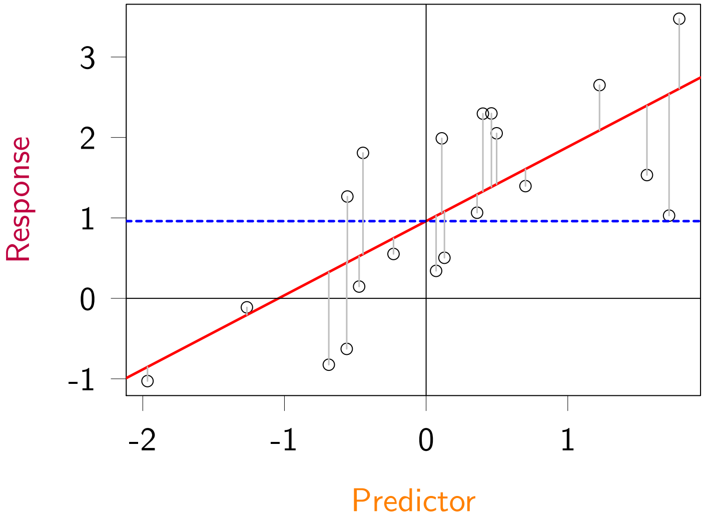
What on earth is going on with my linear models??! Slides with solutions to exercises

March 13, 2018

1. Linear model, reminder
2. Diagnostics
3. A puzzling but simple problem: Over-fit and collinearity
4. Heteroschedasticity: the spooky word

A simple linear model

Response = **Intercept** + **Slope** × **Predictor** + **Error**



A simple linear model

Response = **Intercept** + **Slope** × **Predictor** + **Error**

In R:

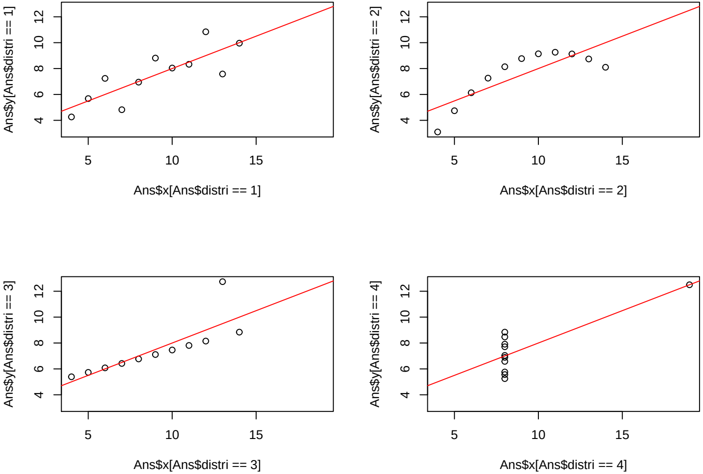
```
lm(response ~ 1 + predictor1 + predictor2, data=data)
```

- Intercept can be explicit or implicit
- Can remove intercept with ...$\sim$ `0 + ...`
- Error is implicit
- Feed the option `data=` to keep code short, reliable and flexible
- Order of predictors do not matter

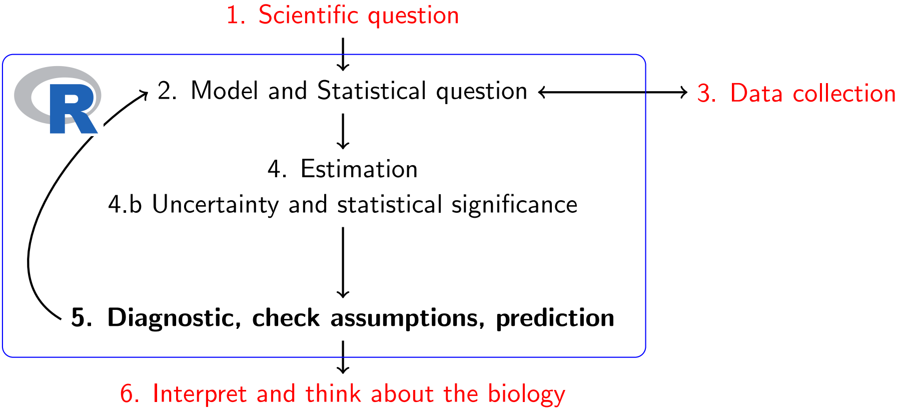
1. Linear model, reminder
2. Diagnostics
3. A puzzling but simple problem: Over-fit and collinearity
4. Heteroschedasticity: the spooky word

Why we need checks: summary(lm) isn't enough

```r
Ans <- read.csv(file = "Anscombe.csv")
```



General approach



1. Scientific question

2. Model and Statistical question

3. Data collection

4. Estimation

4.b Uncertainty and statistical significance

5. Diagnostic, check assumptions, prediction

6. Interpret and think about the biology

Linear model basic assumptions

Not necessarily wrong, but typical interpretation assumes:

- Linear combination of parameters (including transformation, polynoms, interactions. . .)
  Risk: biologically meaningless; e.g. static allometry

Linear model basic assumptions

Not necessarily wrong, but typical interpretation assumes:

- Linear combination of parameters (including transformation, polynoms, interactions...)
  Risk: biologically meaningless; e.g. static allometry
- Predictor not perfectly correlated
  Risk: Model won't run, unstable convergence, or huge SE

Linear model basic assumptions

Not necessarily wrong, but typical interpretation assumes:

- Linear combination of parameters (including transformation, polynoms, interactions. . .)
  Risk: biologically meaningless; e.g. static allometry
- Predictor not perfectly correlated
  Risk: Model won't run, unstable convergence, or huge SE
- Measurement error in predictors
  Risk: bias estimates (underestimate with Gaussian error)

Linear model basic assumptions

Not necessarily wrong, but typical interpretation assumes:

- Linear combination of parameters (including transformation, polynoms, interactions. . .)
  Risk: biologically meaningless; e.g. static allometry
- Predictor not perfectly correlated
  Risk: Model won't run, unstable convergence, or huge SE
- Measurement error in predictors
  Risk: bias estimates (underestimate with Gaussian error)
- Gaussian error distribution
  Risk: Poor predictions

Linear model basic assumptions

Not necessarily wrong, but typical interpretation assumes:

- Linear combination of parameters (including transformation, polynoms, interactions. . .)
  Risk: biologically meaningless; e.g. static allometry
- Predictor not perfectly correlated
  Risk: Model won't run, unstable convergence, or huge SE
- Measurement error in predictors
  Risk: bias estimates (underestimate with Gaussian error)
- Gaussian error distribution
  Risk: Poor predictions
- Homoscedasticity (constant error variance)
  Risk: Over-optimistic uncertainty, unreliable predictions

Linear model basic assumptions

Not necessarily wrong, but typical interpretation assumes:

- Linear combination of parameters (including transformation, polynoms, interactions. . .)
  Risk: biologically meaningless; e.g. static allometry
- Predictor not perfectly correlated
  Risk: Model won't run, unstable convergence, or huge SE
- Measurement error in predictors
  Risk: bias estimates (underestimate with Gaussian error)
- Gaussian error distribution
  Risk: Poor predictions
- Homoscedasticity (constant error variance)
  Risk: Over-optimistic uncertainty, unreliable predictions
- Independence of error
  Risk: Bias and over-optimistic uncertainty

Why we need checks: missing a relationship

```r
forprediction <- read.csv(file = "forprediction.csv")
```

Does ”predictor” predict ”obs”?

Why we need checks: missing a relationship

```r
forprediction <- read.csv(file = "forprediction.csv")
```

Does ”predictor” predict ”obs”?

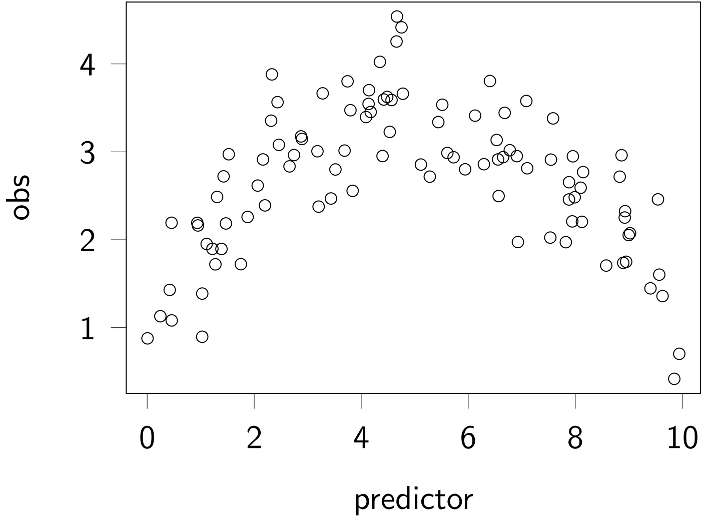
```r
 summary(lm(obs ~ 1 + predictor, data=forprediction) )
```

Why we need checks: missing a relationship

Does "predictor" predict "obs"? Apparently not:

```
  summary(lm(obs ~ 1 + predictor, data=forprediction) )



Call:
lm(formula = obs ~ 1 + predictor, data = forprediction)

Residuals:
    Min      1Q  Median      3Q     Max 
-2.1962 -0.5326  0.1378  0.5785  1.8664 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  2.72530    0.16953  16.076   <2e-16 ***
predictor   -0.01129    0.02956  -0.382    0.703    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.8382 on 98 degrees of freedom
Multiple R-squared:  0.001487,Adjusted R-squared:  -0.008702 
F-statistic: 0.1459 on 1 and 98 DF,  p-value: 0.7033
```

Why we need checks: missing a relationship

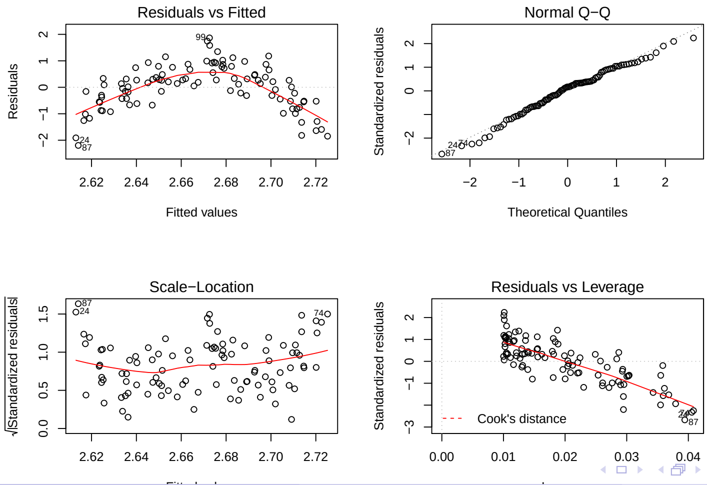


How to check?

```
m0 <- lm(obs ~ 1 + predictor, data=forprediction)
summary(m0)


Call:
lm(formula = obs ~ 1 + predictor, data = forprediction)

Residuals:
    Min      1Q  Median      3Q     Max
-2.1962 -0.5326  0.1378  0.5785  1.8664

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.72530    0.16953  16.076   <2e-16 ***
predictor   -0.01129    0.02956  -0.382    0.703
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.8382 on 98 degrees of freedom
Multiple R-squared:  0.001487,Adjusted R-squared:  -0.008702
F-statistic: 0.1459 on 1 and 98 DF,  p-value: 0.7033
```

How to check?

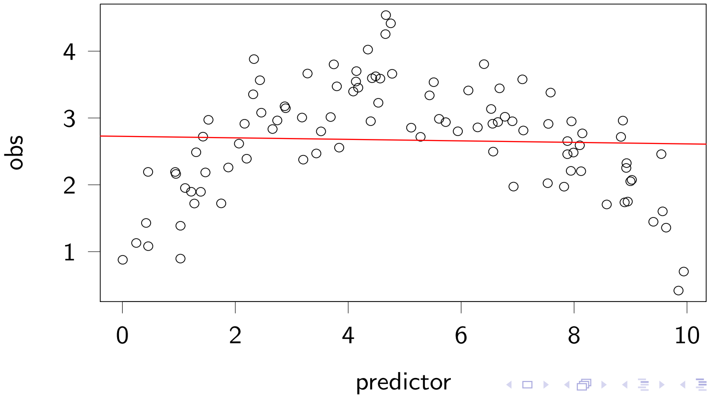
```
par(mfrow=c(2,2))
  plot(m0)
```



How to check?

```
m0 <- lm(obs ~ 1 + predictor, data=forprediction)
```

```
    setPar()
plot(x=forprediction$predictor, y=forprediction$obs, xlab="predictor", ylab
abline(m0, col="red", lwd=3) #simple prediction, without SE
```



Check checklist

- **Visualize your data**
- Residual in summary(): are they symmetrical?
- plot(lm):
  1. trend residual/fitted?
  2. Normal residuals?
  3. trend in residual variance?
  4. outliers?
- Predictions: range and biological meaning

Fix?

Plot suggests a quadratic relationship

```
lm(obs ~ 1 + predictor , data=forprediction)
```

Fix?

Plot suggests a quadratic relationship

```
lm(obs ~ 1 + predictor , data=forprediction)
```

```
m1 <- lm(obs ~ 1 + predictor + I(predictor^2), data=forprediction)
plot(m1)
```

How about prediction? (abline(m1) won't work here because not straight line)

Introduction to prediction

```
m1 <- lm(obs ~ 1 + predictor + I(predictor^2), data=forprediction)
coef(m1)

   (Intercept)      predictor I(predictor^2) 
    1.07782598     0.94760312    -0.09524182 
```

Exercise

1. Write mathematically the relationship between obs and predictor
2. Input regression coefficients in there to predict "obs" from "predictor"
3. Add a prediction line on the plot obs/predictor Is the fit satisfactory?

Introduction to prediction: solution

Write mathematically the relationship between obs and predictor

$$\text{obs} = \beta_0 + \beta_1 \text{predictor} + \beta_2 \text{predictor}^2 + \epsilon$$

where the β are the coefficients of the linear model, and ϵ represent residuals.
Input regression coefficients in there to predict "obs" from "predictor"

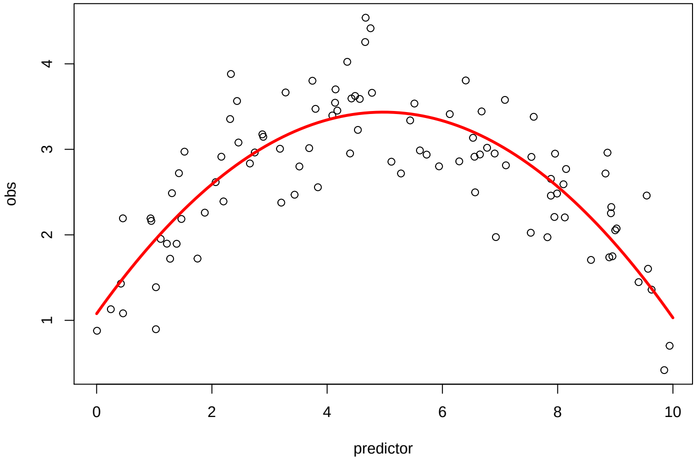
```
  xx <- seq(from=0, to = 10, length.out = 100)
predxx <- coef(m1)[1] + xx * coef(m1)[2] + (xx^2) * coef(m1)[3]
```

Add a prediction line on the plot obs/predictor

```
plot(x=forprediction$predictor, y=forprediction$obs, xlab="predictor",
 ylab="obs")
lines(x = xx, y=predxx, col="red", lwd=3)
```

The fit looks pretty good.

Introduction to prediction



Relaxation slide

This is a snow vole, I used to spend time squeezing them in Switzerland. They gave me a diploma for that.

1 Linear model, reminder

2 Diagnostics

3 A puzzling but simple problem: Over-fit and collinearity

4 Heteroschedasticity: the spooky word

Over-fit and collinearity

Small exercise

Load Cdata.csv, fit models of y predited by x1 and x2, or x2 and x3.
Something is weird, what is going on? What to do?

Over-fit and collinearity: Solution

```
cdata <- read.csv(file = "Cdata.csv")
summary(lm(y ~ x1 + x2, data=cdata))
summary(lm(y ~ x2 + x3, data=cdata))
```

One regression gives NA for the second coefficient, the other regression gives ridiculously large estimates and standard errors. In the former case, the two predictors are perfectly correlated and a regression cannot disentangle their respective effects. In the latter case, they are almost perfectly correlated, there is a tiny bit of information about their independent effects, but not enough to get good estimates. See the correlations:

```
  plot(cdata$x1, cdata$x2)
  plot(cdata$x2, cdata$x3)
```

In both cases you should drop one on the two predictors; they contain the same information. The models give logical answers and good predictions, but there is nothing to be learned about the independent effects of the two predictors and therefore the models are not biologically interesting.

1 Linear model, reminder

2 Diagnostics

3 A puzzling but simple problem: Over-fit and collinearity

4 Heteroschedasticity: the spooky word

Exponential data

Load the dataset htrsdt.csv.

```r
plot(htrsdt$x, htrsdt$obs, ylim = c(-20, max(obs)))
abline(lm(obs ~ x, data=htrsdt))
summary(lm(obs ~ x, data=htrsdt))
plot(lm(obs ~ x, data=htrsdt))
```

Exponential data: prediction

Make a prediction over the range of x, with prediction interval

Exponential data: prediction

Make a prediction over the range of x, with prediction interval ”By-hand”

```r
lmhtrsdt <- lm(obs ~ x, data=htrsdt)
x <- seq(from=min(x), to=max(x), length.out = 100)
predhtrsdt <- coef(lmhtrsdt)[1] + x * coef(lmhtrsdt)[2]
```

Exponential data: prediction

Make a prediction over the range of x, with prediction interval ”By-hand”

```r
lmhtrsdt <- lm(obs ~ x, data=htrsdt)
x <- seq(from=min(x), to=max(x), length.out = 100)
predhtrsdt <- coef(lmhtrsdt)[1] + x * coef(lmhtrsdt)[2]
```

Function predict

```r
Xnewdata <- data.frame(x=seq(from=min(x), to=max(x),
                             length.out = 100))
Xpred <- predict(object =  lm(obs ~ x, data=htrsdt), newdata = Xnewdata,
                 se.fit = TRUE, interval = "prediction")
```

Exponential data: prediction

Make a prediction over the range of x, with prediction interval ”By-hand”

```
lmhtrsdt <- lm(obs ~ x, data=htrsdt)
x <- seq(from=min(x), to=max(x), length.out = 100)
predhtrsdt <- coef(lmhtrsdt)[1] + x * coef(lmhtrsdt)[2]
```

Function predict

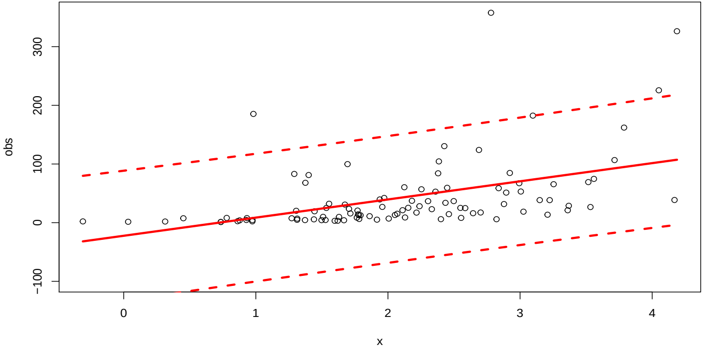
```
Xnewdata <- data.frame(x=seq(from=min(x), to=max(x),
                             length.out = 100))
Xpred <- predict(object =  lm(obs ~ x, data=htrsdt), newdata = Xnewdata,
                 se.fit = TRUE, interval = "prediction")
```

```
Xnewdata <- cbind(Xnewdata, Xpred)
```

```
head(Xnewdata)
```

Exponential data: prediction

```
plot(x, obs, ylim = c(-100, max(obs)))
lines(Xnewdata$x, Xnewdata$fit.fit, col="red", lwd=3)
lines(Xnewdata$x, Xnewdata$fit.lwr, col="red", lty=2, lwd=3)
lines(Xnewdata$x, Xnewdata$fit.upr, col="red", lty=2, lwd=3)
```



Exponential data: confidence

Prediction interval: Where the model predicts new data would be sampled, including variation unrelated to predictor

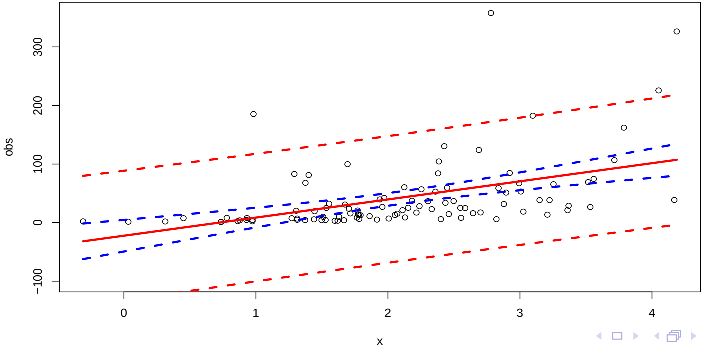
Exponential data: confidence

Prediction interval: Where the model predicts new data would be sampled, including variation unrelated to predictor
Confidence interval: Where your model predicts the MEAN new data would be sampled; or where is the true relationship with predictor

```r
Xconf <- predict(object = lm(obs ~ x), newdata = Xnewdata,
                 se.fit = TRUE, interval = "confidence")
Xnewdata[,c("conf.lwr", "conf.upr")] <-  Xconf$fit[,2:3]
```

Exponential data: confidence

```
plot(x, obs, ylim = c(-100, max(obs)))
lines(Xnewdata$x, Xnewdata$fit.fit, col="red", lwd=3)
lines(Xnewdata$x, Xnewdata$fit.lwr, col="red", lty=2, lwd=3)
lines(Xnewdata$x, Xnewdata$fit.upr, col="red", lty=2, lwd=3)
lines(Xnewdata$x, Xnewdata$conf.lwr, col="blue", lty=2, lwd=3)
lines(Xnewdata$x, Xnewdata$conf.upr, col="blue", lty=2, lwd=3)
```



Exponential data: fix

Model sufficient to show positive relationship. . . BUT

- no negative values should exist
- too many outliers
- too much uncertainty on the left. . .

Consequences:

- impossible to understand the biological mechanism
- impossible to predict future observations

What to do?

Exponential data: fix

Log-transform

```
plot(x, log(obs)); abline(lm(log(obs) ~ x))
summary(lm(log(obs) ~ x))
plot(lm(log(obs) ~ x))
```

Exponential data: fix

Log-transform

```
plot(x, log(obs)); abline(lm(log(obs) ~ x))
summary(lm(log(obs) ~ x))
plot(lm(log(obs) ~ x))
```

By the way, the simulation process:

```
set.seed(123)
x <- 2+rnorm(100)
y <- 1 + x + rnorm(100)
obs <- exp(y)
```

Exponential data: other fix

Non-parametric statistics:
Rank observations, do greater ranks go together?

Exponential data: other fix

Non-parametric statistics:
Rank observations, do greater ranks go together?
Case of two continuous variable: Spearman's rank correlation

Exponential data: other fix

Non-parametric statistics:
Rank observations, do greater ranks go together?
Case of two continuous variable: Spearman's rank correlation

```
cor.test(x = x, y = obs, method = "spearman")


Spearman's rank correlation rho

data:  x and obs
S = 60622, p-value < 2.2e-16
alternative hypothesis: true rho is not equal to 0
sample estimates:
      rho
0.6362316
```

Exponential data: other fix

Non-parametric statistics:
Rank observations, do greater ranks go together?
Case of two continuous variable: Spearman's rank correlation

```
cor.test(x = x, y = obs, method = "spearman")


Spearman's rank correlation rho

data:  x and obs
S = 60622, p-value < 2.2e-16
alternative hypothesis: true rho is not equal to 0
sample estimates:
      rho
0.6362316
```

Significant positive correlation confirmed, BUT, no biological mechanism, little predictive power.

Linear model basic assumptions

Not necessarily wrong, but typical interpretation assumes:

- Linear combination of parameters (including transformation, polynoms, interactions. . .)
- Predictor not perfectly correlated
- Measurement error in predictors
- Gaussian error distribution
- Homoscedasticity (constant error variance)
- Independence of error

Linear model basic assumptions

Not necessarily wrong, but typical interpretation assumes:

- Linear combination of parameters (including transformation, polynoms, interactions. . .)
- Predictor not perfectly correlated
- Measurement error in predictors
- Gaussian error distribution
- Homoscedasticity (constant error variance)
- Independence of error

TEASER: The most difficult problems are the reason for

- Multiple regression

Linear model basic assumptions

Not necessarily wrong, but typical interpretation assumes:

- Linear combination of parameters (including transformation, polynoms, interactions. . .)
- Predictor not perfectly correlated
- Measurement error in predictors
- Gaussian error distribution
- Homoscedasticity (constant error variance)
- Independence of error

TEASER: The most difficult problems are the reason for

- Multiple regression
- Generalized Linear Models

Linear model basic assumptions

Not necessarily wrong, but typical interpretation assumes:

- Linear combination of parameters (including transformation, polynoms, interactions. . .)
- Predictor not perfectly correlated
- Measurement error in predictors
- Gaussian error distribution
- Homoscedasticity (constant error variance)
- Independence of error

TEASER: The most difficult problems are the reason for

- Multiple regression
- Generalized Linear Models
- (Generalized) Linear Mixed Models

Linear model basic assumptions

Not necessarily wrong, but typical interpretation assumes:

- Linear combination of parameters (including transformation, polynoms, interactions. . .)
- Predictor not perfectly correlated
- Measurement error in predictors
- Gaussian error distribution
- Homoscedasticity (constant error variance)
- Independence of error

TEASER: The most difficult problems are the reason for

- Multiple regression
- Generalized Linear Models
- (Generalized) Linear Mixed Models
- Non-linear models