Default-Argument für `type` ist `conditional`.
- ▶ Scatterplot von `mpg` und `wt` mit Regressionslinie und Konfidenzbändern

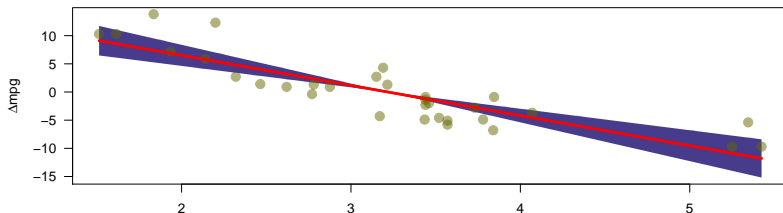
```
visreg(m1, "wt", type = "conditional")
```



Visualisierung mit visreg

- ▶ Zweites Argument - Spezifikation der Kovariaten in der Graphik
- ▶ Das Diagramm zeigt die Auswirkung auf den erwarteten Wert des Regressors, wenn die Variable x von einem Referenzpunkt auf der x -Achse wegbewegt wird (bei numerischen Variablen der Mittelwert).

```
visreg(m1, "wt", type = "contrast")
```



REGRESSION MIT FAKTOREN

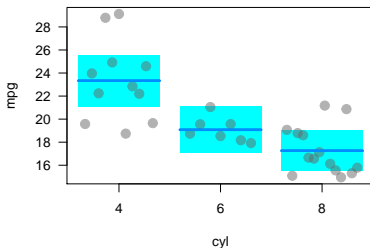
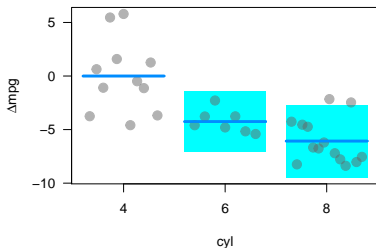
- ▶ Die Effekte von Faktoren können auch mit visreg visualisiert werden:

```
mtcars$cyl <- as.factor(mtcars$cyl)
m4 <- lm(mpg ~ cyl + wt, data = mtcars)
# summary(m4)
```

##	Estimate	Std. Error	t value	Pr(> t)
## (Intercept)	33.990794	1.8877934	18.005569	6.257246e-17
## cyl6	-4.255582	1.3860728	-3.070244	4.717834e-03
## cyl8	-6.070860	1.6522878	-3.674214	9.991893e-04
## wt	-3.205613	0.7538957	-4.252065	2.130435e-04

EFFEKTE VON FAKTOREN

```
par(mfrow=c(1,2))  
visreg(m4, "cyl", type = "contrast")  
visreg(m4, "cyl", type = "conditional")
```



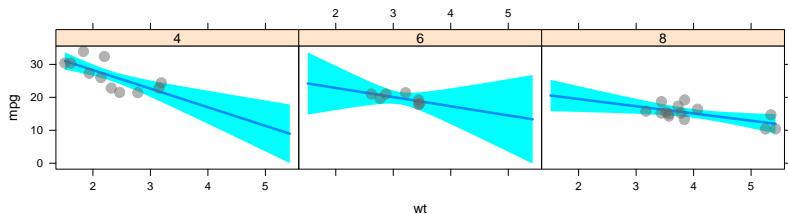
DAS PAKET VISREG - INTERAKTIONEN

```
m5 <- lm(mpg ~ cyl*wt, data = mtcars)
# summary(m5)
```

##	Estimate	Std. Error	t value	Pr(> t)
## (Intercept)	39.571196	3.193940	12.3894599	2.058359e-1
## cyl6	-11.162351	9.355346	-1.1931522	2.435843e-0
## cyl8	-15.703167	4.839464	-3.2448150	3.223216e-0
## wt	-5.647025	1.359498	-4.1537586	3.127578e-0
## cyl6:wt	2.866919	3.117330	0.9196716	3.661987e-0
## cyl8:wt	3.454587	1.627261	2.1229458	4.344037e-0

DEN GRAPHIKOUTPUT MIT LAYOUT KONTROLLIEREN

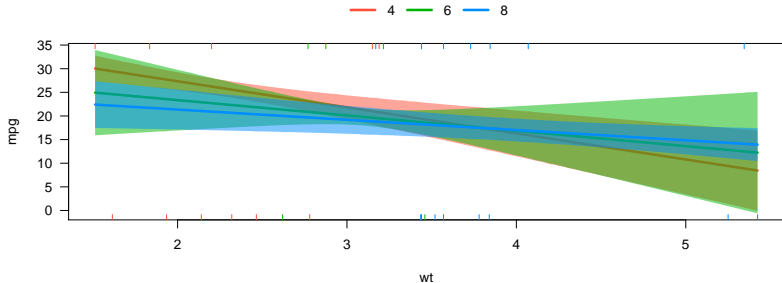
```
visreg(m5, "wt", by = "cyl", layout=c(3,1))
```



DAS PAKET `visreg` - INTERAKTIONSEFFEKTE ÜBEREINANDER LEGEN

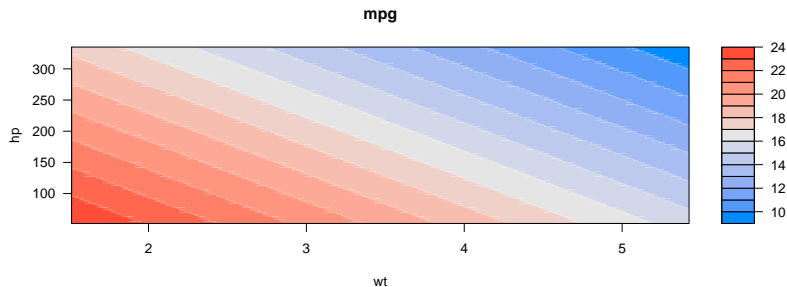
```
m6 <- lm(mpg ~ hp + wt * cyl, data = mtcars)
```

```
visreg(m6, "wt", by="cyl", overlay=TRUE, partial=FALSE)
```



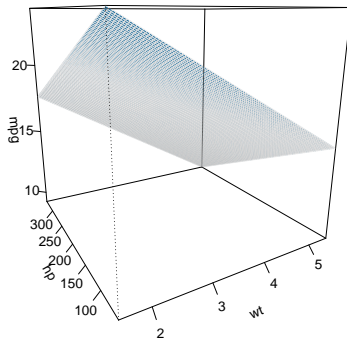
DAS PAKET VISREG - VISREG2D

```
visreg2d(m6, "wt", "hp", plot.type = "image")
```



DAS PAKET VISREG - SURFACE

```
visreg2d(m6, "wt", "hp", plot.type = "persp")
```



B3A AUFGABE LINEARE REGRESSION

Der Datensatz `toycars` beschreibt die Route von drei Spielzeugautos, die Rampen in verschiedenen Winkeln runterfahren.

- ▶ `angle`: Rampenwinkel
 - ▶ `distance`: Entfernung die von dem Spielzeugauto zurück gelegt wird.
 - ▶ `car`: Autotyp (1, 2 or 3)
- A) Lese den Datensatz `toycars` ein und konvertiere die Variable `car` des Datensatzes in einen Faktor (`as.factor`).
- (B) Erstelle drei Box-Plots, in denen die von den Autotypen zurückgelegte Strecke visualisiert wird.

AUFGABE LINEARE REGRESSION II

- (c) Schätze für jeden Autotyp die Parameter des folgenden linearen Modell; nutze dafür die Funktion `lm()`

$$distance_i = \beta_0 + \beta_1 \cdot angle_i + \epsilon_i$$

- (d) Überprüfe die Anpassung des Modells indem Du die drei Regressionslinien in den Scatterplot einzeichnest (`distance` gegen `angle`). Spricht das R^2 für eine gute Modellanpassung?

EINEN SCHÖNEN OUTPUT MIT DEM PAKET **stargazer** erzeugen

```
library(stargazer)
stargazer(m3, type="html")
```

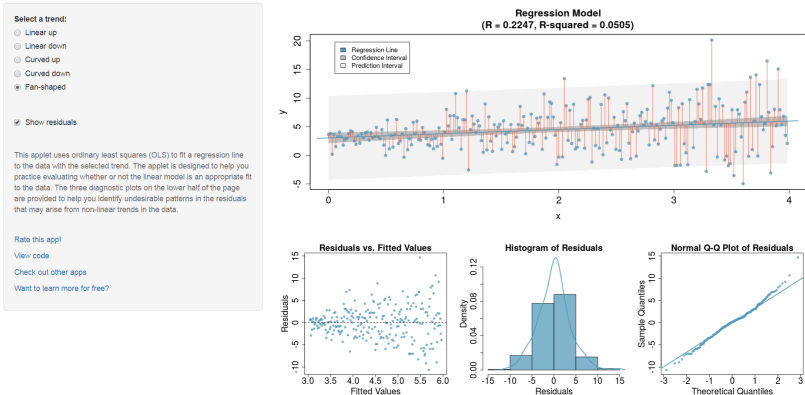
BEISPIEL HTML OUTPUTS:

	<i>Dependent variable:</i>
	mpg
wt	-3.125*** (0.911)
cyl	-1.510*** (0.422)
am	0.176 (1.304)
Constant	39.418***

SHINY APP - DIAGNOSTIKEN FÜR DIE EINFACHE LINEARE REGRESSION

https://gallery.shinyapps.io/slr_diag/

Diagnostics for simple linear regression



LINKS - LINEARE REGRESSION

- ▶ Regression - **r-bloggers**
- ▶ Das komplette Buch von **Faraway**- sehr intuitiv geschriebenes Buch
- ▶ Gute Einführung auf **Quick-R**
- ▶ **Multiple Regression**
- ▶ **15 Arten von Regressionen die man kennen sollte**
- ▶ **ggeffects** - Erzeuge saubere Datensätze mit marginellen Effekten für 'ggplot' aus Modell Outputs