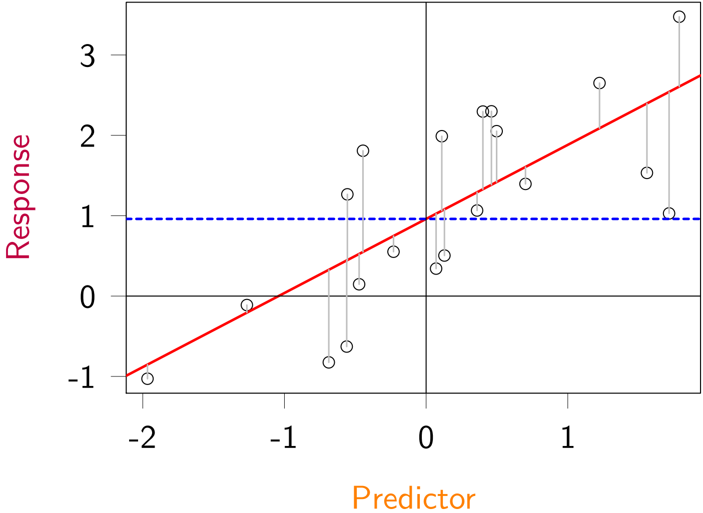
What on earth is going on with my linear models??!

March 8, 2018

1. Linear model, reminder
2. Checks and prediction
3. Cures
4. Bonus fun

A simple linear model

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



A simple linear model

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

In R:

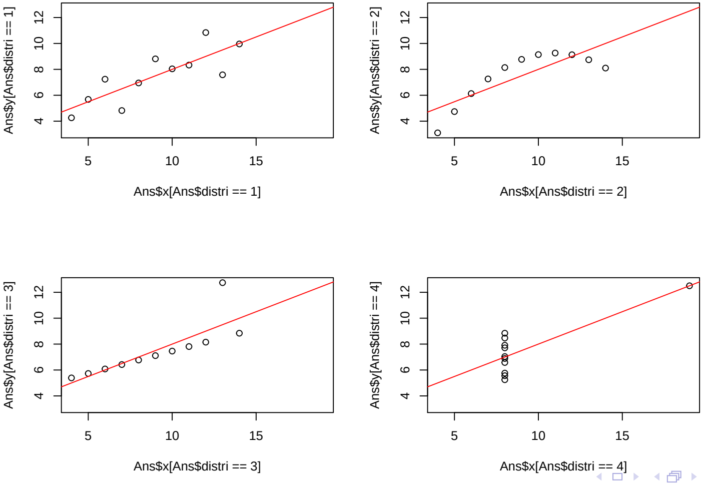
```
lm(response ~ 1 + predictor1 + predictor2, data=data)
  # equivalent to
lm(response ~ predictor1 + predictor2, data=data)
```

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

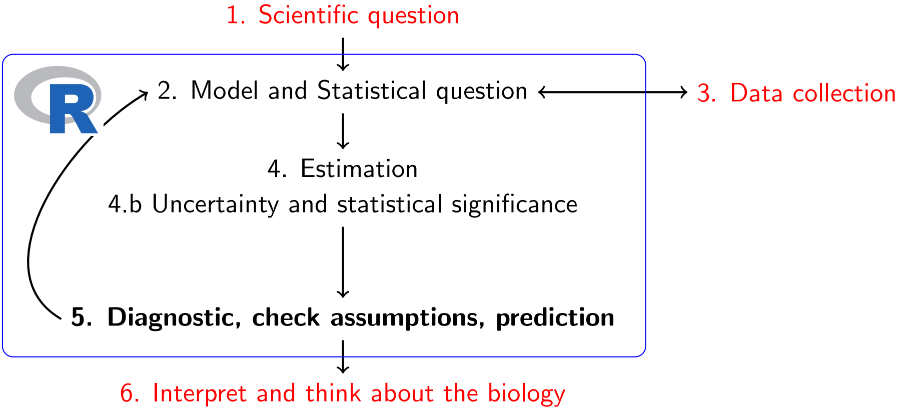
1. Linear model, reminder
2. Checks and prediction
3. Cures
4. Bonus fun

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

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”?

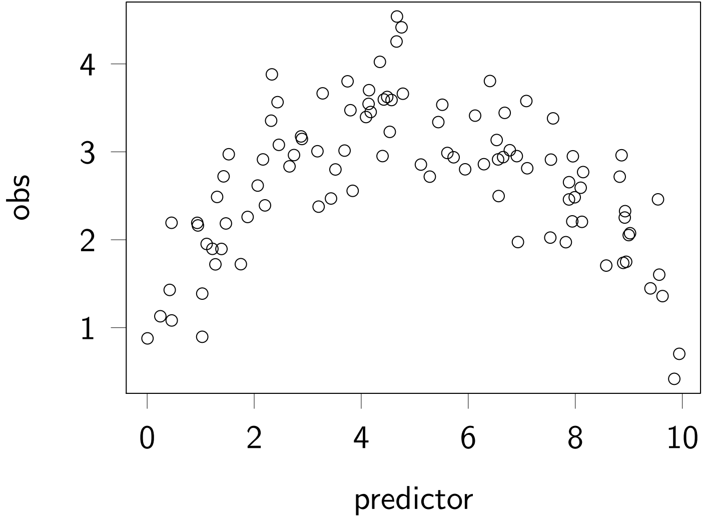
```r
  summary(lm(obs ~ 1 + predictor, data=forprediction) )
```

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
```

Why we need checks: missing a relationship



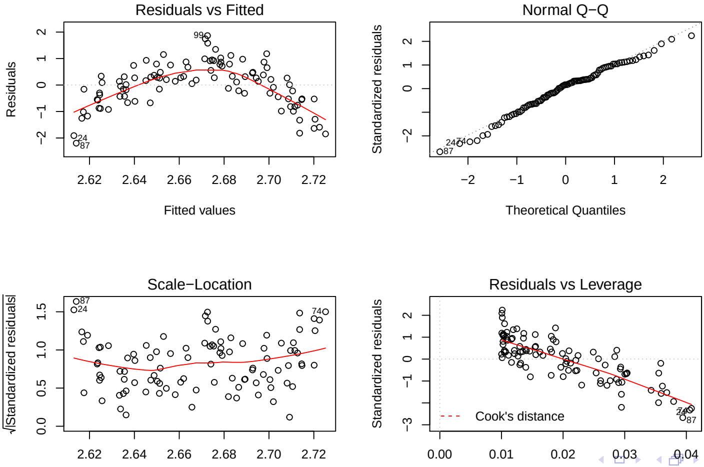
How to check?

```
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
```

How to check?

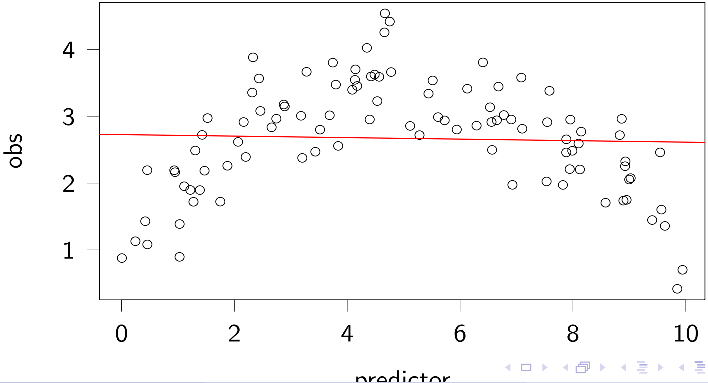
```
par(mfrow=c(2,2))
  plot(m0)
```



How to check?

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

```
    setPar()
plot(x=forprediction$predictor, y=forprediction$obs, xlab="predictor
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?

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

Fix?

```
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)

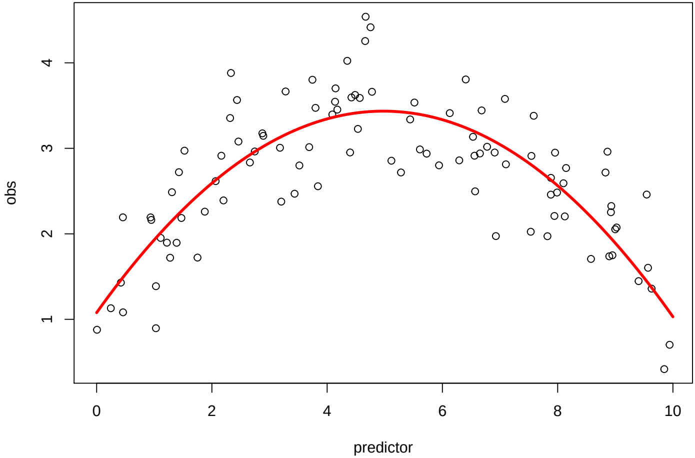
Introduction to prediction

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

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

Small exercise

Use regression coefficients to predict ”obs” from ”predictor”, and add a prediction line on the plot obs/predictor.
Is the fit satisfactory?

Introduction to prediction



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?

```
Call:
lm(formula = y ~ x1 + x2, data = cdata)

Residuals:
    Min      1Q  Median      3Q     Max
-0.9679 -0.4763 -0.1581  0.4476  1.5434

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   -22.45     127.66  -0.176    0.862
x1             23.43     127.74   0.183    0.856
x2             22.36     127.62   0.175    0.862

Residual standard error: 0.6513 on 32 degrees of freedom
```

Practice lm() with parasites

What explains variation in parasitic load?

You collected ecto-parasites on some furry large mammals at three locations. Parasites break easily when we collect them and are impossible to count, so we decide to measure parasitic load as their mass. **Why do some mammals have larger parasitic load?**

Practice lm() with parasites

What explains variation in parasitic load?

You collected ecto-parasites on some furry large mammals at three locations. Parasites break easily when we collect them and are impossible to count, so we decide to measure parasitic load as their mass. **Why do some mammals have larger parasitic load?**

- Load the `Para.csv` data (don't forget: str(), summary(), plot()...)
- Model `Parasite_Mass` using `lm()`
- Find what variables predict `Parasite_Mass`
- How good are your models? Assumptions? Prediction?
- What biological interpretation can you imagine?

1. Linear model, reminder
2. Checks and prediction
3. Cures
4. Bonus fun

Multiple regression

Transformations

Linear model basic assumptions

Not necessarily wrong, but typical interpretation assumes:

- Linear combination of parameters (including transformation, polynoms, interactions. . .)
  Risk: biologically meaningless

Linear model basic assumptions

Not necessarily wrong, but typical interpretation assumes:

- Linear combination of parameters (including transformation, polynoms, interactions. . .)
  Risk: biologically meaningless
- 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
- 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
- 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
- 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
- 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

1 Linear model, reminder

2 Checks and prediction

3 Cures

4 Bonus fun

Extra exercises

General R coding

1. What is the fastest way to get row averages in a data-frame?
2. Create a function called colVars, like colMeans but for variance
3. Create nice plots to visualize iris data (ideally journal-quality)

Linear models

1. 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?
2. For model that can be fitted with t.test, aov, and lm, is one of the function faster?
3. Write your own code to obtain a prediction from a lm (that is, a simpler version of the predict function), with confidence interval. (extra toughness: do it using the matrix formulation of the analytical solution to a linear model)

What do you want to learn about?

Topics

-