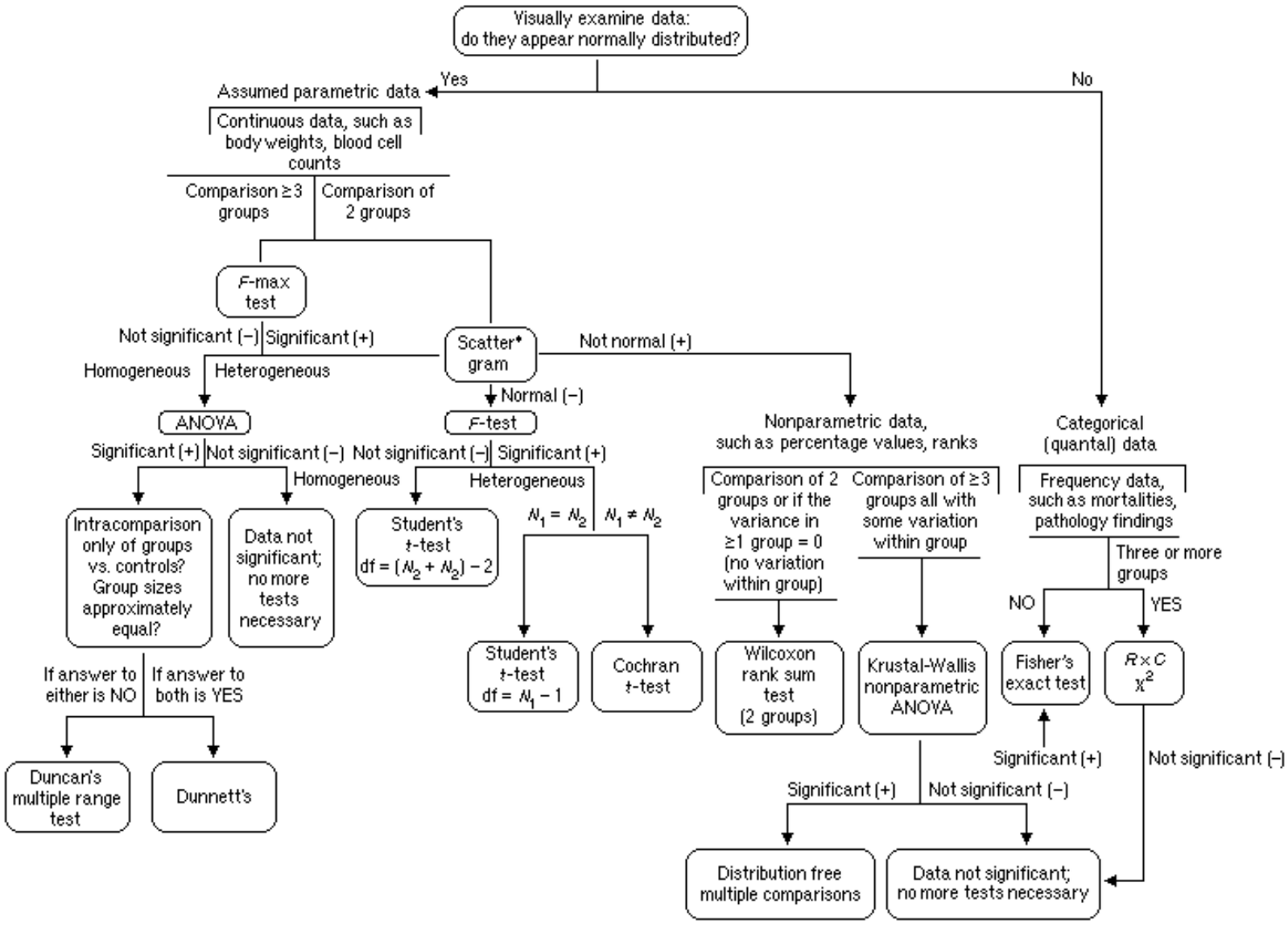
Linear, Generalized, and Mixed/Multilevel models - an introduction with R

Francisco Rodriguez-Sanchez

http://bit.ly/frod_san

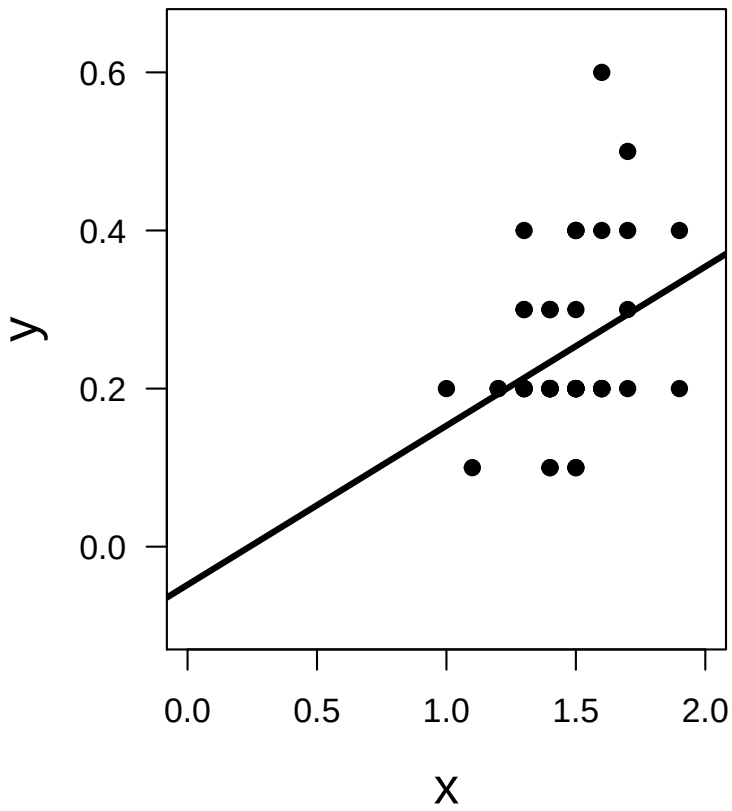
Introduction to linear models

Modern statistics are easier than this



Our overarching regression framework

$$y_i = a + bx_i + \varepsilon_i$$
$$\varepsilon_i \sim N\left(0, \sigma^2\right)$$



Data

y = response variable

x = predictor

Parameters

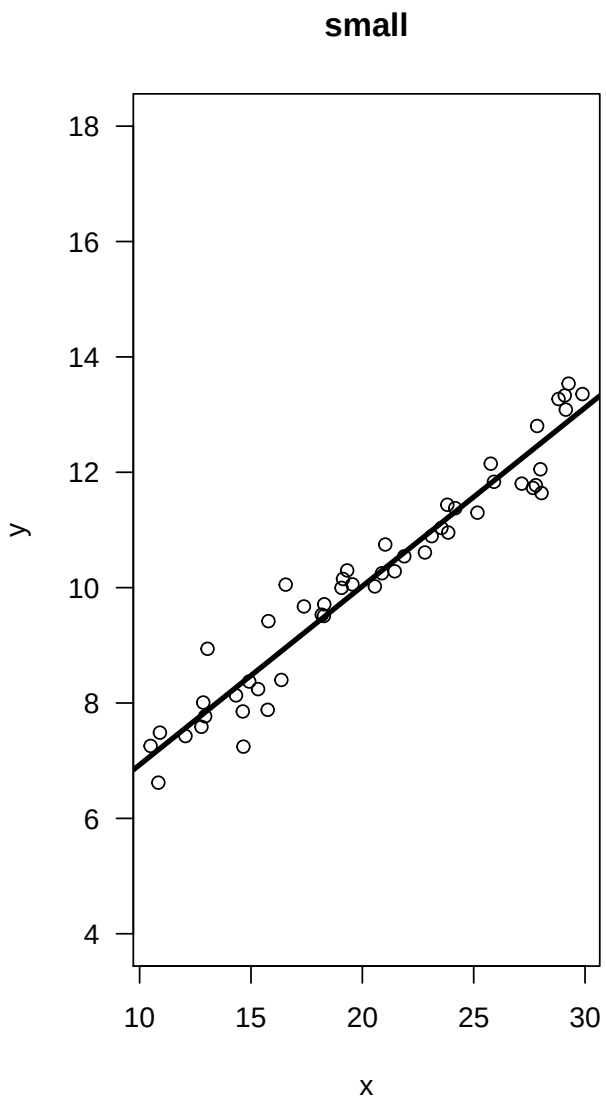
a = intercept

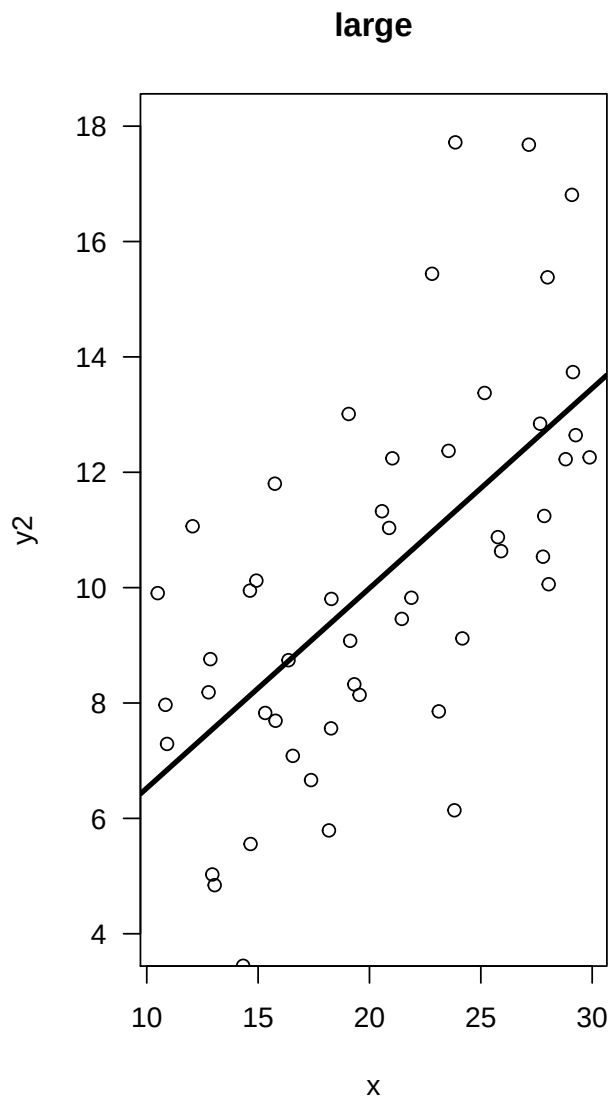
b = slope

σ = residual variation

ε = residuals

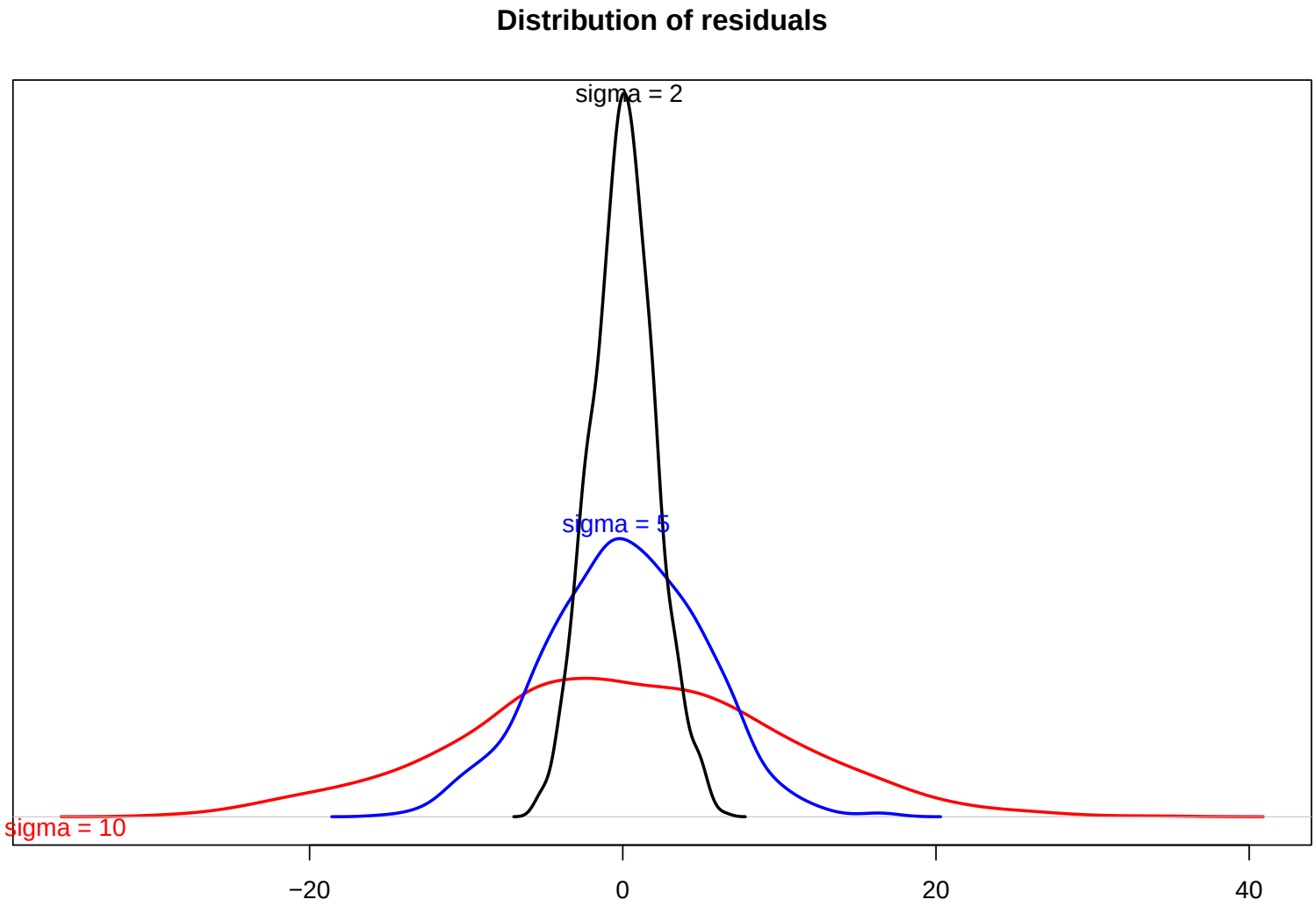
Residual variation (error)



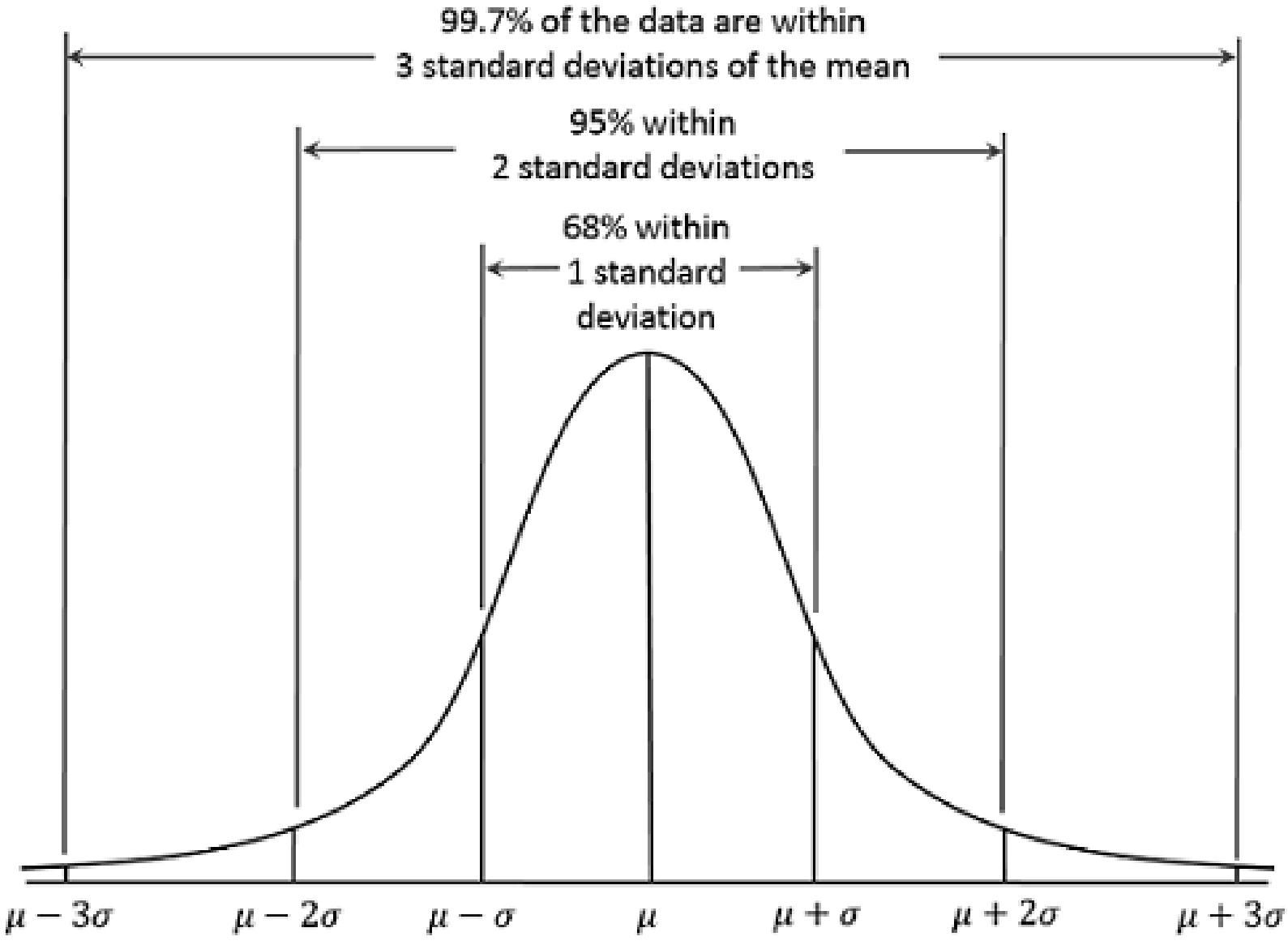


Residual variation

$$\varepsilon_i \sim N\left(0, \sigma^2\right)$$



In a Normal distribution



Different ways to write same model

$$y_i = a + bx_i + \varepsilon_i$$
$$\varepsilon_i \sim N\left(0, \sigma^2\right)$$

.

$$y_i \sim N\left(\mu_i, \sigma^2\right)$$
$$\mu_i = a + bx_i$$
$$\varepsilon_i \sim N\left(0, \sigma^2\right)$$

Linear models

Example dataset: paper planes flying experiment

```
library(paperplanes)
head(paperplanes)
```

```
# A tibble: 6 x 8
     id hour    person   gender   age plane       paper distance
  <int> <fct>   <chr>    <fct>  <dbl> <chr>       <int>    <dbl>
1     1 [17,18) Roland   male      30 Standard80     80      7.8
2     2 [17,18) Astrid   female    30 Concorde120   120      2.7
3     3 [17,18) Roland   male      30 Standard120   120      9.2
4     4 [17,18) Isabella female    48 Standard120   120      6
5     5 [17,18) Fabienne female    17 Standard120   120      7.3
6     6 [17,18) Fabienne female    17 Standard120   120      7.8
```

Questions

- What is the relationship between age and distance flown?

Questions

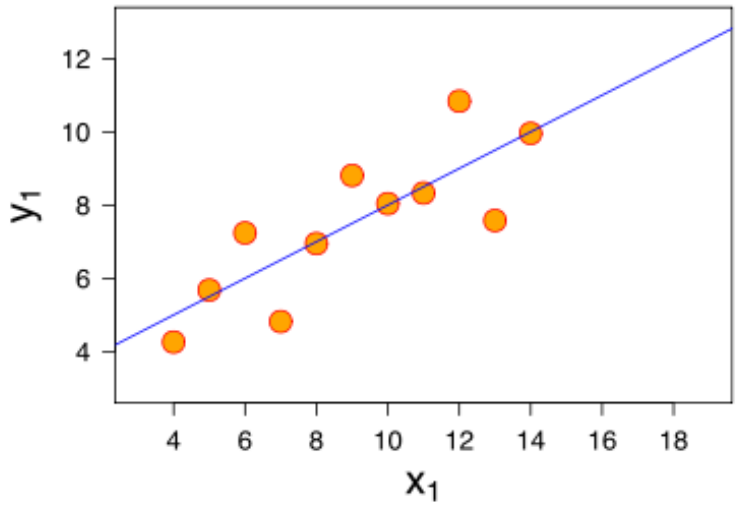
- What is the relationship between age and distance flown?
- Do adults achieve longer distances?

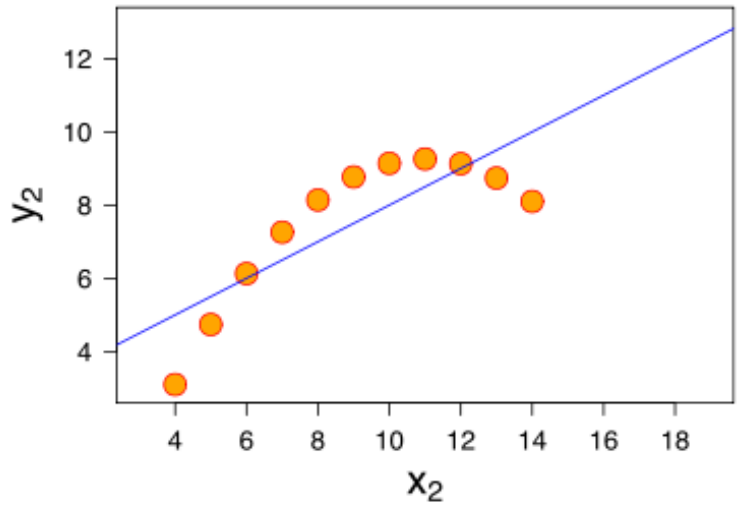
Questions

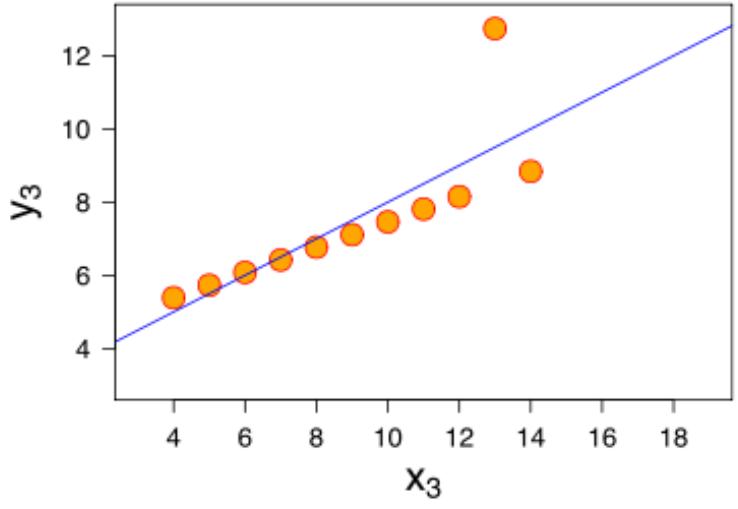
- What is the relationship between age and distance flown?
- Do adults achieve longer distances?
- Can we predict distance flown from participant's age? How well?

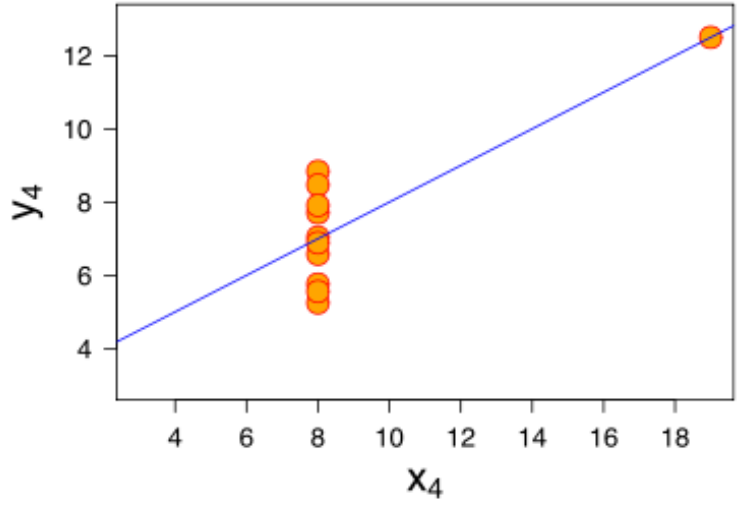
Always plot your data first!

Always plot your data first!





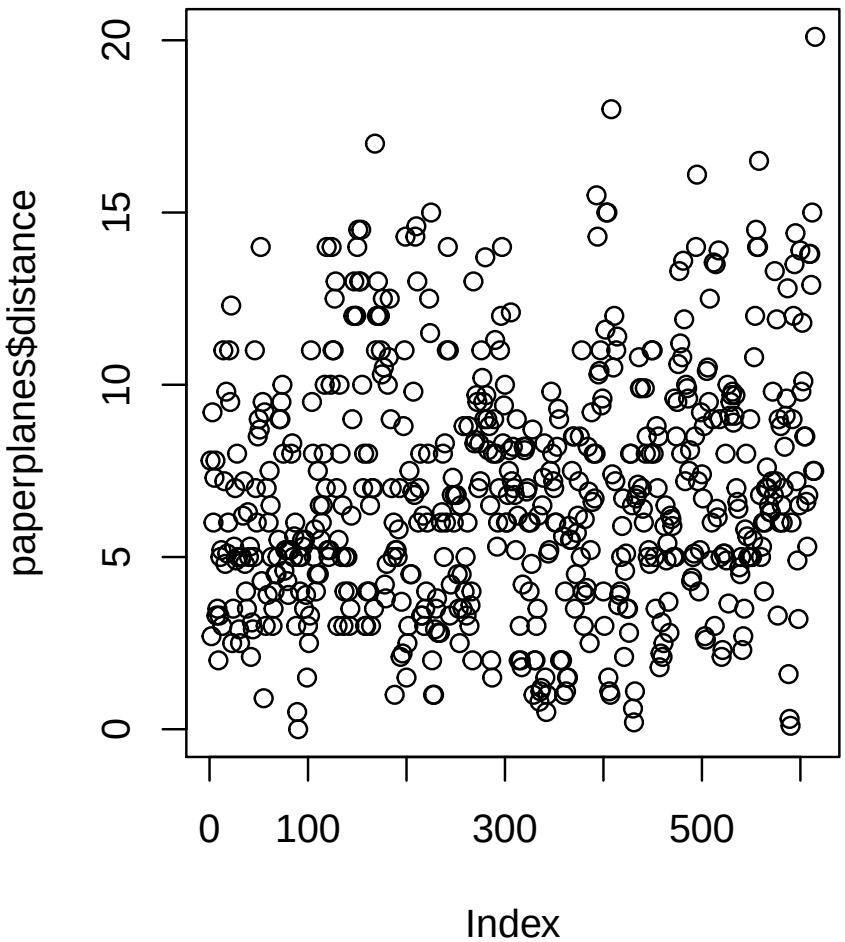




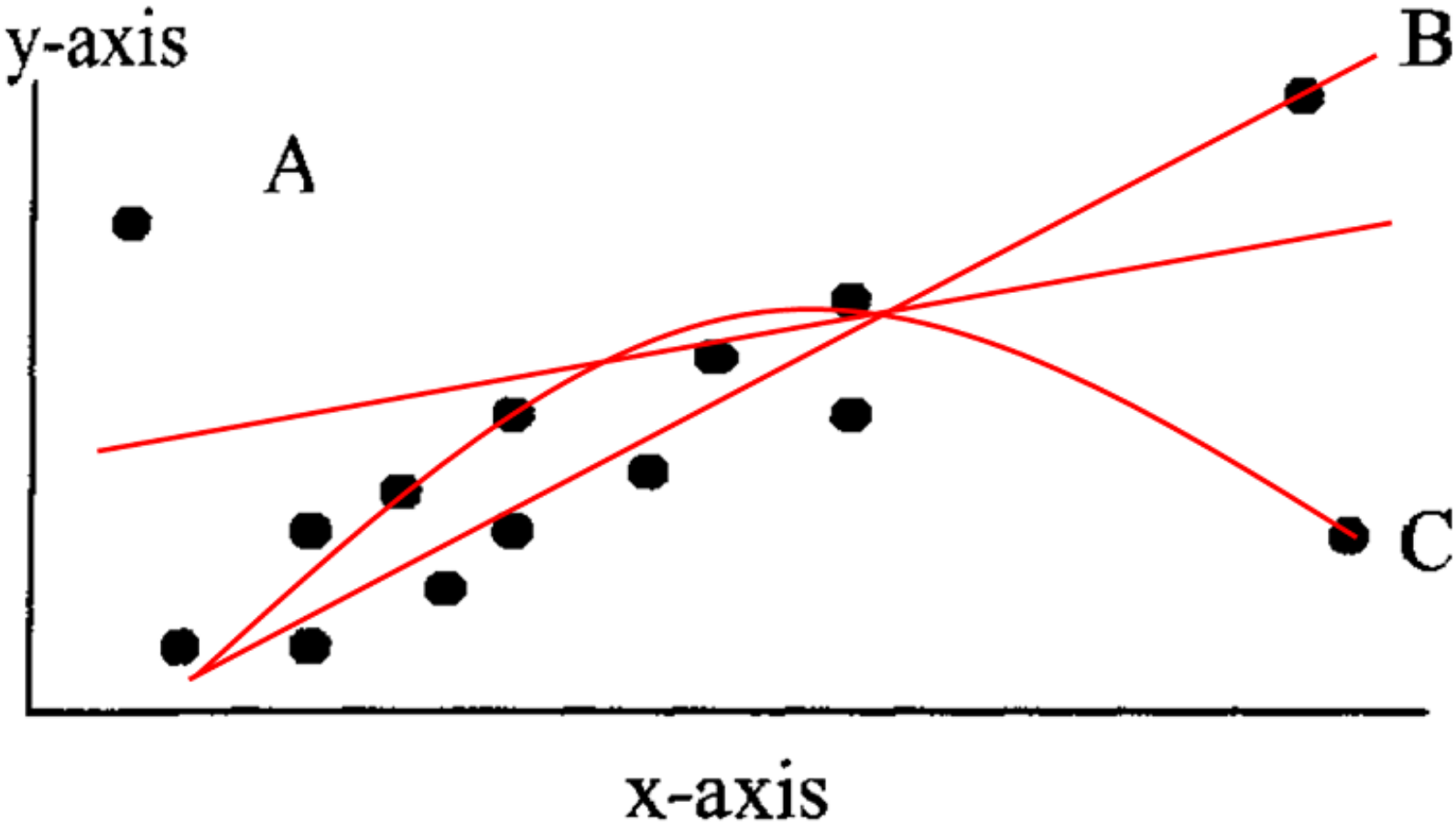
Exploratory Data Analysis (EDA)

Outliers

```
plot(paperplanes$distance)
```



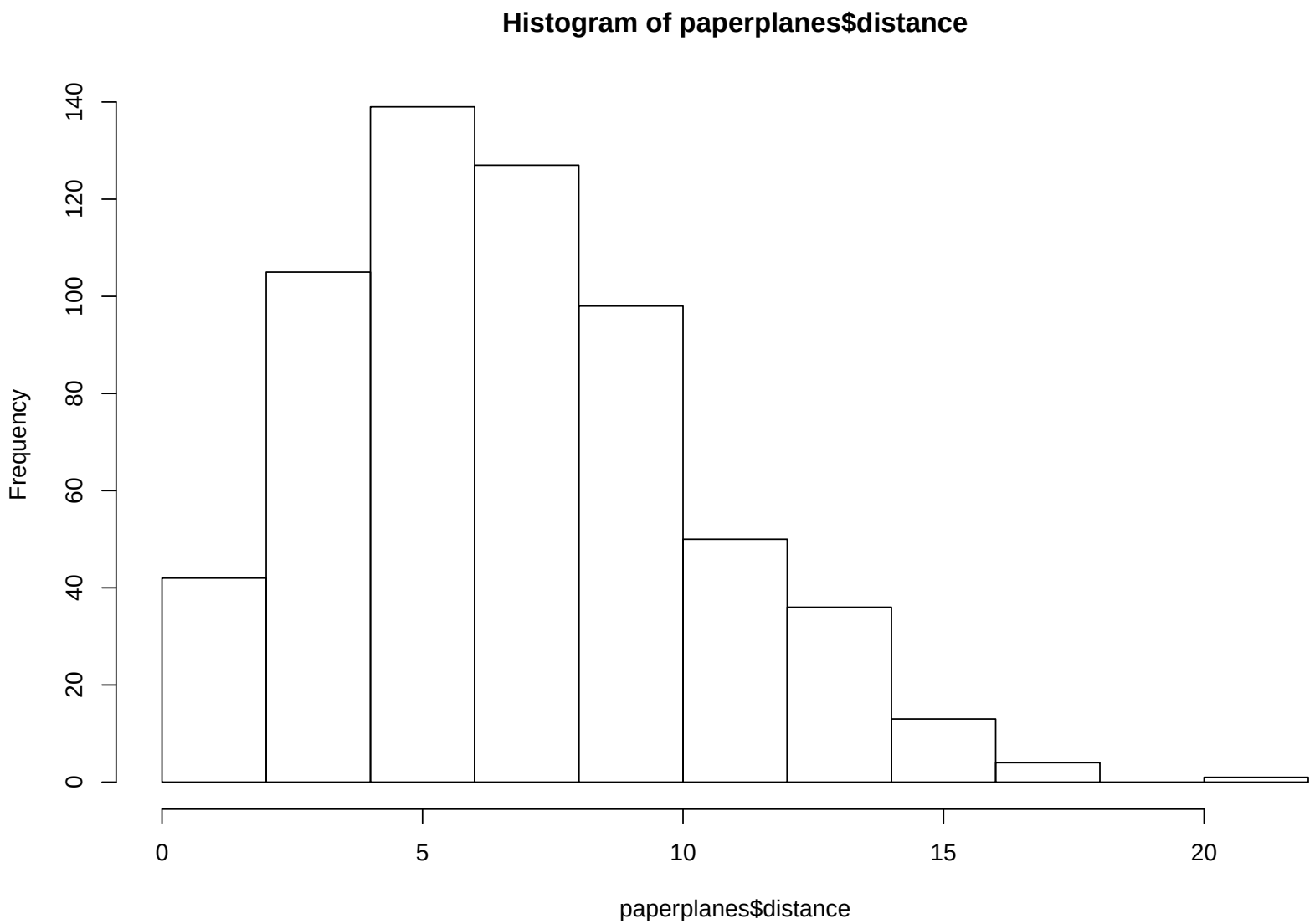
Outliers impact on regression



See http://rpsychologist.com/d3/correlation/

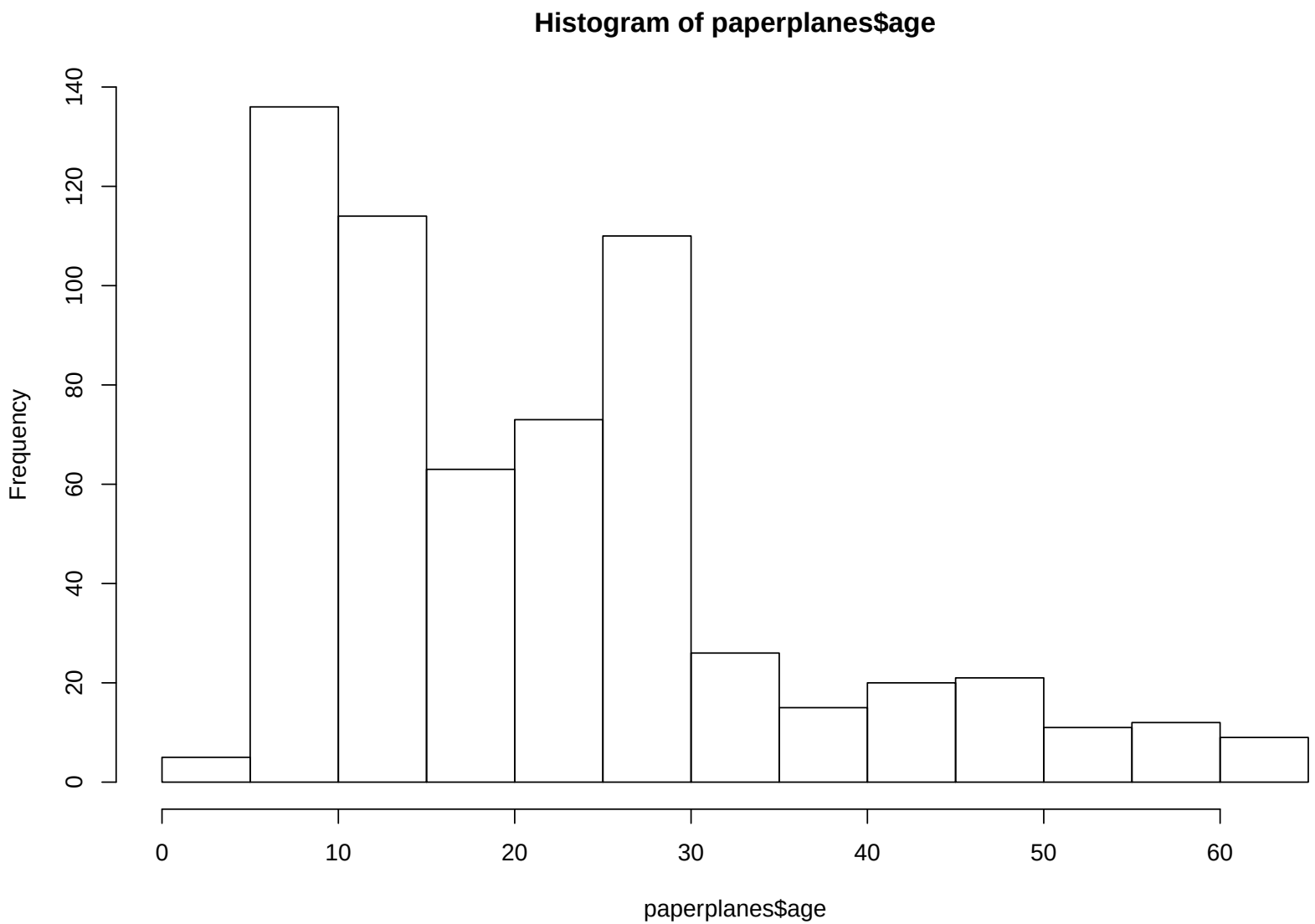
Histogram of response variable

```
hist(paperplanes$distance)
```



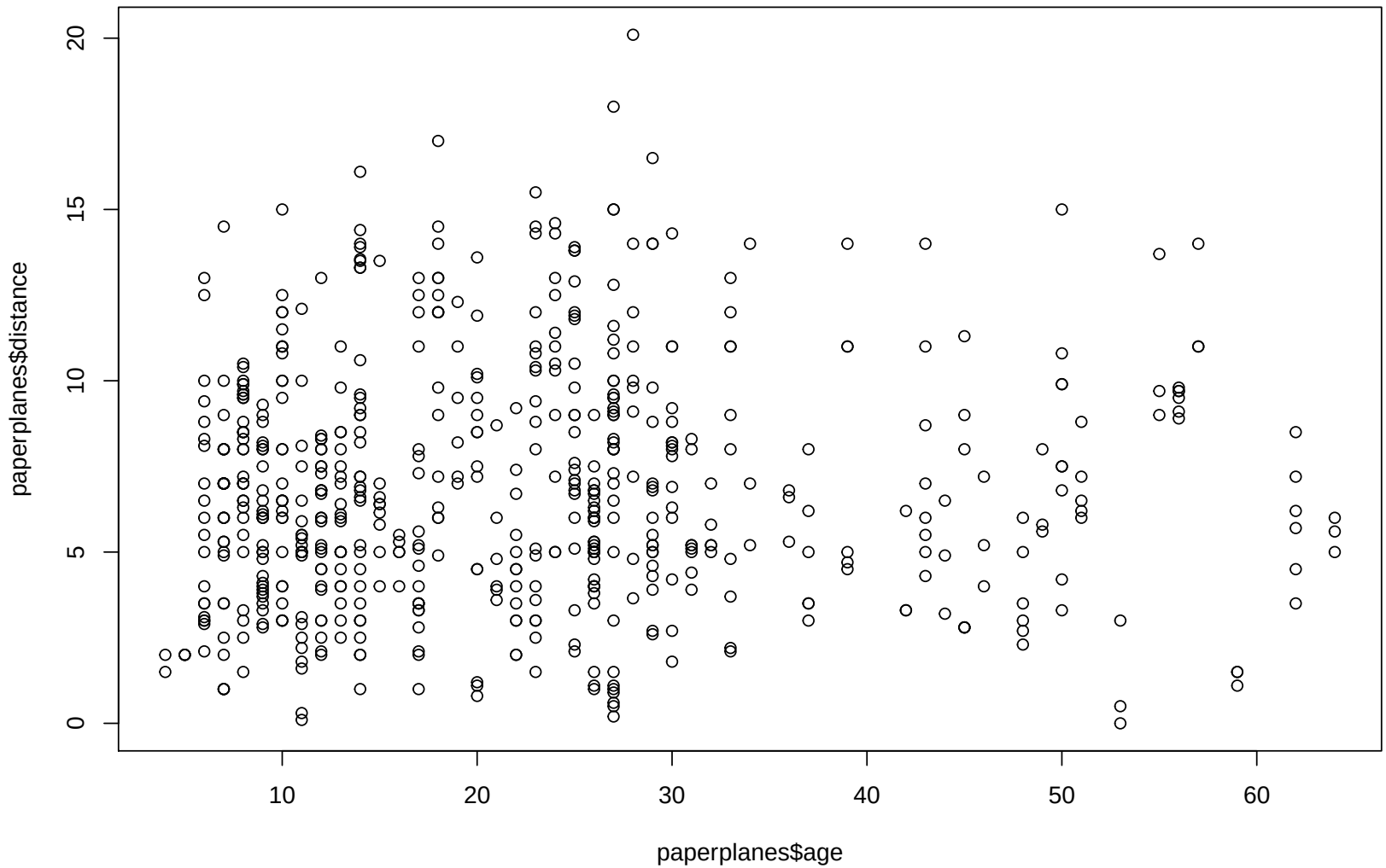
Histogram of predictor variable

```
hist(paperplanes$age)
```



Scatterplot

```
plot(paperplanes$age, paperplanes$distance)
```



Model fitting

Now fit model

Hint: `lm`

Now fit model

```
m1 <- lm(distance ~ age, data = paperplanes)
```

which corresponds to

$$Distance_i = a + b \cdot age_i + \varepsilon_i$$
$$\varepsilon_i \sim N\left(0, \sigma^2\right)$$

Model interpretation

What does this mean?

```
Call:
lm(formula = distance ~ age, data = paperplanes)

Residuals:
    Min      1Q  Median      3Q     Max
-7.1929 -2.6014 -0.3789  2.1572 13.1658

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  6.64440    0.26982  24.626   <2e-16 ***
age          0.01035    0.01040   0.996     0.32
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.504 on 613 degrees of freedom
Multiple R-squared:  0.001614,  Adjusted R-squared:  -1.434e-05
F-statistic: 0.9912 on 1 and 613 DF,  p-value: 0.3198
```

Presenting model results

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	6.64	0.27	24.63	0.00
age	0.01	0.01	1.00	0.32

Presenting model results

	Model 1
(Intercept)	6.64 (0.27)***
age	0.01 (0.01)
R^2	0.00
Adj. R^2	-0.00
Num. obs.	615
RMSE	3.50

$^{***}p < 0.001$, $^{**}p < 0.01$, $^{*}p < 0.05$

Table 2: Statistical models

Retrieving model coefficients

```
coef(m1)
```

```
(Intercept)         age
 6.64439782  0.01034968
```

Tidy up model coefficients with broom

```
library(broom)
tidy(m1)
```

```
# A tibble: 2 x 5
  term        estimate std.error statistic  p.value
  <chr>          <dbl>     <dbl>     <dbl>    <dbl>
1 (Intercept)   6.64      0.270     24.6   1.29e-93
2 age           0.0103    0.0104     0.996 3.20e- 1
```

```
glance(m1)
```

```
# A tibble: 1 x 11
  r.squared adj.r.squared sigma statistic p.value    df logLik
      <dbl>         <dbl> <dbl>     <dbl>   <dbl> <int>  <dbl> <
1   0.00161    -0.0000143  3.50     0.991   0.320     2 -1643. 3
# ... with 2 more variables: deviance <dbl>, df.residual <int>
```

Confidence intervals

```
confint(m1)
```

```
                   2.5 %     97.5 %
(Intercept)  6.11452177 7.17427388
age         -0.01006553 0.03076489
```

Using effects package

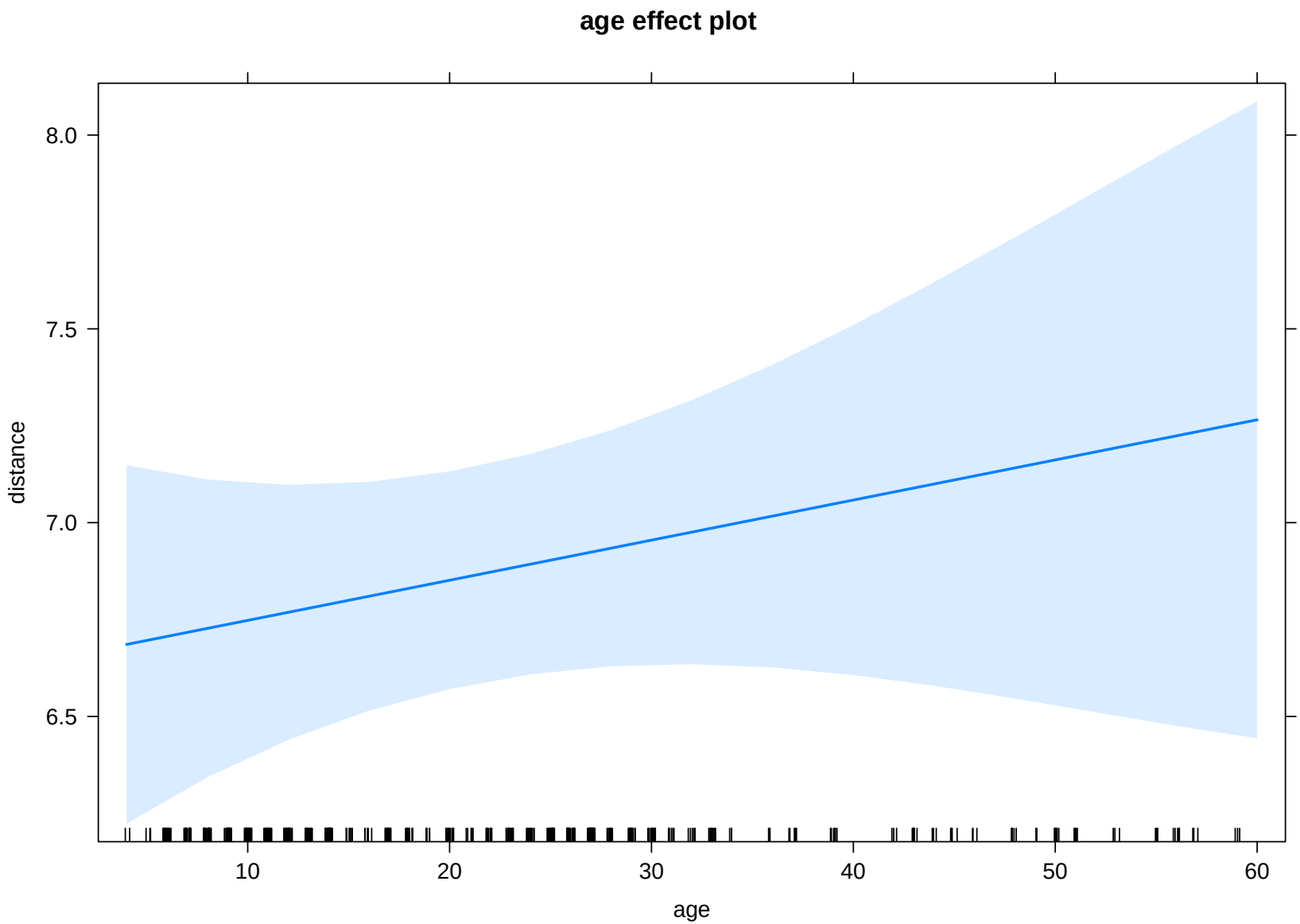
```
library(effects)
summary(allEffects(m1))
```

```
 model: distance ~ age

 age effect
age
       4        20        30        50        60
6.685797 6.851391 6.954888 7.161882 7.265379

 Lower 95 Percent Confidence Limits
age
       4        20        30        50        60
6.223509 6.570601 6.634085 6.528536 6.443633

 Upper 95 Percent Confidence Limits
age
       4        20        30        50        60
7.148084 7.132182 7.275692 7.795228 8.087125
```

Visualising fitted model

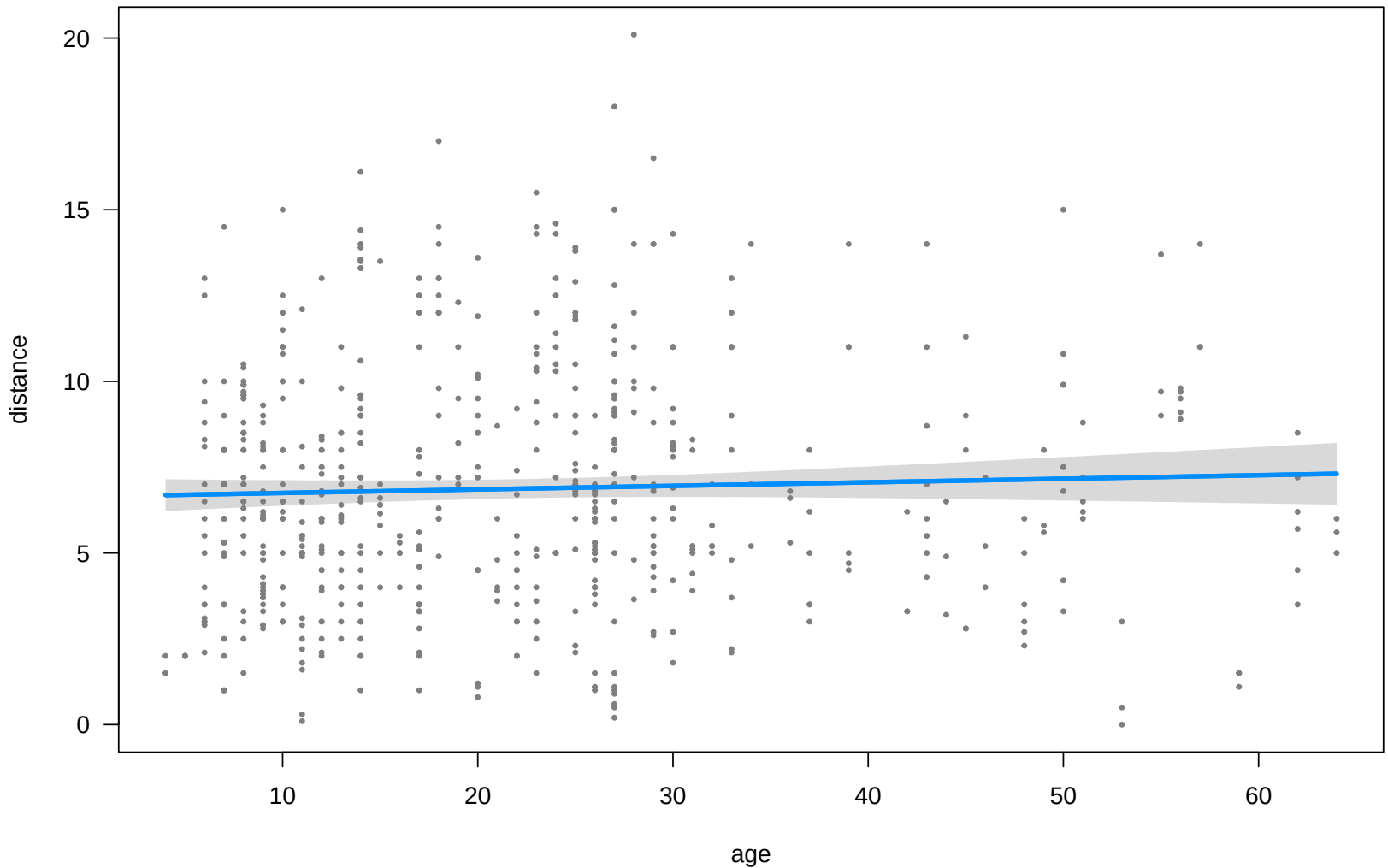
Plot effects

```
plot(allEffects(m1))
```



Plot model (visreg)

```
library(visreg)
visreg(m1)
```



Model checking

Linear model assumptions

- Linearity (transformations, GAM...)

Linear model assumptions

- Linearity (transformations, GAM. . .)
- Residuals:

Linear model assumptions

- Linearity (transformations, GAM...)
- Residuals:
  - Independent

Linear model assumptions

- Linearity (transformations, GAM...)
- Residuals:
  - Independent
  - Equal variance

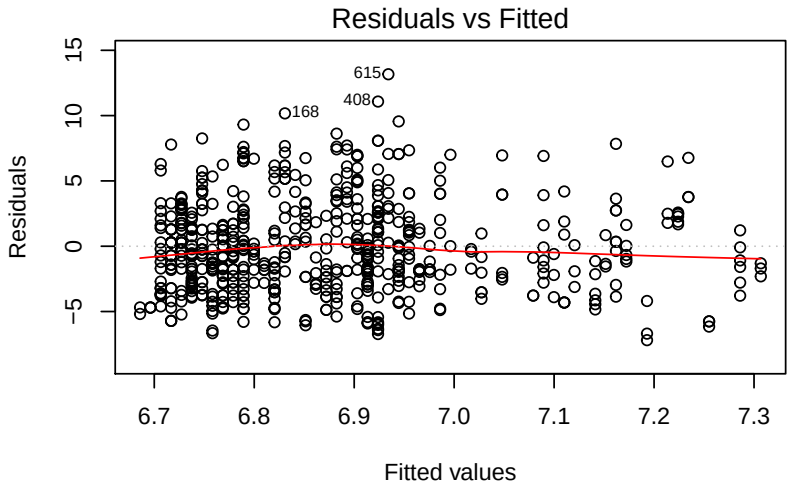
Linear model assumptions

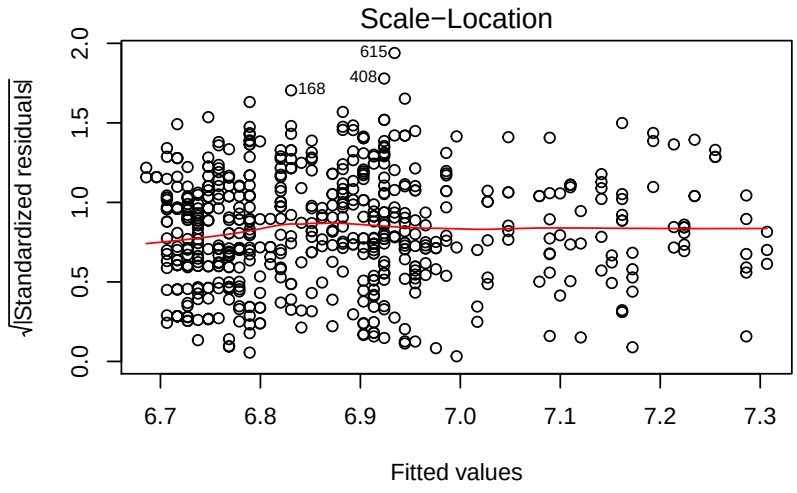
- Linearity (transformations, GAM...)
- Residuals:
  - Independent
  - Equal variance
  - Normal

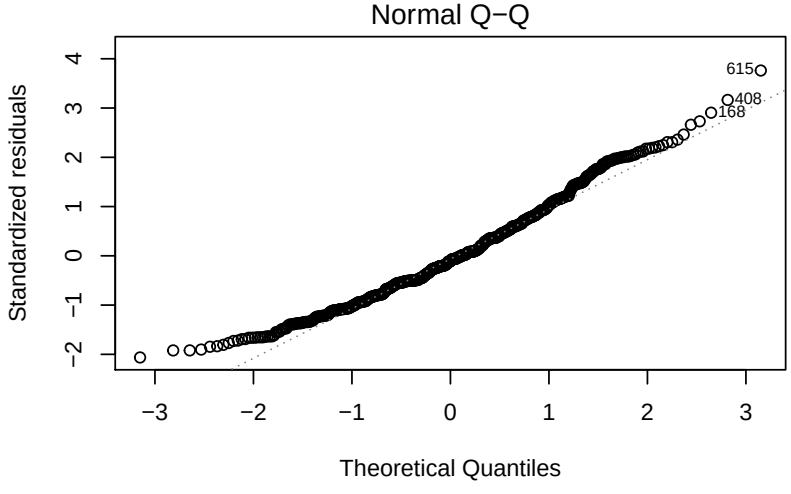
Linear model assumptions

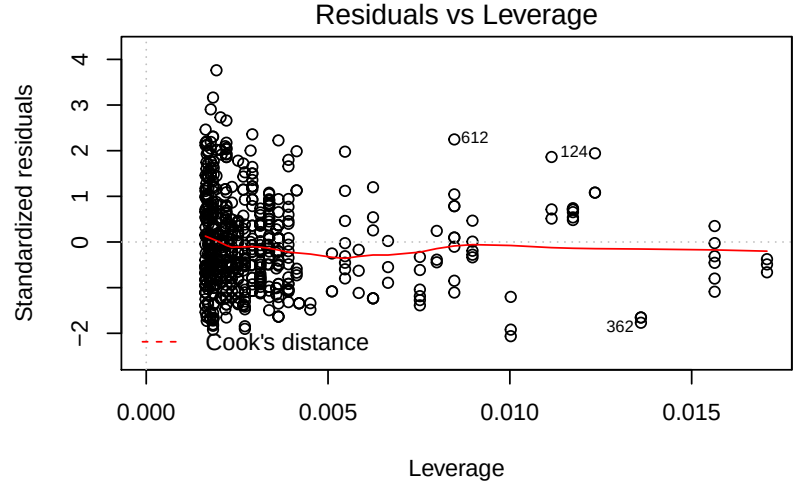
- Linearity (transformations, GAM...)
- Residuals:
  - Independent
  - Equal variance
  - Normal
- No measurement error in predictors

Model checking: residuals



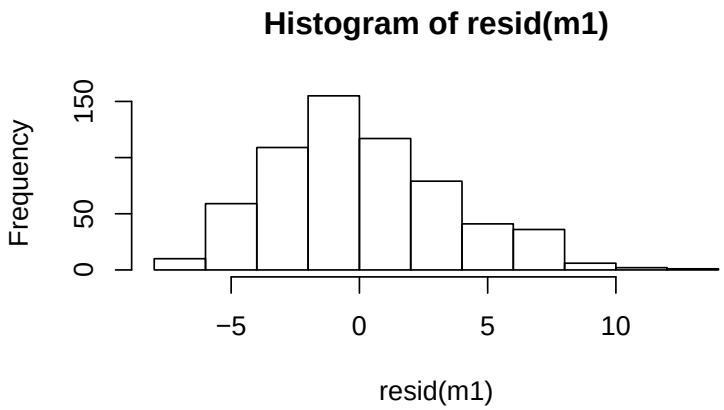






Are residuals normal?

```
hist(resid(m1))
```




```
lm(formula = distance ~ age, data = paperplane
            coef.est coef.se
(Intercept) 6.64     0.27
age         0.01     0.01
---
n = 615, k = 2
residual sd = 3.50, R-Squared = 0.00
```

SD of residuals = 3.5 coincides with estimate of `sigma`.

Using model for prediction

How good is the model in predicting distance?

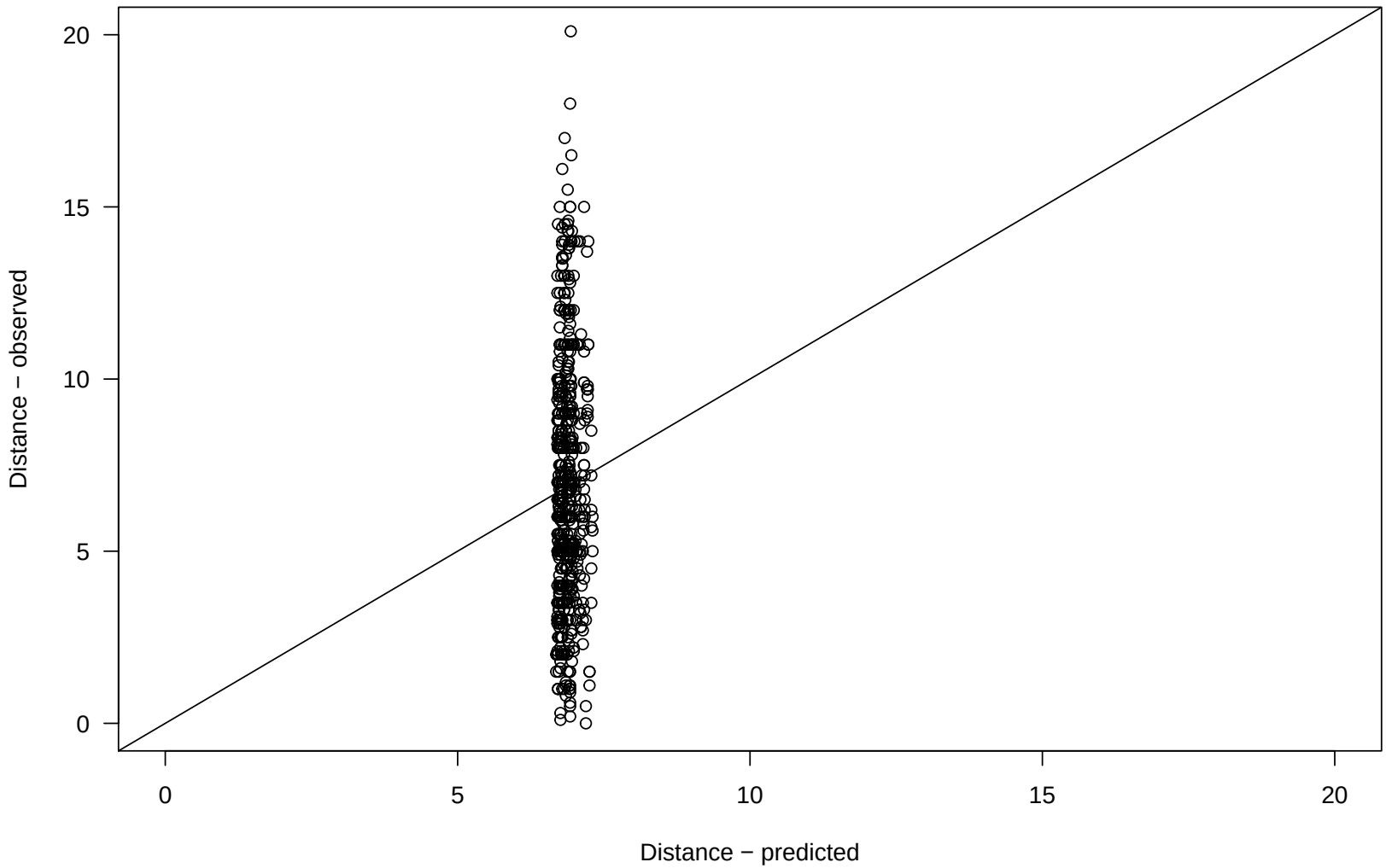
`fitted` gives predictions for each observation

```
paperplanes$distance.pred <- fitted(m1)
head(paperplanes)
```

```
# A tibble: 6 x 9
     id hour    person  gender   age plane     paper distance distance.pred
  <int> <fct>   <chr>   <fct>  <dbl> <chr>     <int>    <dbl>          <dbl>
1     1 [17,18) Roland  male      30 Standard~    80      7.8           6.95
2     2 [17,18) Astrid  female    30 Concorde~   120      2.7           6.95
3     3 [17,18) Roland  male      30 Standard~   120      9.2           6.95
4     4 [17,18) Isabel~ female    48 Standard~   120      6             7.14
5     5 [17,18) Fabien~ female    17 Standard~   120      7.3           6.82
6     6 [17,18) Fabien~ female    17 Standard~   120      7.8           6.82
```

Calibration plot: Observed vs Predicted values

```
plot(paperplanes$distance.pred, paperplanes$distance, xlab = "Di
```



Using fitted model for prediction

Q: Expected distance if age = 30?

```
new.age <- data.frame(age = c(30))
predict(m1, new.age, se.fit = TRUE)
```

```
$fit
       1
6.954888

$se.fit
[1] 0.1633552

$df
[1] 613

$residual.scale
[1] 3.503736
```

Using fitted model for prediction

Q: Expected distance if age = 30?

```
new.age <- data.frame(age = c(30))
predict(m1, new.age, se.fit = TRUE, interval = "confidence", lev
```

```
$fit
       fit      lwr      upr
1 6.954888 6.634085 7.275692

$se.fit
[1] 0.1633552

$df
[1] 613

$residual.scale
[1] 3.503736
```

Using fitted model for prediction

Q: Expected distance if age = 30?

```
new.age <- data.frame(age = c(30))
predict(m1, new.age, se.fit = TRUE, interval = "prediction", lev
```

```
$fit
       fit         lwr      upr
1 6.954888 0.06663211 13.84314

$se.fit
[1] 0.1633552

$df
[1] 613

$residual.scale
[1] 3.503736
```

Important functions

- `plot`

Important functions

- `plot`
- `summary`

Important functions

- `plot`
- `summary`
- `coef`

Important functions

- `plot`
- `summary`
- `coef`
- `confint`

Important functions

- `plot`
- `summary`
- `coef`
- `confint`
- `fitted`

Important functions

- `plot`
- `summary`
- `coef`
- `confint`
- `fitted`
- `resid`

Important functions

- `plot`
- `summary`
- `coef`
- `confint`
- `fitted`
- `resid`
- `allEffects`

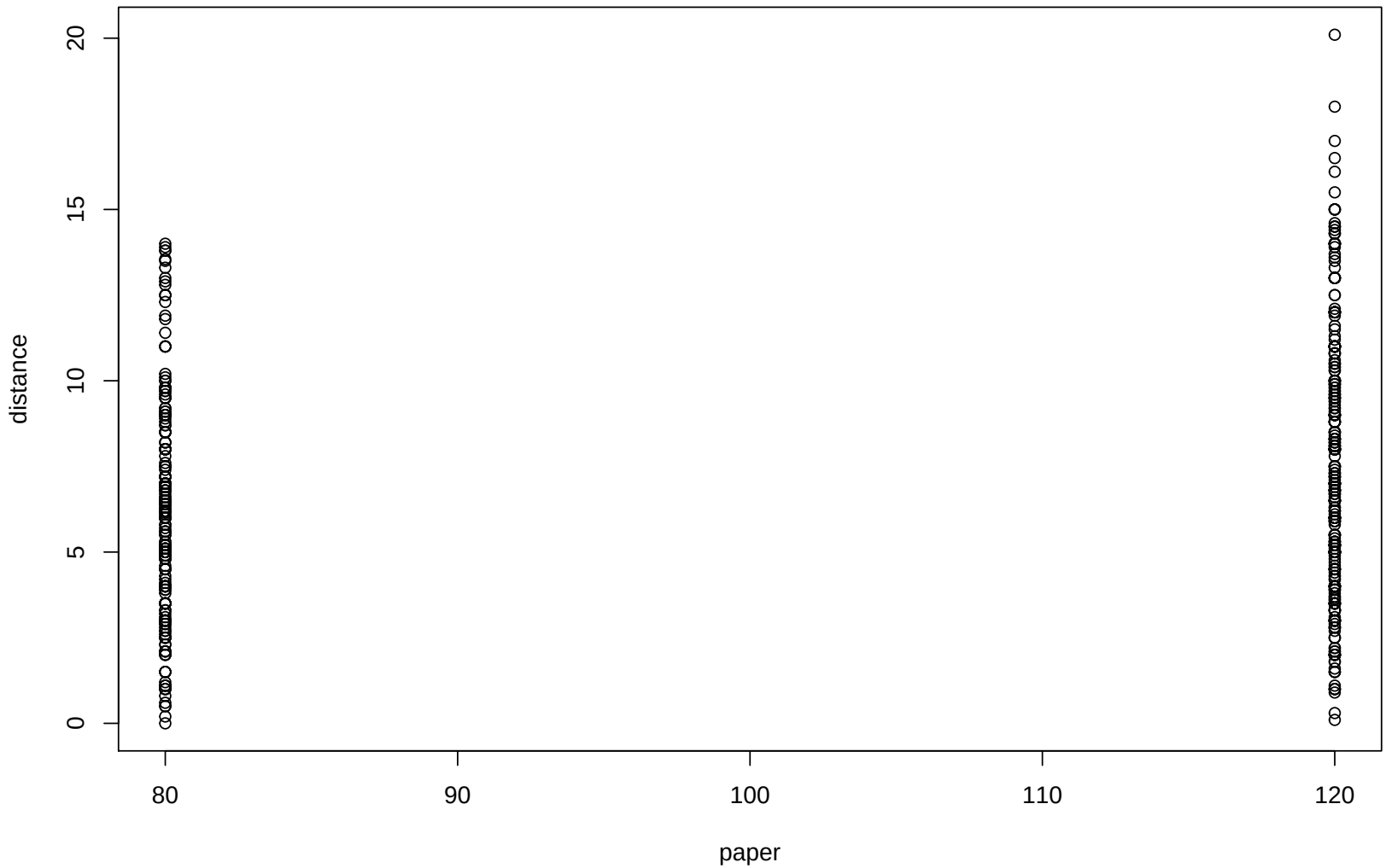
Important functions

- `plot`
- `summary`
- `coef`
- `confint`
- `fitted`
- `resid`
- `allEffects`
- `predict`

Categorical predictors (factors)

Q: Does distance vary with paper type?

```
plot(distance ~ paper, data = paperplanes)
```



Model distance ~ paper

All right here?

```
m2 <- lm(distance ~ paper, data = paperplanes)
```

```
Call:
lm(formula = distance ~ paper, data = paperplanes)

Residuals:
    Min      1Q  Median      3Q     Max
-7.2756 -2.3756 -0.3756  2.2244 12.7244

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 3.638290   0.750041   4.851 1.56e-06 ***
paper       0.031144   0.007095   4.389 1.34e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.453 on 613 degrees of freedom
Multiple R-squared:  0.03047,   Adjusted R-squared:  0.02889
```

Model distance ~ paper

Paper is a factor!

```
paperplanes$paper <- as.factor(paperplanes$paper)
```

```
       id              hour        person             gender
 Min.   :  1.0   [19,20) :139   Length:615         female:213
 1st Qu.:154.5   [22,23) :108   Class :character   male  :402
 Median :308.0   [21,22) : 89   Mode  :character
 Mean   :308.0   [18,19) : 86
 3rd Qu.:461.5   [23,Inf): 78
 Max.   :615.0   [17,18) : 75
                 (Other) : 40
      age            plane            paper        distance
 Min.   : 4.00   Length:615         80 :248   Min.   : 0.000
 1st Qu.:11.00   Class :character   120:367   1st Qu.: 4.350
 Median :20.00   Mode  :character             Median : 6.500
 Mean   :22.11                               Mean   : 6.873
 3rd Qu.:28.00                               3rd Qu.: 9.000
 Max.   :64.00                               Max.   :20.100

 distance.pred
```

Model distance ~ paper

```
m2 <- lm(distance ~ paper, data = paperplanes)
```

```
Call:
lm(formula = distance ~ paper, data = paperplanes)

Residuals:
    Min      1Q  Median      3Q     Max
-7.2756 -2.3756 -0.3756  2.2244 12.7244

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   6.1298     0.2192  27.958  < 2e-16 ***
paper120      1.2458     0.2838   4.389 1.34e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.453 on 613 degrees of freedom
Multiple R-squared:  0.03047,   Adjusted R-squared:  0.02889
F-statistic: 19.27 on 1 and 613 DF,  p-value: 1.339e-05
```

Linear model with categorical predictors

```
m2 <- lm(distance ~ paper, data = paperplanes)
```

which corresponds to

$$y_i = a + bx_i + \varepsilon_i$$

$$distance_i = a + b_{paper120} + \varepsilon_i$$

Model distance ~ paper

```
m2 <- lm(distance ~ paper, data = paperplanes)
```

```
Call:
lm(formula = distance ~ paper, data = paperplanes)

Residuals:
    Min      1Q  Median      3Q     Max
-7.2756 -2.3756 -0.3756  2.2244 12.7244

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   6.1298     0.2192  27.958  < 2e-16 ***
paper120      1.2458     0.2838   4.389 1.34e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.453 on 613 degrees of freedom
Multiple R-squared:  0.03047,   Adjusted R-squared:  0.02889
F-statistic: 19.27 on 1 and 613 DF,  p-value: 1.339e-05
```

Effects: Estimated Distance ~ paper

```
summary(allEffects(m2))
```

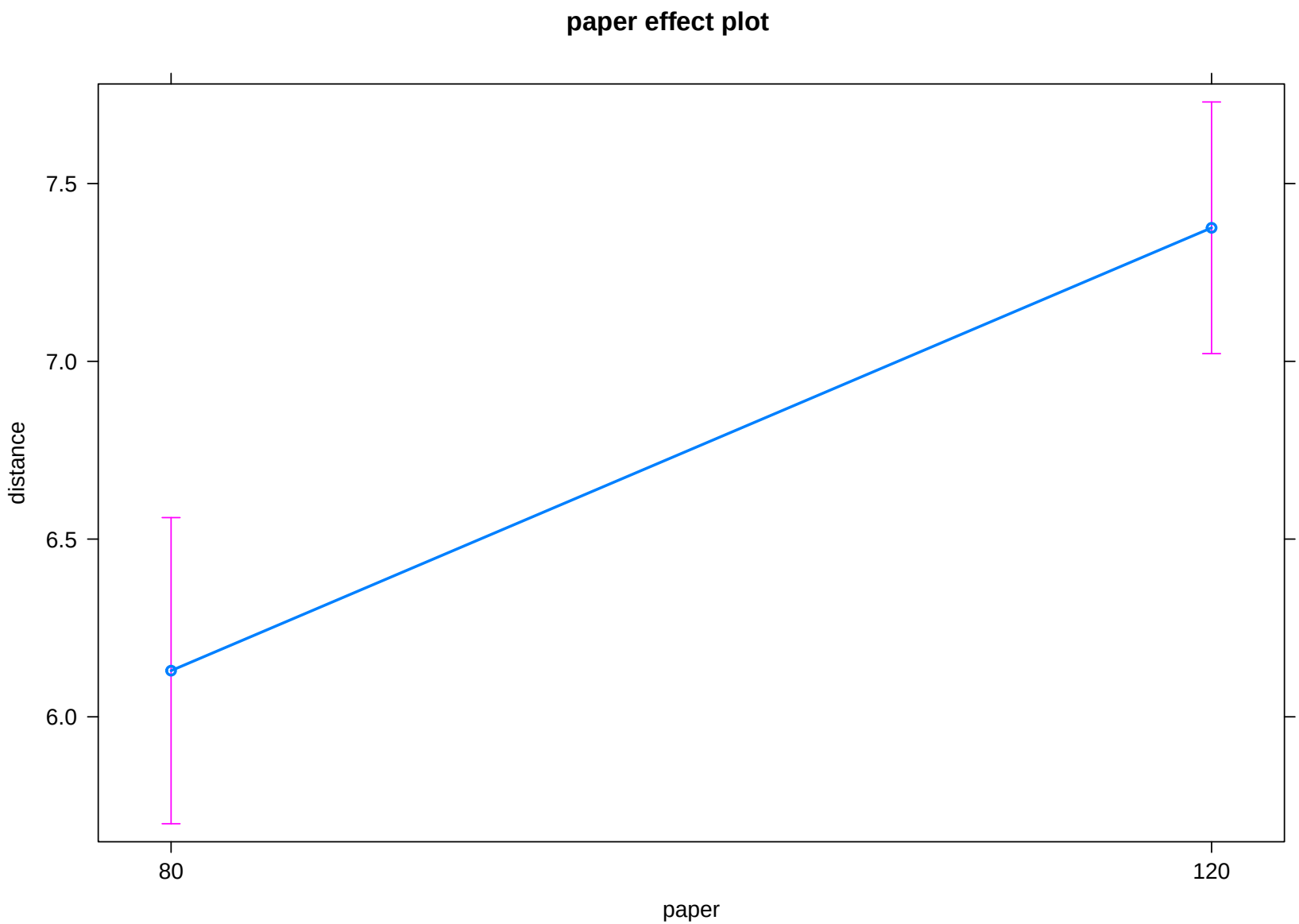
```
 model: distance ~ paper

 paper effect
paper
      80       120
6.129839 7.375613

 Lower 95 Percent Confidence Limits
paper
      80       120
5.699269 7.021668

 Upper 95 Percent Confidence Limits
paper
      80       120
6.560408 7.729558
```

Presenting model results

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	6.13	0.22	27.96	0
paper120	1.25	0.28	4.39	0

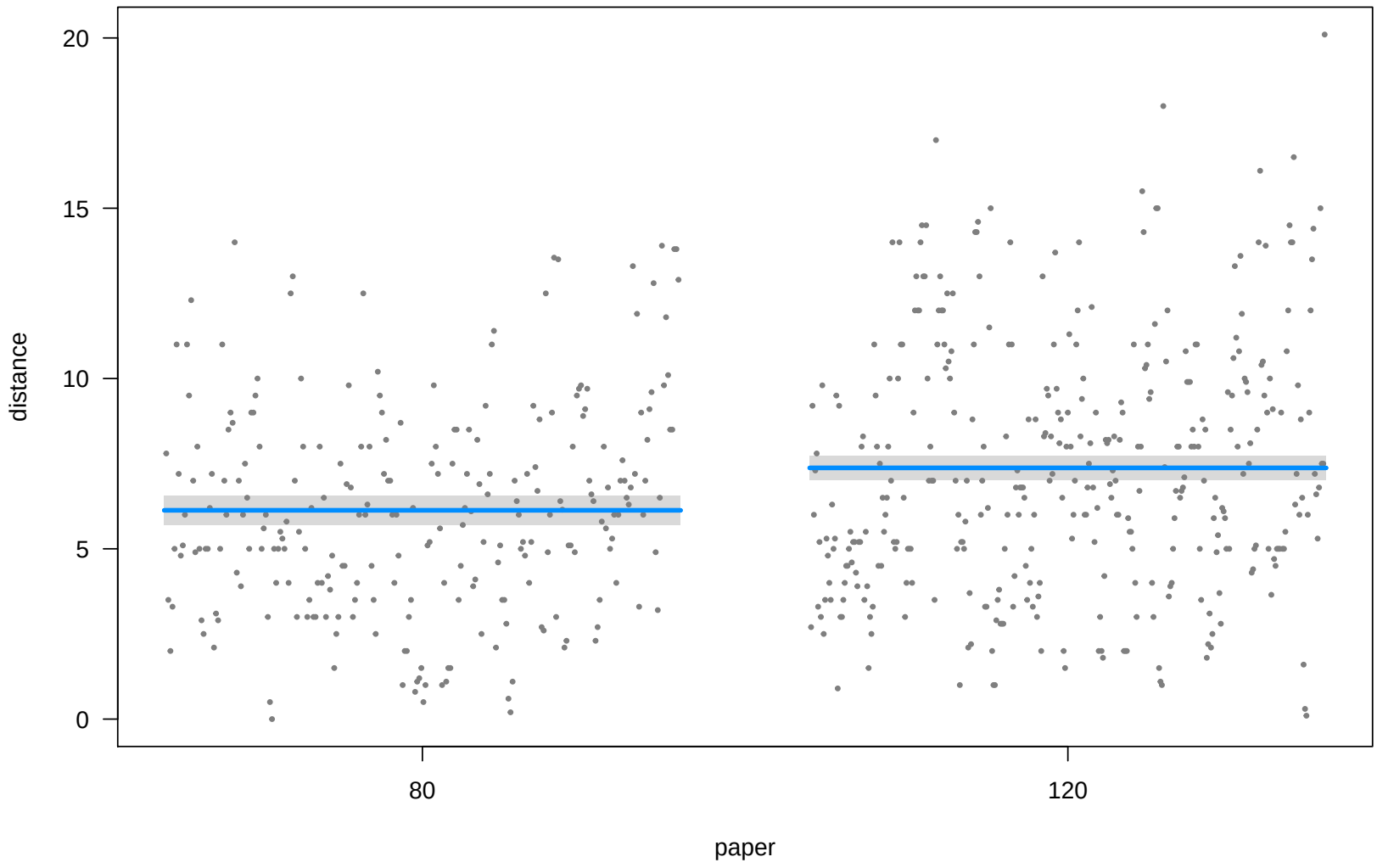
Plot

```
plot(allEffects(m2))
```

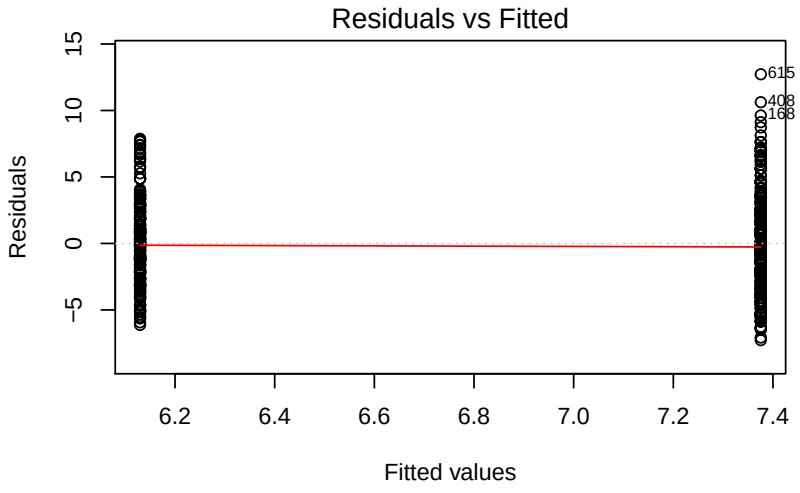


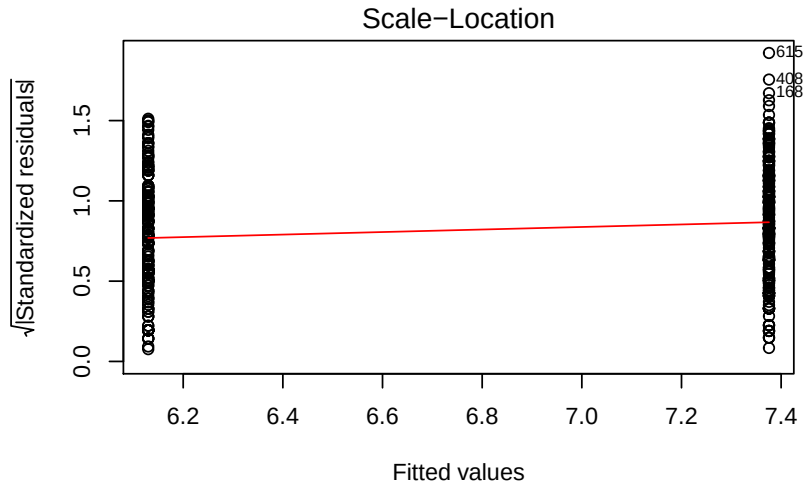
Plot (visreg)

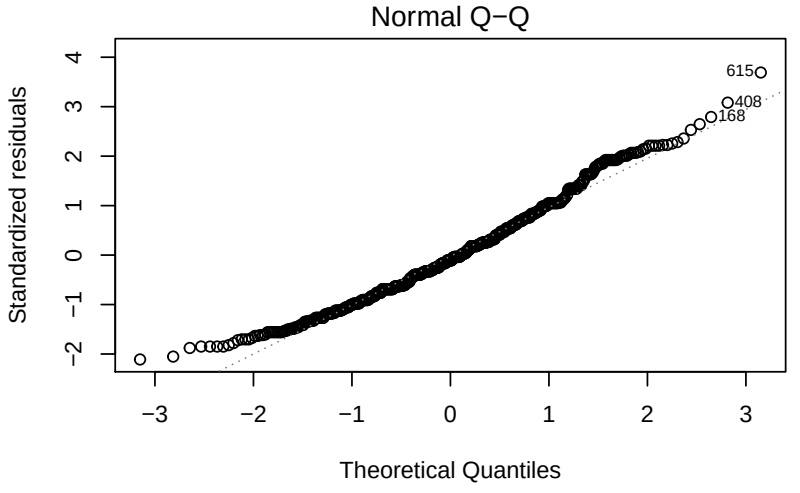
```
visreg(m2)
```

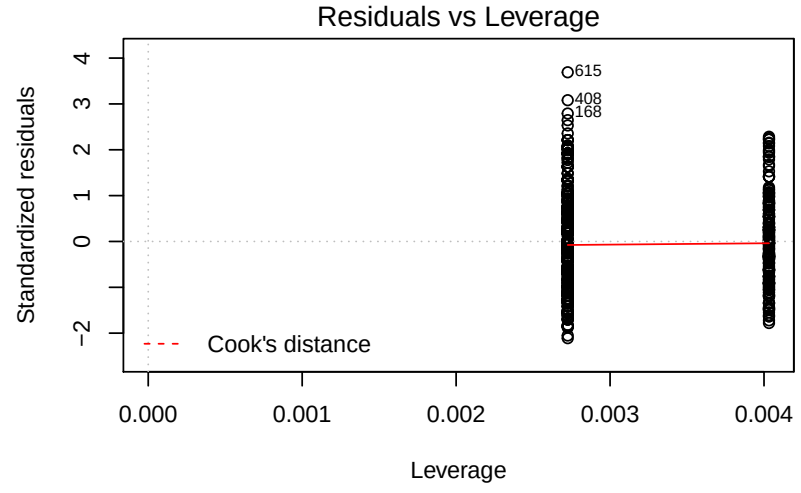


Model checking: residuals



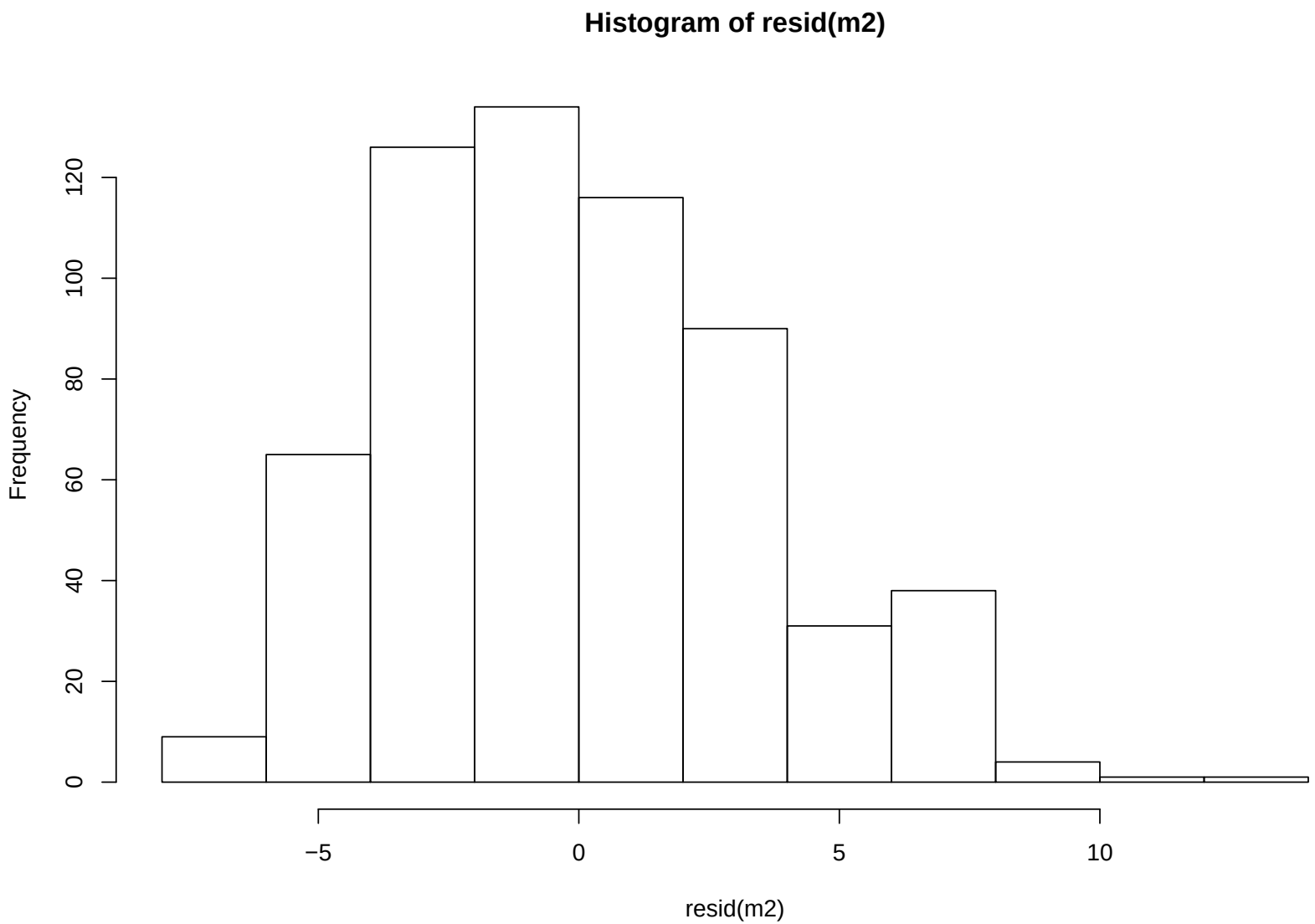






Model checking: residuals

```
hist(resid(m2))
```



Exercise: Does distance vary with gender?

Combining continuous and categorical predictors

Predicting distance based on age and paper type

```
lm(distance ~ paper + age, data = paperplanes)
```

$$y_i = a + bx_i + \varepsilon_i$$

$$distance_i = a + b_{paper120} + c \cdot age_i + \varepsilon_i$$

Predicting distance based on age and paper type

```
Call:
lm(formula = distance ~ age + paper, data = paperplanes)

Residuals:
    Min      1Q  Median      3Q     Max
-7.1092 -2.4753 -0.3576  2.2523 12.5892

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  5.69210    0.33641  16.920  < 2e-16 ***
age          0.01774    0.01035   1.714   0.0871 .
paper120     1.32192    0.28683   4.609 4.93e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.447 on 612 degrees of freedom
Multiple R-squared:  0.0351,    Adjusted R-squared:  0.03195
F-statistic: 11.13 on 2 and 612 DF,  p-value: 1.784e-05
```

Presenting model results

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	5.69	0.34	16.92	0.00
age	0.02	0.01	1.71	0.09
paper120	1.32	0.29	4.61	0.00

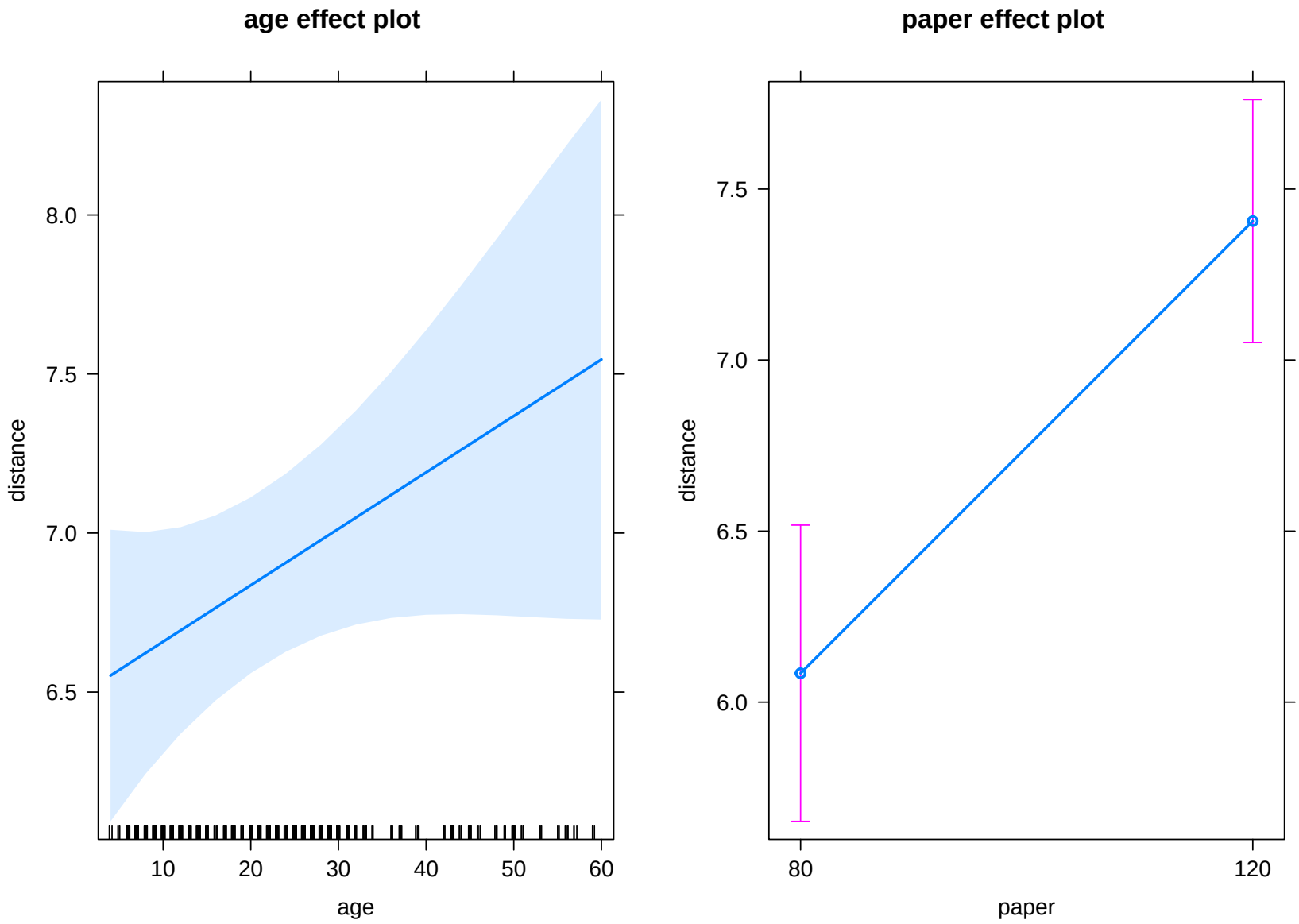
Estimated distance

```
summary(allEffects(multreg))
```

```
 model: distance ~ age + paper

 age effect
age
       4        20        30        50        60
6.551921 6.835779 7.013191 7.368014 7.545425

 Lower 95 Percent Confidence Limits
age
       4        20        30        50        60
6.093516 6.559431 6.696578 6.738709 6.728156

 Upper 95 Percent Confidence Limits
age
       4        20        30        50        60
7.010326 7.112127 7.329803 7.997318 8.362694

 paper effect
paper
      80      120
6.084400 7.406318

 Lower 95 Percent Confidence Limits
paper
      80      120
5.651366 7.051182

 Upper 95 Percent Confidence Limits
```

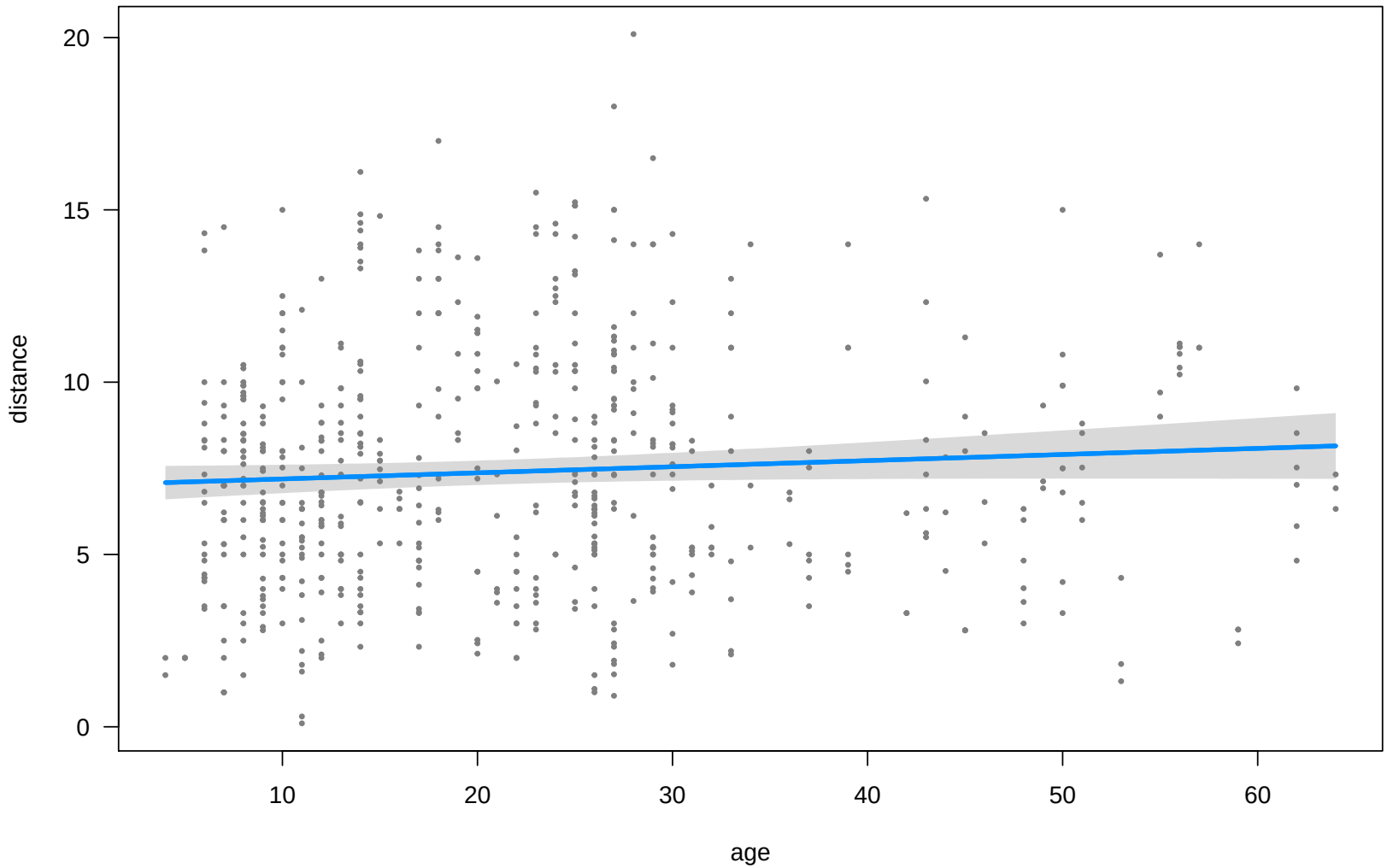
Plot

```
plot(allEffects(multreg))
```

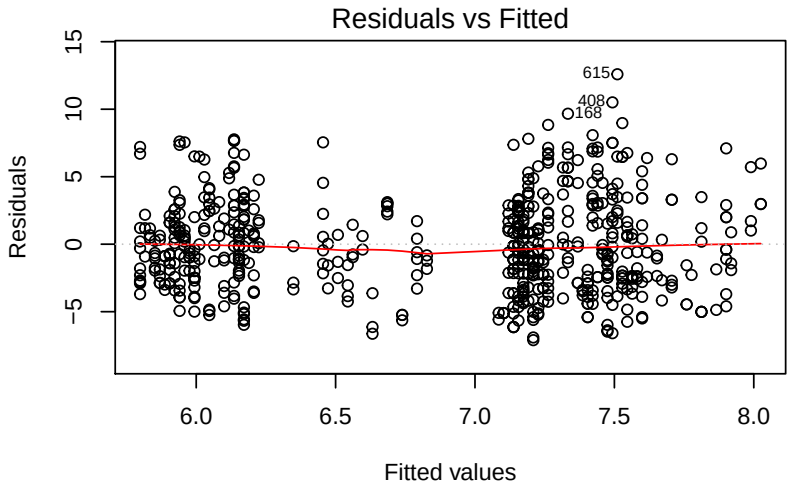


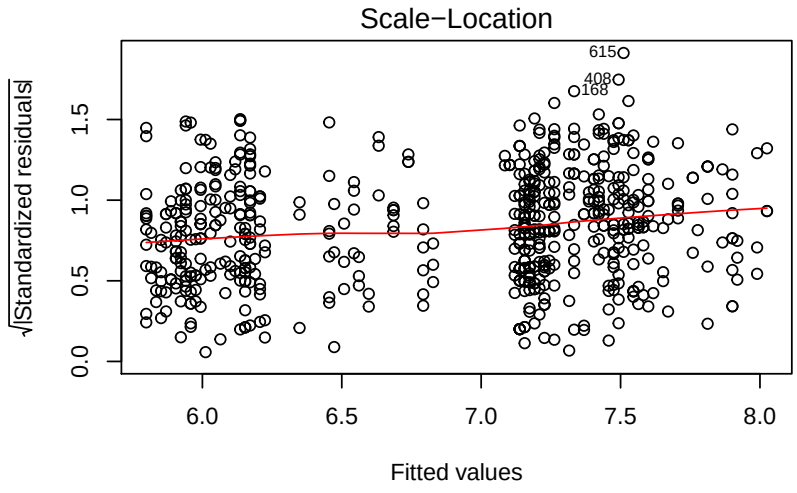
Plot (visreg)

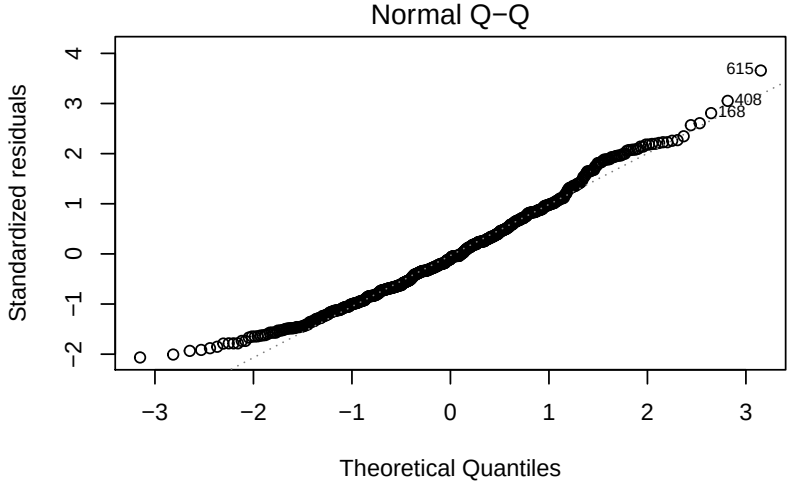
```
visreg(multreg)
```

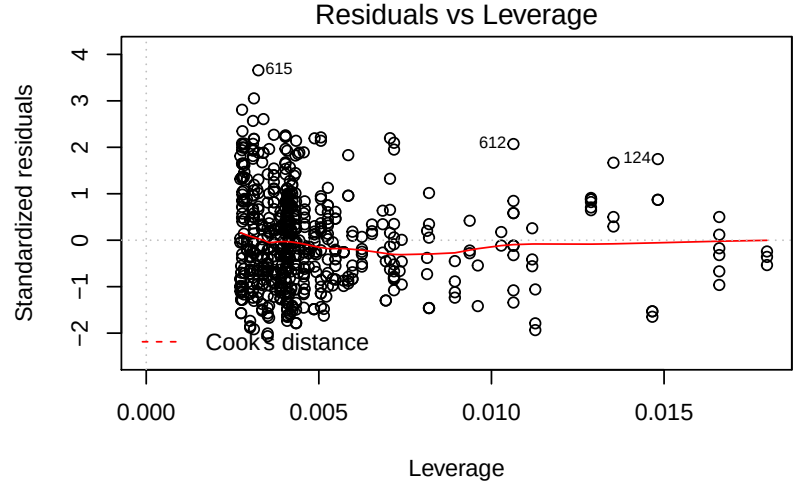


Model checking: residuals



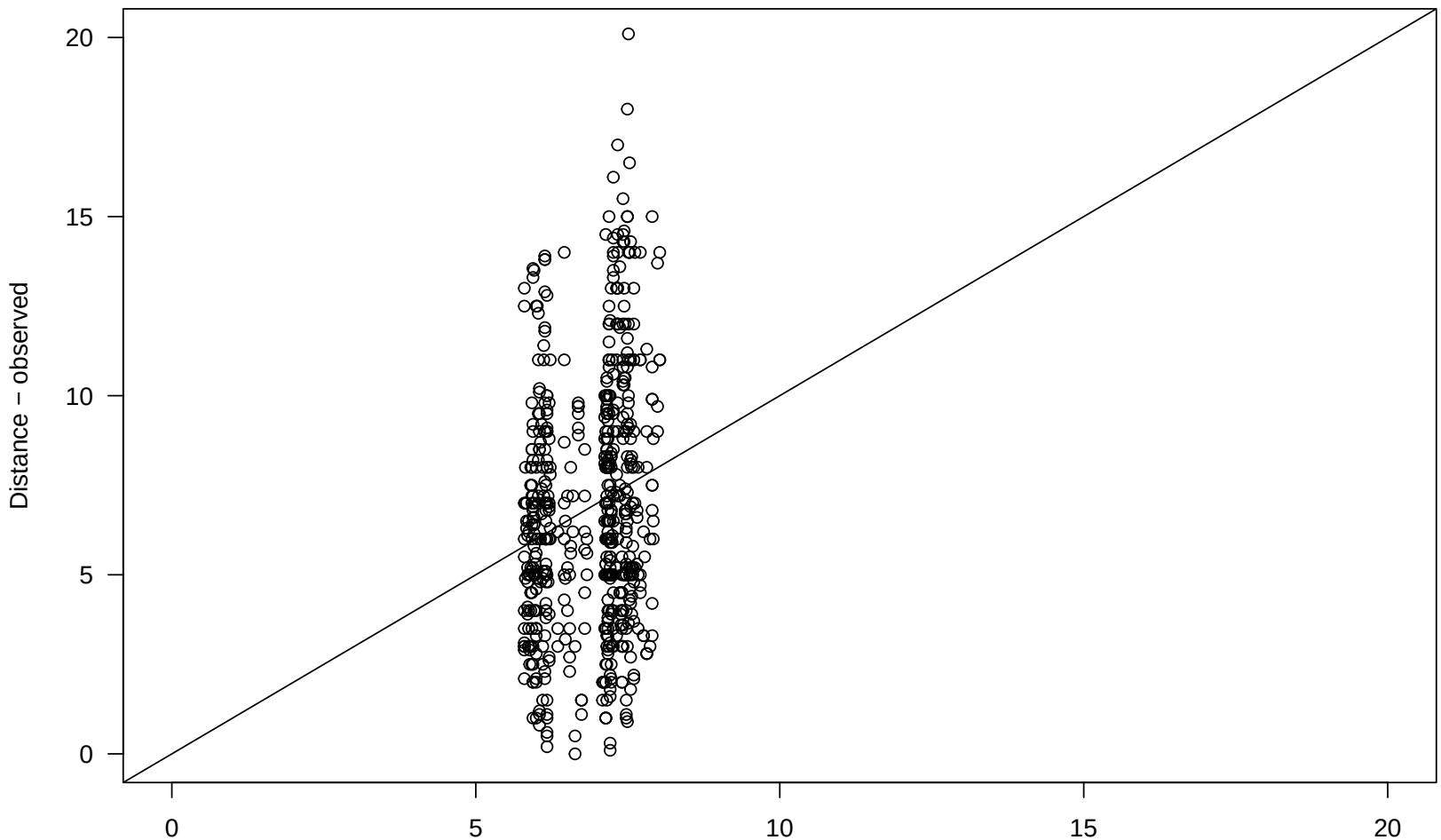






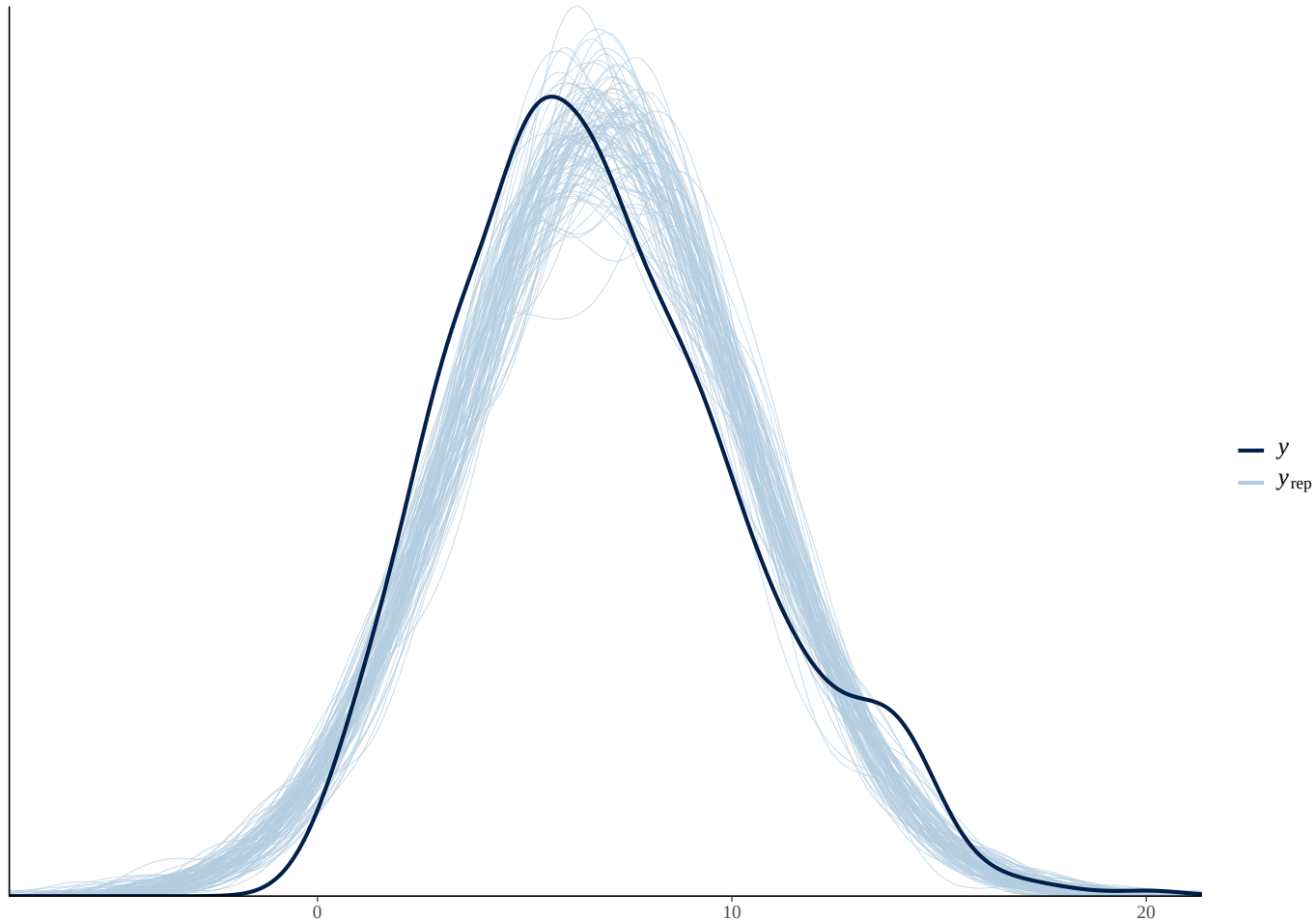
How good is this model? Calibration plot

```
paperplanes$distance.pred <- fitted(multreg)
plot(paperplanes$distance.pred, paperplanes$distance, xlab = "Di
abline(a = 0, b = 1)
```



Model checking with simulated data

```
library(bayesplot)
sims <- simulate(multreg, nsim = 100)
ppc_dens_overlay(paperplanes$distance, yrep = t(as.matrix(sims)))
```



Extra exercises

- mammal sleep: Are sleep patterns related to diet?

Extra exercises

- mammal sleep: Are sleep patterns related to diet?
- iris: Predict petal length $\sim$ petal width and species

Generalised Linear Models: Logistic regression

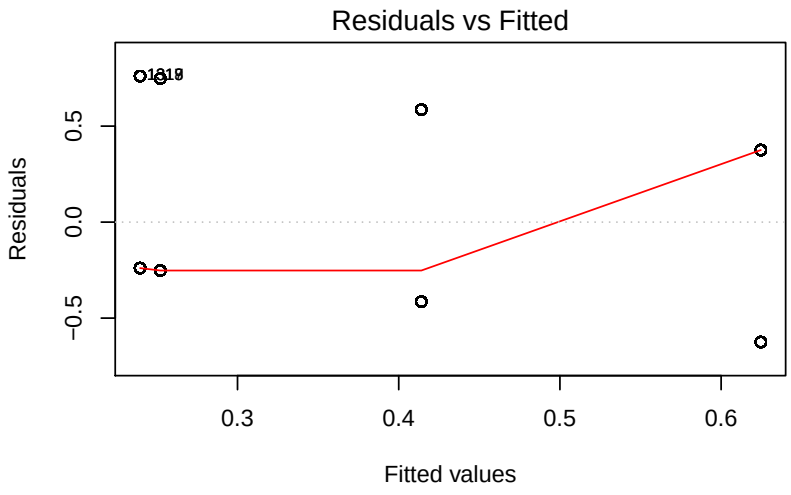
Q: Survival of passengers on the Titanic ~ Class

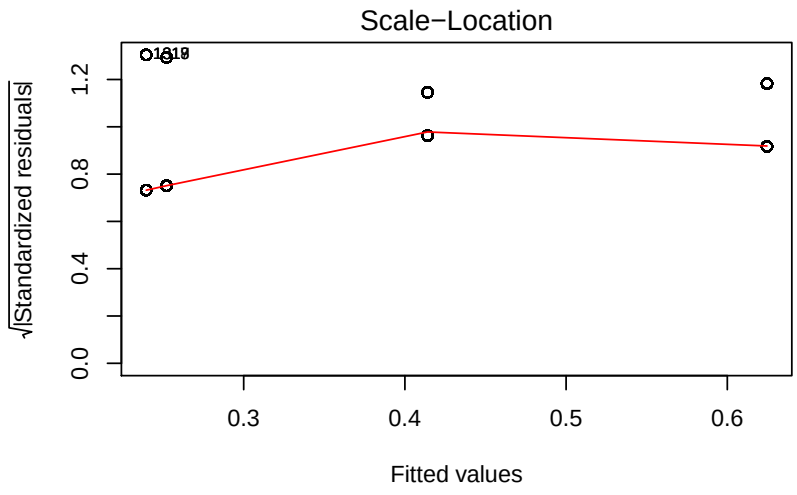
Read `titanic_long.csv` dataset.

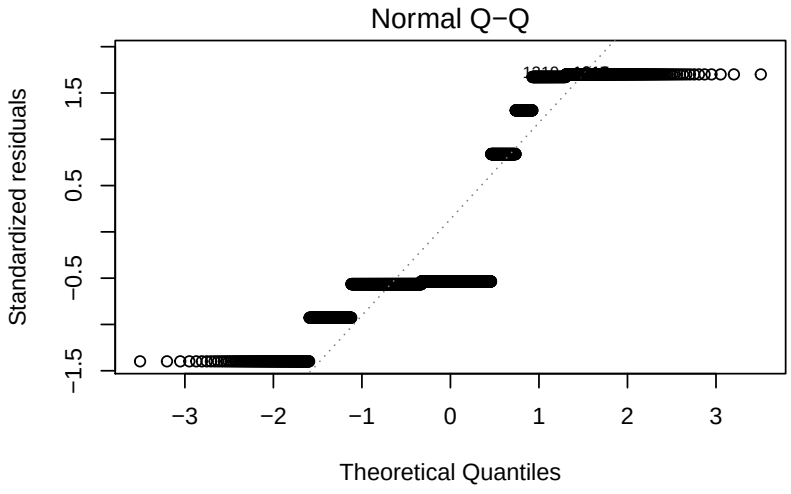
```
  class   age  sex survived
1 first adult male        1
2 first adult male        1
3 first adult male        1
4 first adult male        1
5 first adult male        1
6 first adult male        1
```

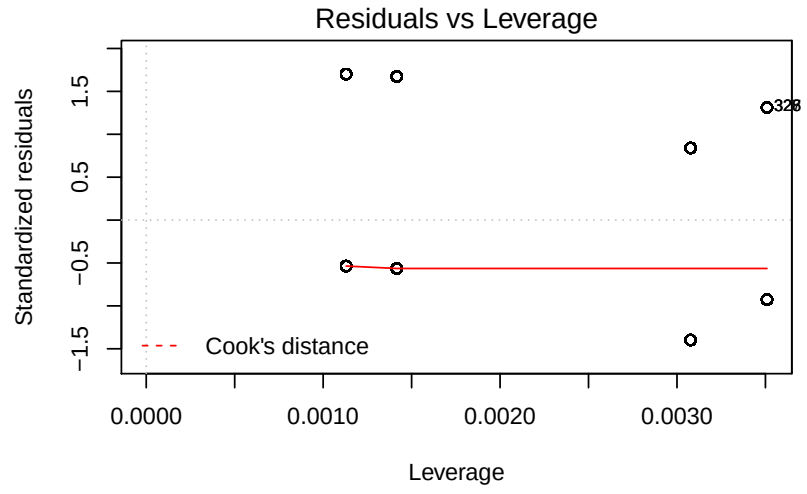
Let's fit linear model:

```
m5 <- lm(survived ~ class, data = titanic)
```

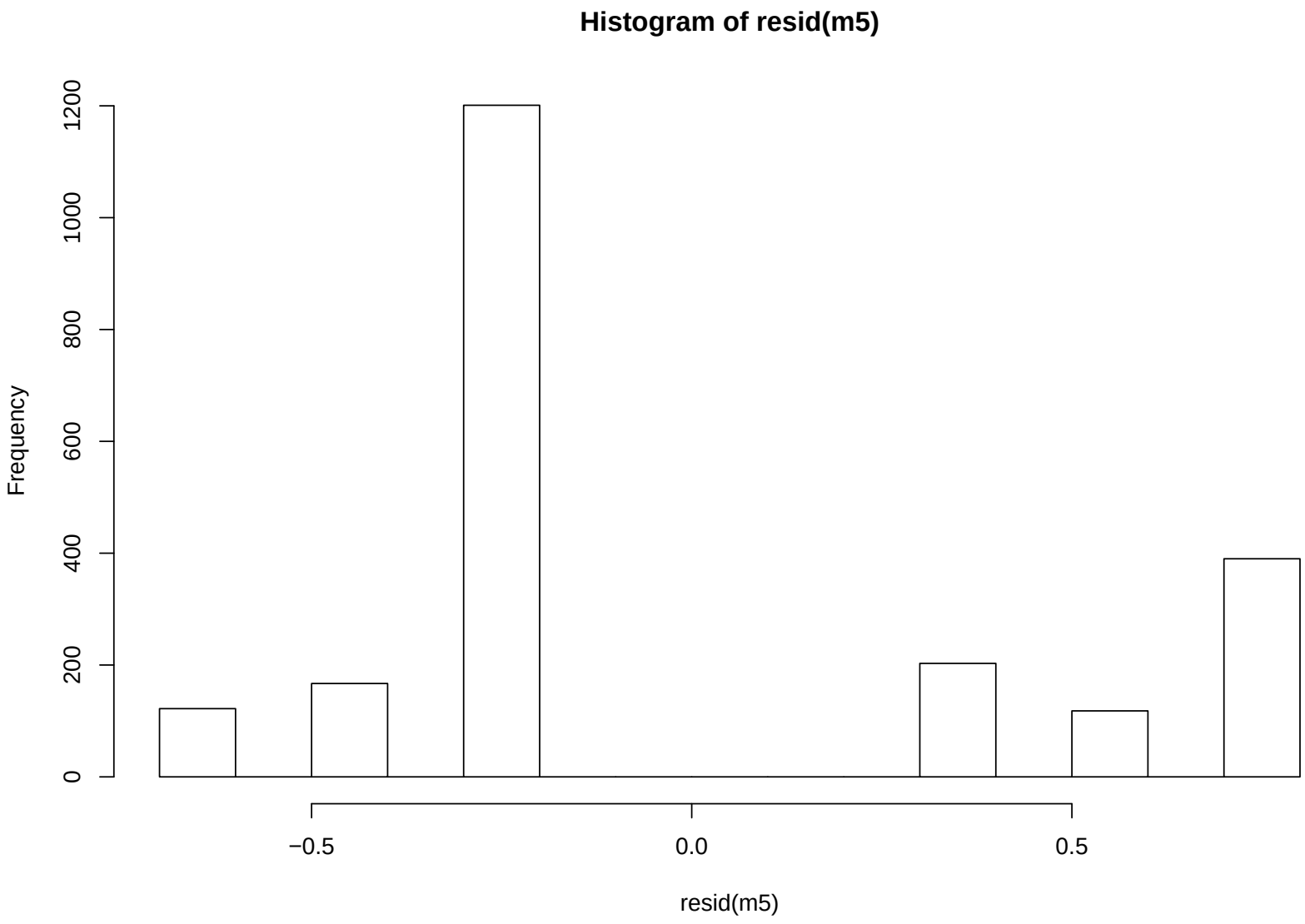








Weird residuals!



What if your residuals are clearly non-normal? | And variance not constant (heteroscedasticity)?

- Binary variables (0/1)

What if your residuals are clearly non-normal? | And variance not constant (heteroscedasticity)?

- Binary variables (0/1)
- Counts (0, 1, 2, 3, . . .)

Generalised Linear Models

1. **Response variable** - distribution `family`

Generalised Linear Models

1. **Response variable** - distribution `family`
   - Bernouilli - Binomial

Generalised Linear Models

1. **Response variable** - distribution `family`
   - Bernouilli - Binomial
   - Poisson

Generalised Linear Models

1. **Response variable** - distribution `family`
   - Bernouilli - Binomial
   - Poisson
   - Gamma

Generalised Linear Models

1. **Response variable** - distribution `family`
   - Bernouilli - Binomial
   - Poisson
   - Gamma
   - etc

Generalised Linear Models

1. **Response variable** - distribution `family`
   - Bernouilli - Binomial
   - Poisson
   - Gamma
   - etc
2. **Predictors** (continuous or categorical)

Generalised Linear Models

1. **Response variable** - distribution `family`
   - Bernouilli - Binomial
   - Poisson
   - Gamma
   - etc
2. **Predictors** (continuous or categorical)
3. **Link function**

Generalised Linear Models

1. **Response variable** - distribution `family`
   - Bernouilli - Binomial
   - Poisson
   - Gamma
   - etc
2. **Predictors** (continuous or categorical)
3. **Link function**
   - Gaussian: identity

Generalised Linear Models

1. **Response variable** - distribution `family`
   - Bernouilli - Binomial
   - Poisson
   - Gamma
   - etc
2. **Predictors** (continuous or categorical)
3. **Link function**
   - Gaussian: identity
   - Binomial: logit, probit

Generalised Linear Models

1. **Response variable** - distribution `family`
   - Bernouilli - Binomial
   - Poisson
   - Gamma
   - etc
2. **Predictors** (continuous or categorical)
3. **Link function**
   - Gaussian: identity
   - Binomial: logit, probit
   - Poisson: log...

Generalised Linear Models

1. **Response variable** - distribution `family`
   - Bernouilli - Binomial
   - Poisson
   - Gamma
   - etc
2. **Predictors** (continuous or categorical)
3. **Link function**
   - Gaussian: identity
   - Binomial: logit, probit
   - Poisson: log...
   - See `family`.

The modelling process

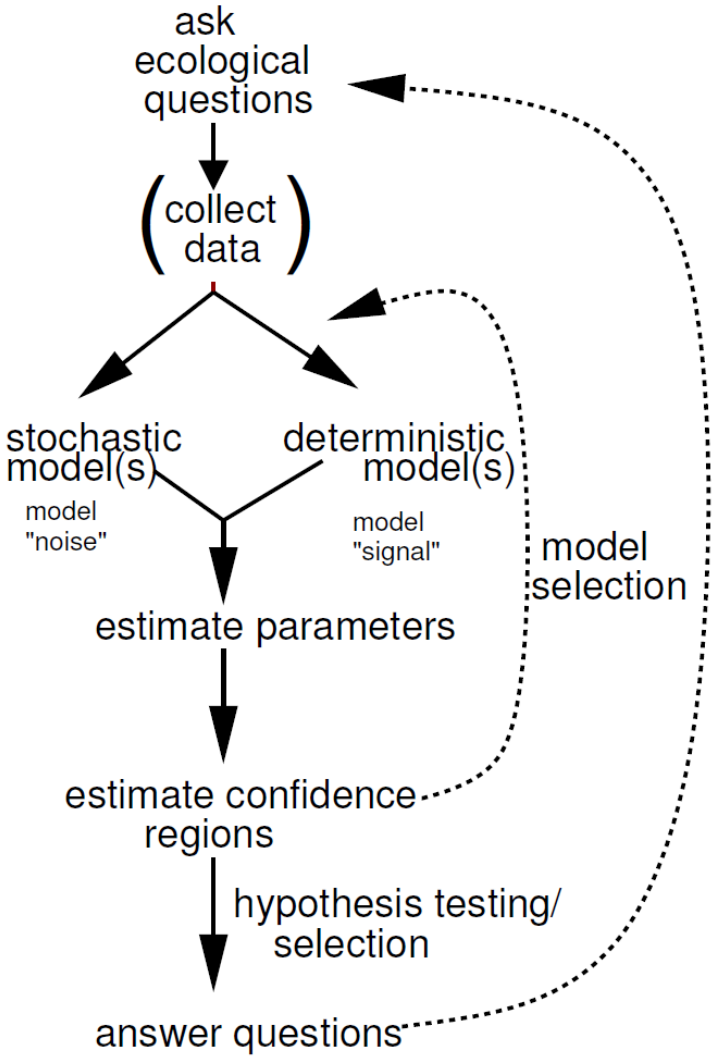


Figure 1.5 Flow of the modeling process.

Bolker 2008

Bernouilli - Binomial distribution (Logistic regression)

- Response variable: Yes/No (e.g. survival, sex, presence/absence)

$$logit(p) = \ln\left(\frac{p}{1-p}\right)$$

Then

$$Pr(alive) = a + bx$$
$$logit(Pr(alive)) = a + bx$$
$$Pr(alive) = invlogit(a + bx) = \frac{e^{a+bx}}{1 + e^{a+bx}}$$

Bernouilli - Binomial distribution (Logistic regression)

- Response variable: Yes/No (e.g. survival, sex, presence/absence)
- Link function: `logit` (others possible, see `family`).

$$logit(p) = \ln\left(\frac{p}{1-p}\right)$$

Then

$$Pr(alive) = a + bx$$
$$logit(Pr(alive)) = a + bx$$
$$Pr(alive) = invlogit(a + bx) = \frac{e^{a+bx}}{1 + e^{a+bx}}$$

Back to survival of Titanic passengers

How many survived in each class?

```
table(titanic$class, titanic$survived)
```

```
          0   1
  crew   673 212
  first  122 203
  second 167 118
  third  528 178
```

Back to survival of Titanic passengers (dplyr)

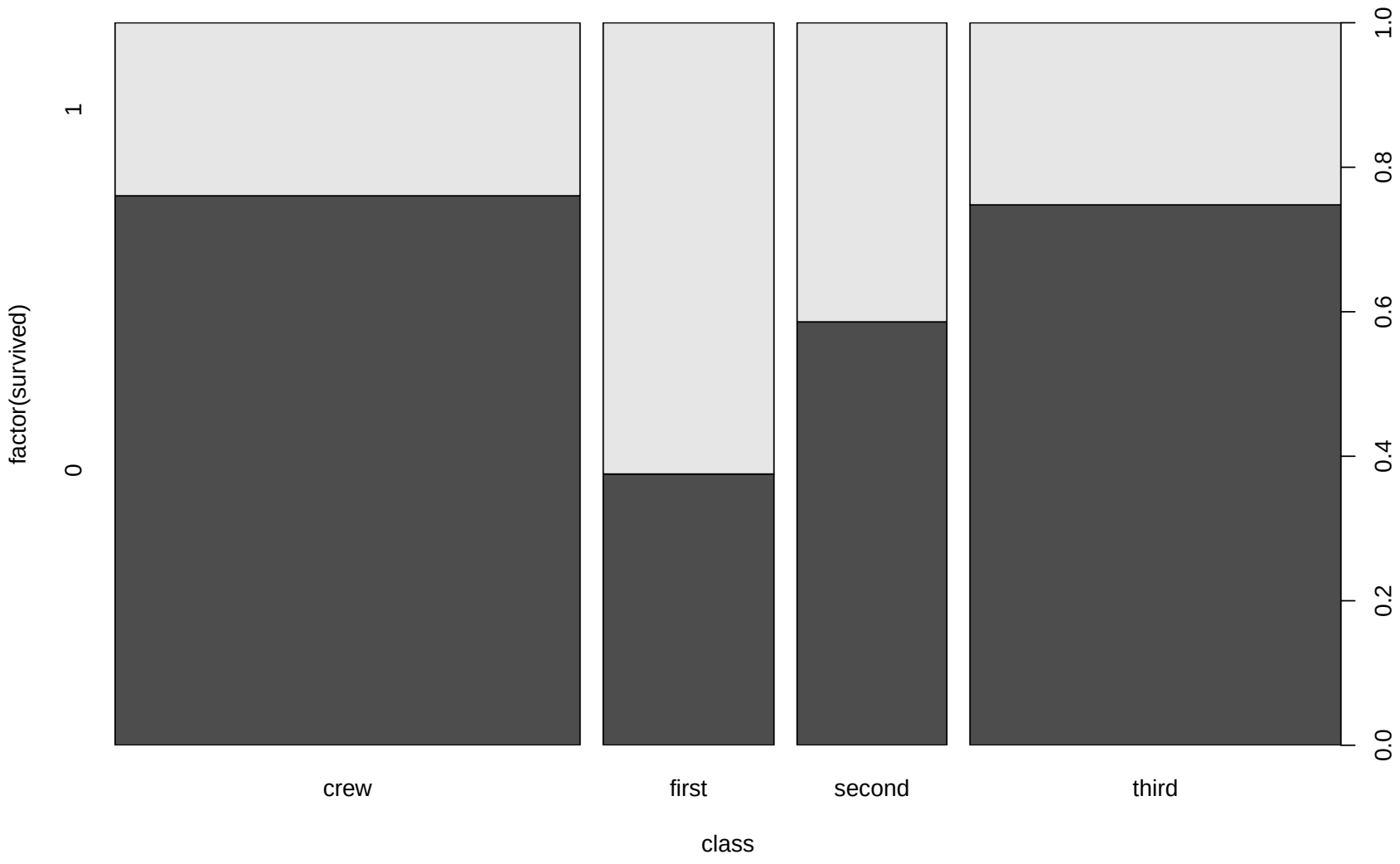
Passenger survival according to class

```
titanic %>%
  group_by(class, survived) %>%
  summarise(count = n())
```

```
# A tibble: 8 x 3
# Groups:   class [?]
  class  survived count
  <fct>     <int> <int>
1 crew          0   673
2 crew          1   212
3 first         0   122
4 first         1   203
5 second        0   167
6 second        1   118
7 third         0   528
8 third         1   178
```

Or graphically...

```
plot(factor(survived) ~ class, data = titanic)
```



Fitting GLMs in R: `glm`

```
tit.glm <- glm(survived ~ class, data = titanic, family = binomial(link = "logit"))
```

which corresponds to

$$logit(Pr(survival)_i) = a + b \cdot class_i$$
$$logit(Pr(survival)_i) = a + b_{first} + c_{second} + d_{third}$$

Fitting GLMs in R: glm

```r
tit.glm <- glm(survived ~ class, data = titanic, family = binomial(link = "logit"))
```

```
Call:
glm(formula = survived ~ class, family = binomial(link = "logit"),
    data = titanic)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.3999  -0.7623  -0.7401   0.9702   1.6906

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.15516    0.07876 -14.667  < 2e-16 ***
classfirst   1.66434    0.13902  11.972  < 2e-16 ***
classsecond  0.80785    0.14375   5.620 1.91e-08 ***
classthird   0.06785    0.11711   0.579    0.562
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 2769.5  on 2200  degrees of freedom
Residual deviance: 2588.6  on 2197  degrees of freedom
AIC: 2596.6

Number of Fisher Scoring iterations: 4
```

These estimates are in logit scale!

Interpreting logistic regression output

Parameter estimates (logit-scale)

```
(Intercept)  classfirst classsecond  classthird
-1.15515905  1.66434399  0.80784987  0.06784632
```

We need to back-transform: apply *inverse logit*
Crew probability of survival:

```
plogis(coef(tit.glm)[1])
```

```
(Intercept)
   0.239548
```

Looking at the data, the proportion of crew who survived is

```
[1] 0.239548
```

Q: Probability of survival for 1st class passengers?

```
plogis(coef(tit.glm)[1] + coef(tit.glm)[2])
```

```
(Intercept)
  0.6246154
```

Needs to add intercept (baseline) to the parameter estimate. Again this value matches the data:

```
sum(titanic$survived[titanic$class == "first"]) /
  nrow(titanic[titanic$class == "first", ])
```

```
[1] 0.6246154
```

Model interpretation using `effects` package

```
library(effects)
allEffects(tit.glm)
```

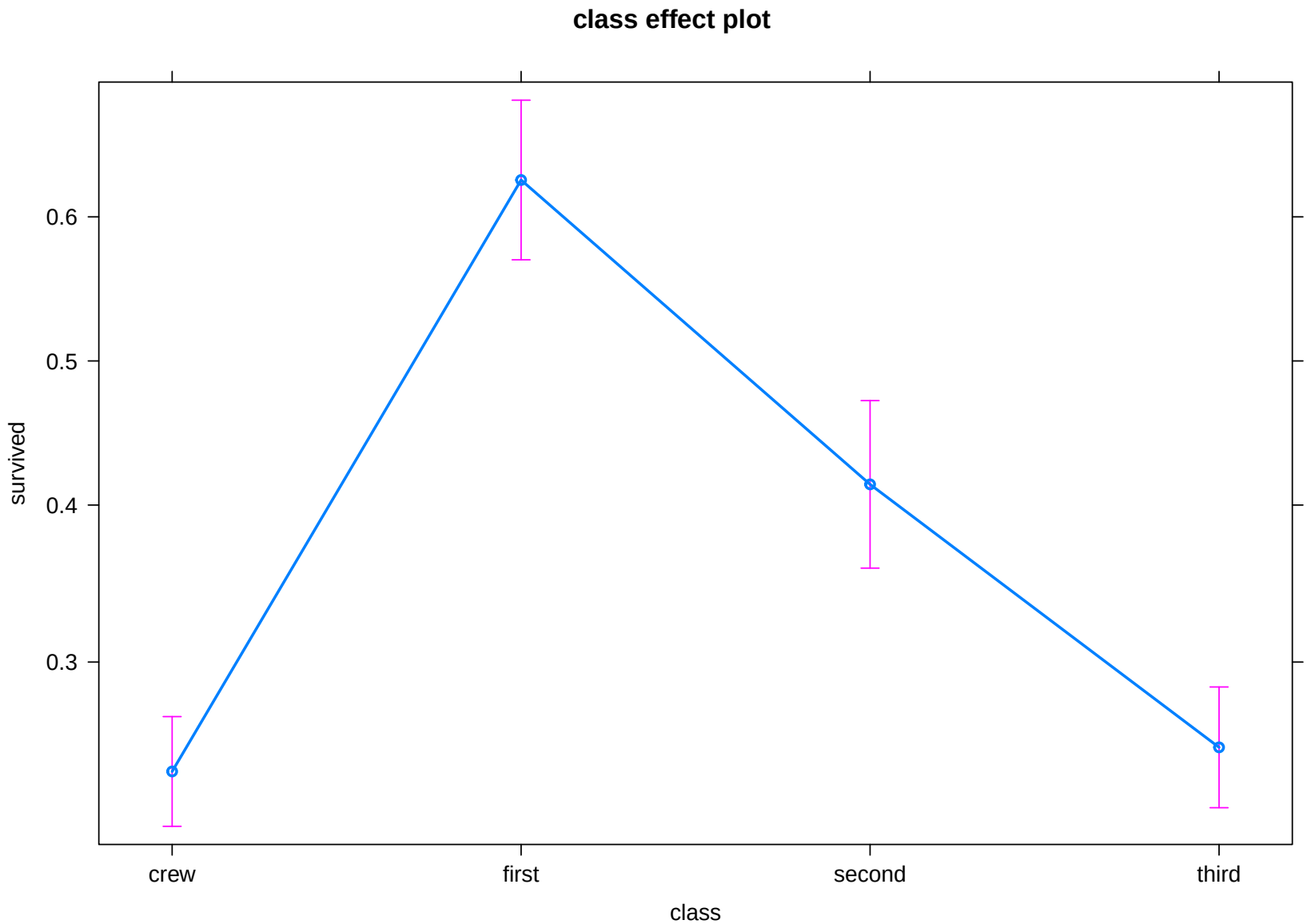
```
 model: survived ~ class

 class effect
class
     crew      first     second      third
0.2395480 0.6246154 0.4140351 0.2521246
```

Presenting model results

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-1.16	0.08	-14.67	0.00
classfirst	1.66	0.14	11.97	0.00
classsecond	0.81	0.14	5.62	0.00
classthird	0.07	0.12	0.58	0.56

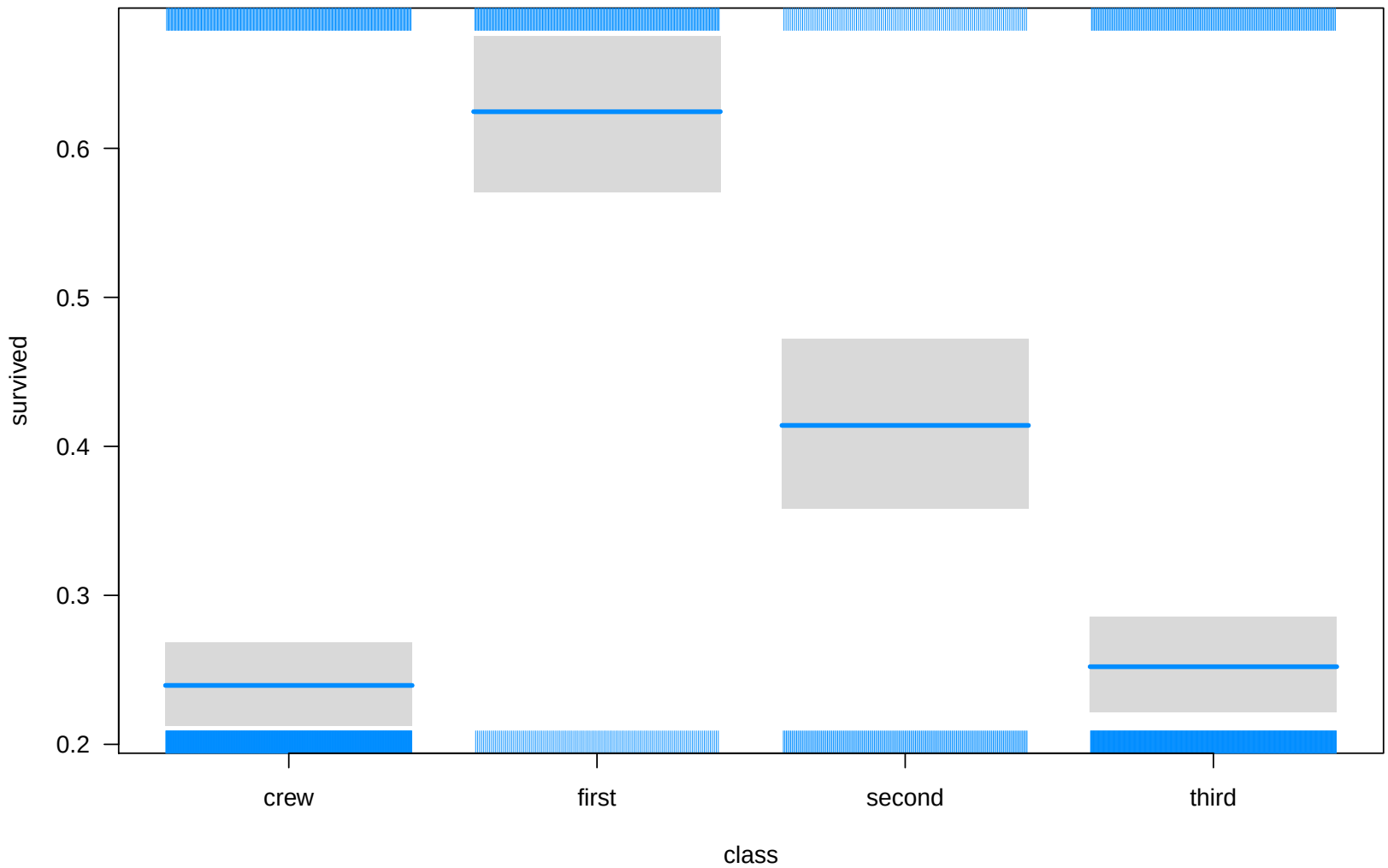
Visualising model: `effects` package

```
plot(allEffects(tit.glm))
```

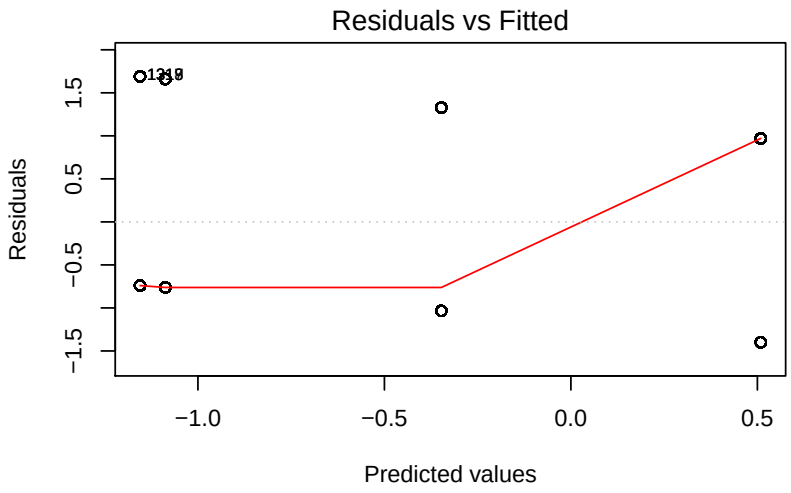


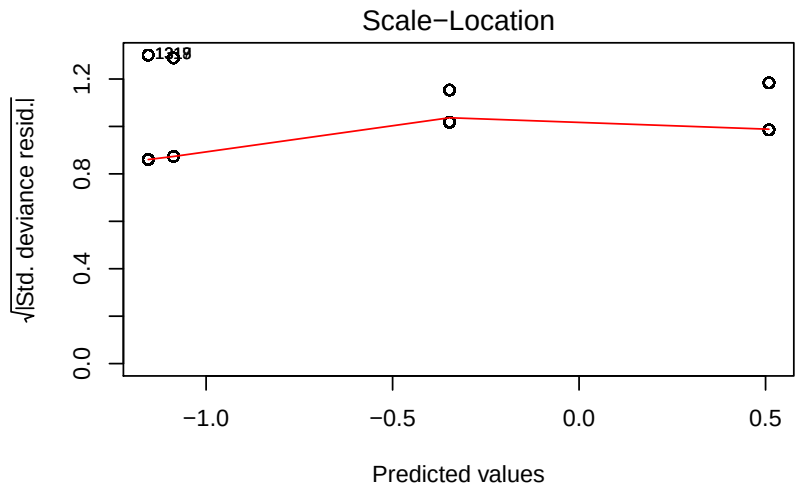
Visualising model: `visreg` package

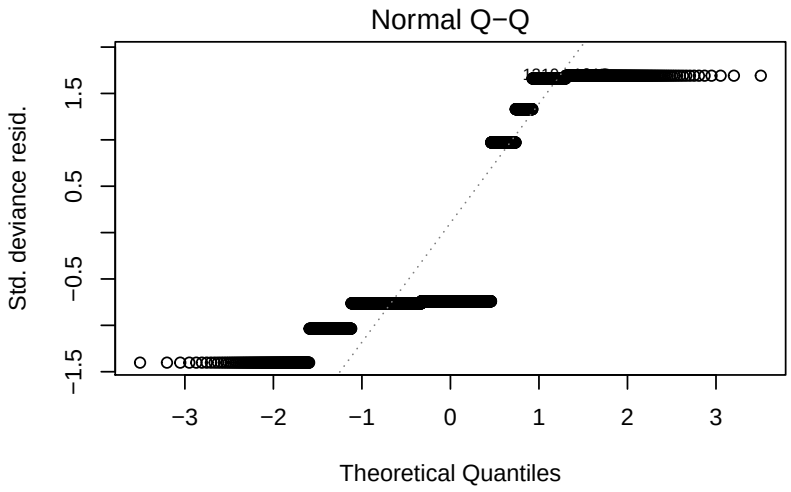
```
visreg(tit.glm, scale = "response")
```

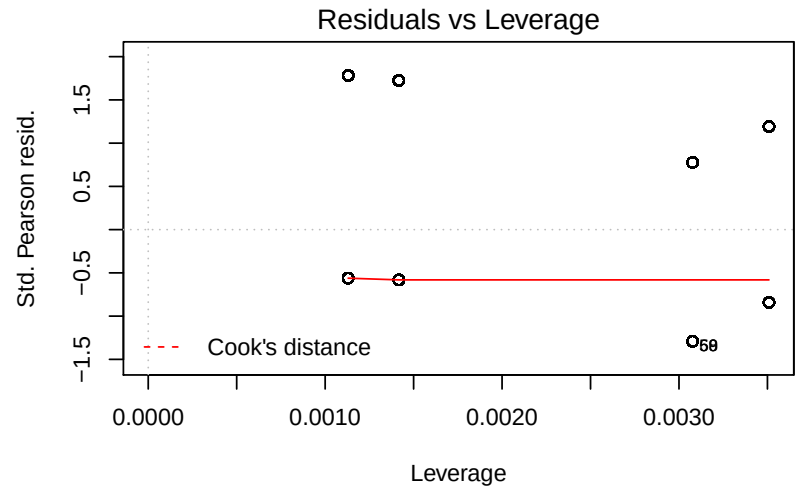


Logistic regression: model checking





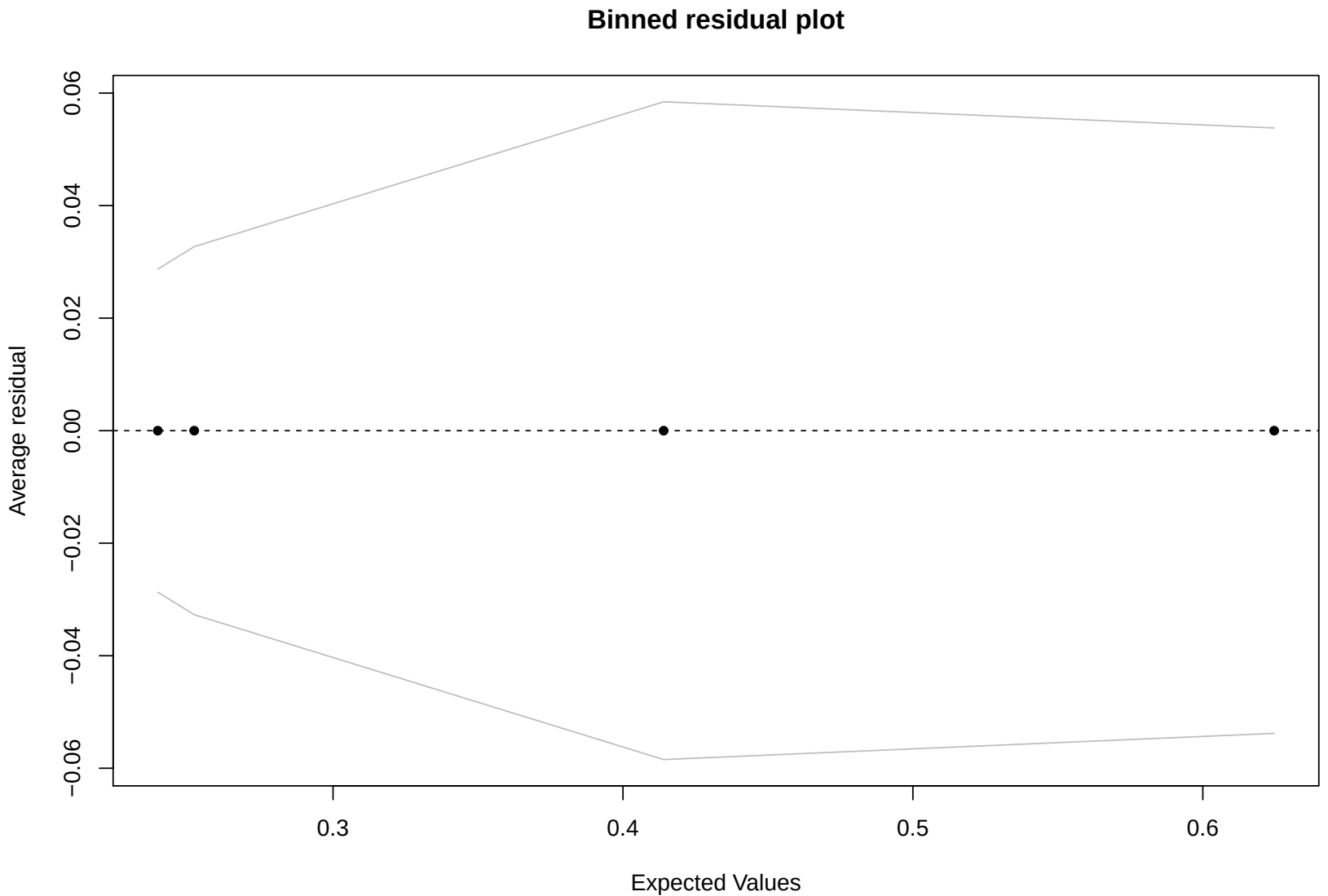





```
null device
          1
```

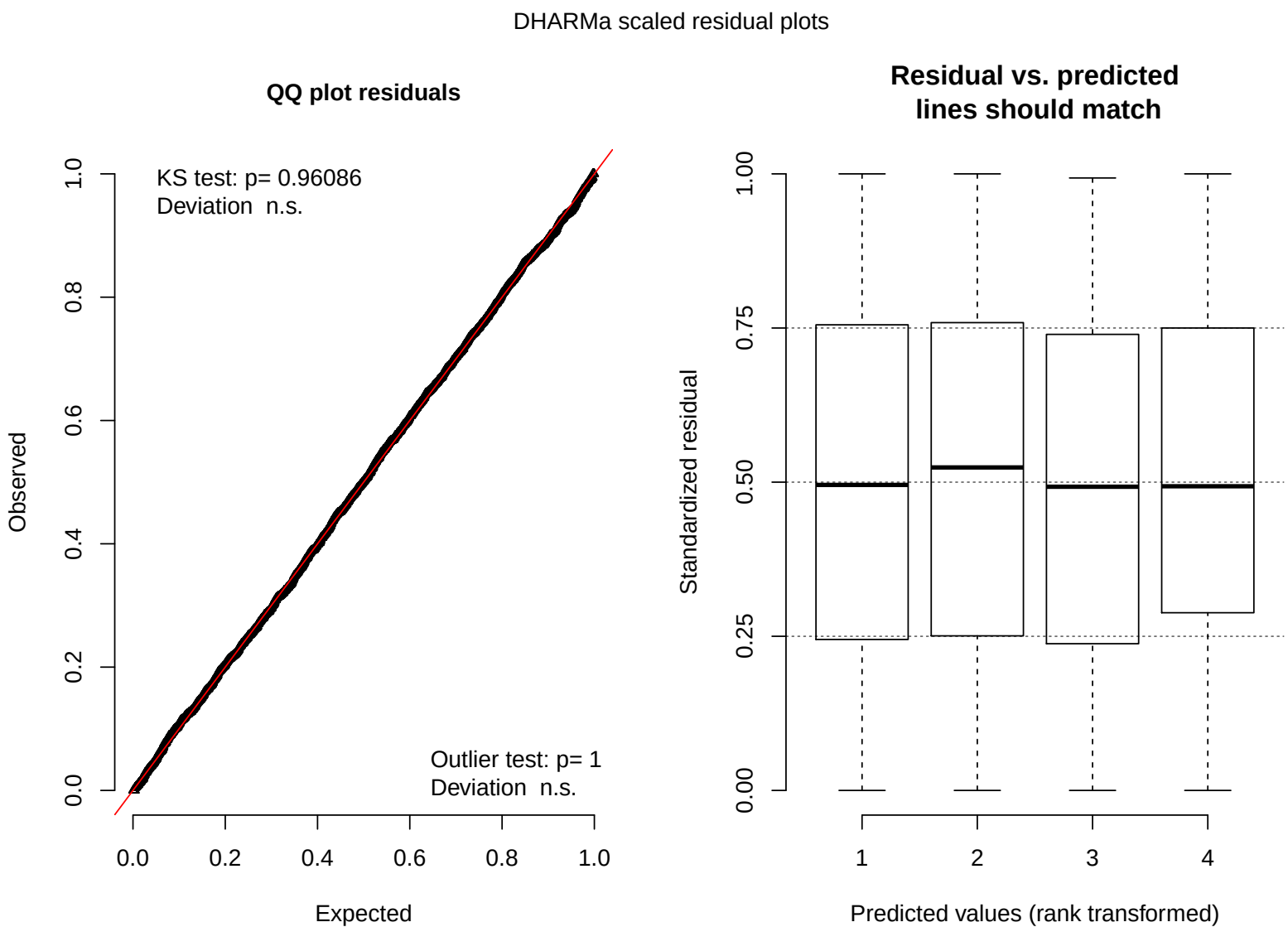
Binned residual plots for logistic regression

```
predvals <- predict(tit.glm, type="response")
arm::binnedplot(predvals, titanic$survived - predvals)
```



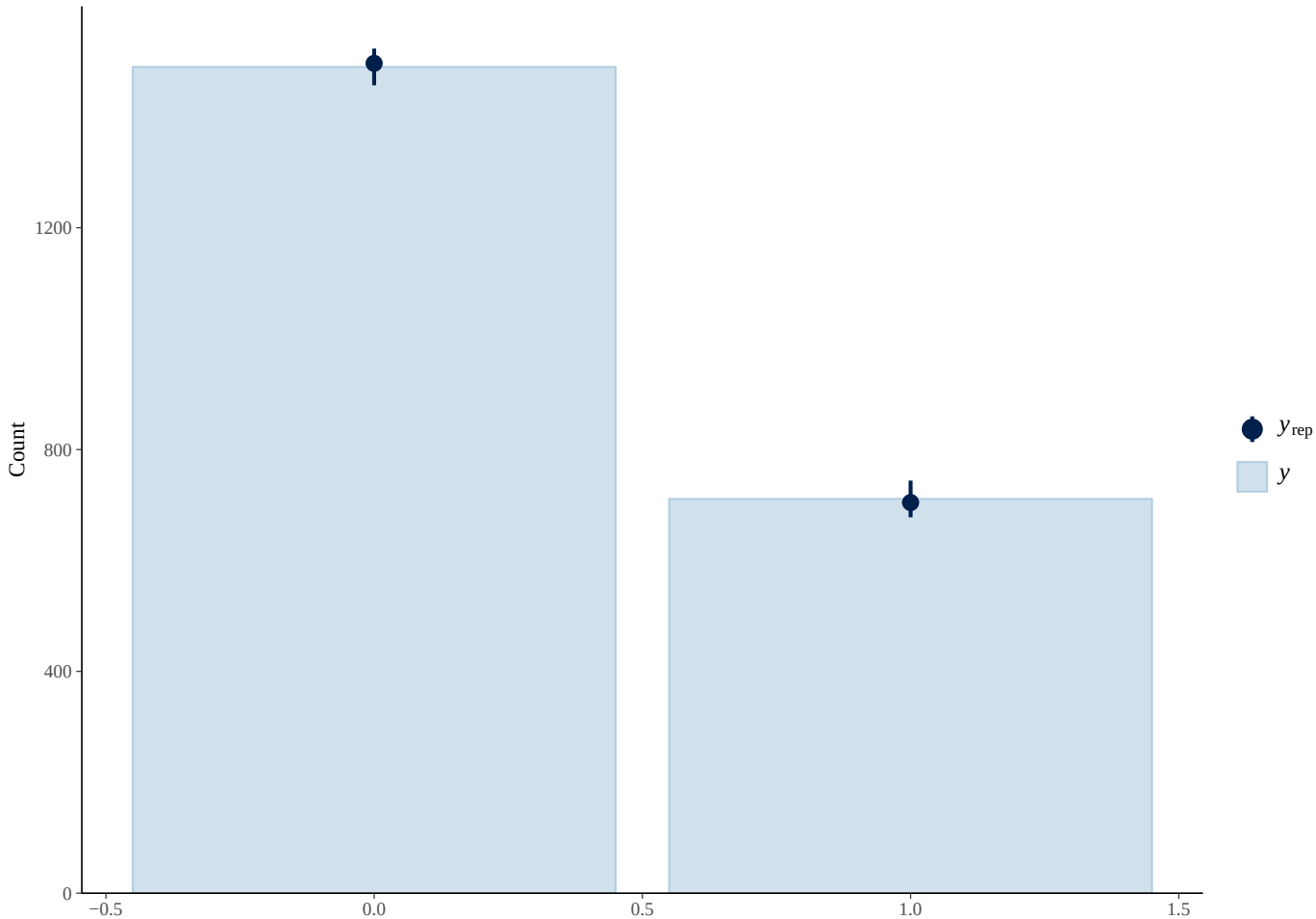
Residual diagnostics with DHARMa

```
library(DHARMa)
simulateResiduals(tit.glm, plot = TRUE)
```



Model checking with simulated data

```
library(bayesplot)
sims <- simulate(tit.glm, nsim = 100)
ppc_bars(titanic$survived, yrep = t(as.matrix(sims)))
```



Pseudo R-squared for GLMs

```
library(sjstats)
r2(tit.glm)
```

```
R-Squared for Generalized Linear Mixed Model

Cox & Snell's R-squared: 0.079
 Nagelkerke's R-squared: 0.110
```

But many caveats apply! (e.g. see here and here)

Recapitulating

1. Import data: `read.table` or `read.csv`

Recapitulating

1. Import data: `read.table` or `read.csv`
2. Check data: `summary`, `head`

Recapitulating

1. Import data: `read.table` or `read.csv`
2. Check data: `summary`, `head`
3. Plot data: `plot`

Recapitulating

1. Import data: `read.table` or `read.csv`
2. Check data: `summary`, `head`
3. Plot data: `plot`
4. Fit model: `glm`. Don't forget to specify `family`!

Recapitulating

1. Import data: `read.table` or `read.csv`
2. Check data: `summary`, `head`
3. Plot data: `plot`
4. Fit model: `glm`. Don't forget to specify `family`!
5. Examine models: `summary`

Recapitulating

1. Import data: `read.table` or `read.csv`
2. Check data: `summary`, `head`
3. Plot data: `plot`
4. Fit model: `glm`. Don't forget to specify `family`!
5. Examine models: `summary`
6. Use `allEffects` to back-transform parameters from logit into probability scale.

Recapitulating

1. Import data: `read.table` or `read.csv`
2. Check data: `summary`, `head`
3. Plot data: `plot`
4. Fit model: `glm`. Don't forget to specify `family`!
5. Examine models: `summary`
6. Use `allEffects` to back-transform parameters from logit into probability scale.
7. Plot model: `plot(allEffects(model))`. Or use `visreg`.

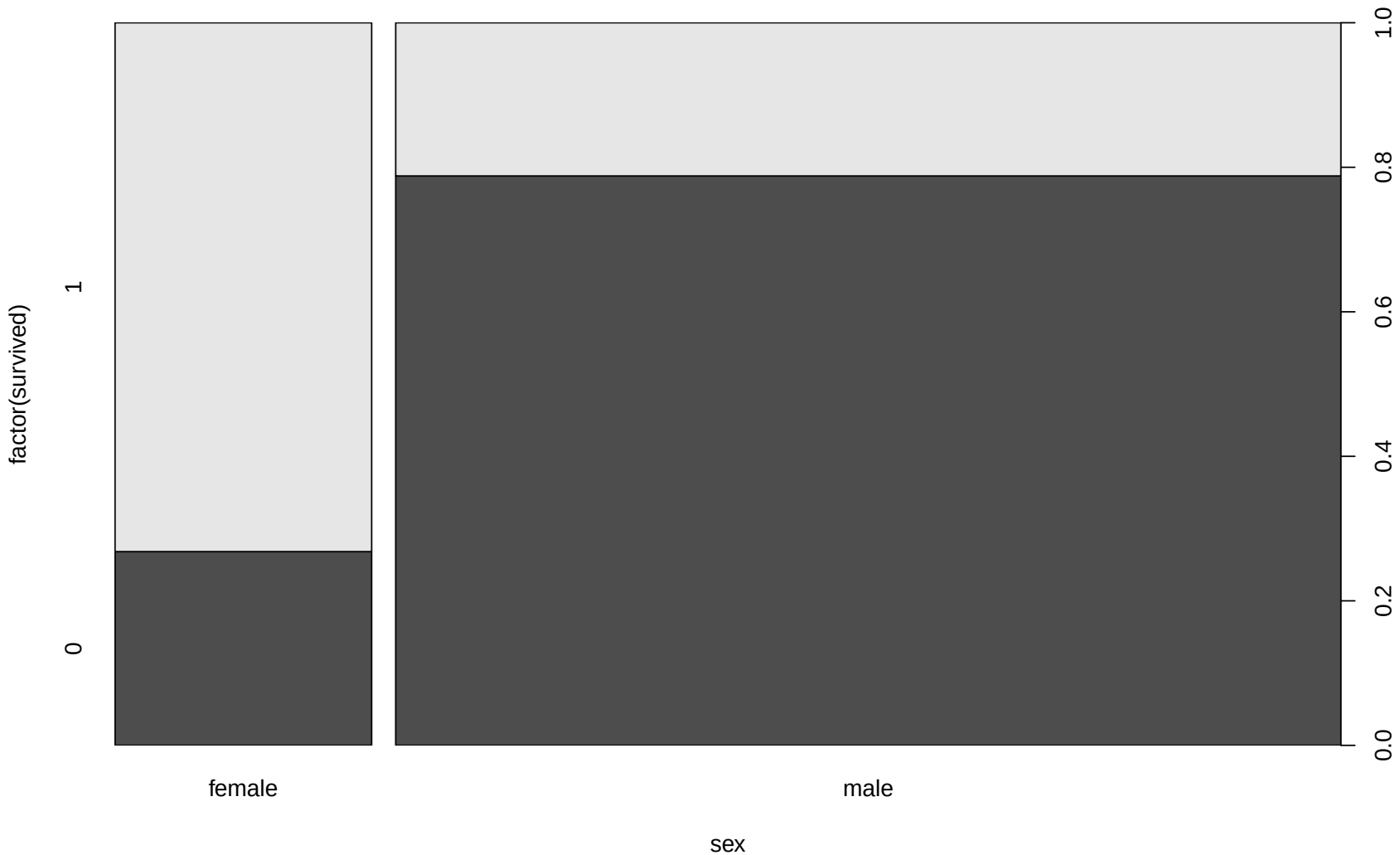
Recapitulating

1. Import data: `read.table` or `read.csv`
2. Check data: `summary`, `head`
3. Plot data: `plot`
4. Fit model: `glm`. Don't forget to specify `family`!
5. Examine models: `summary`
6. Use `allEffects` to back-transform parameters from logit into probability scale.
7. Plot model: `plot(allEffects(model))`. Or use `visreg`.
8. Examine residuals: `DHARMa::simulateResiduals`.

Q: Did men have higher survival than women?

Plot first

```
plot(factor(survived) ~ sex, data = titanic)
```

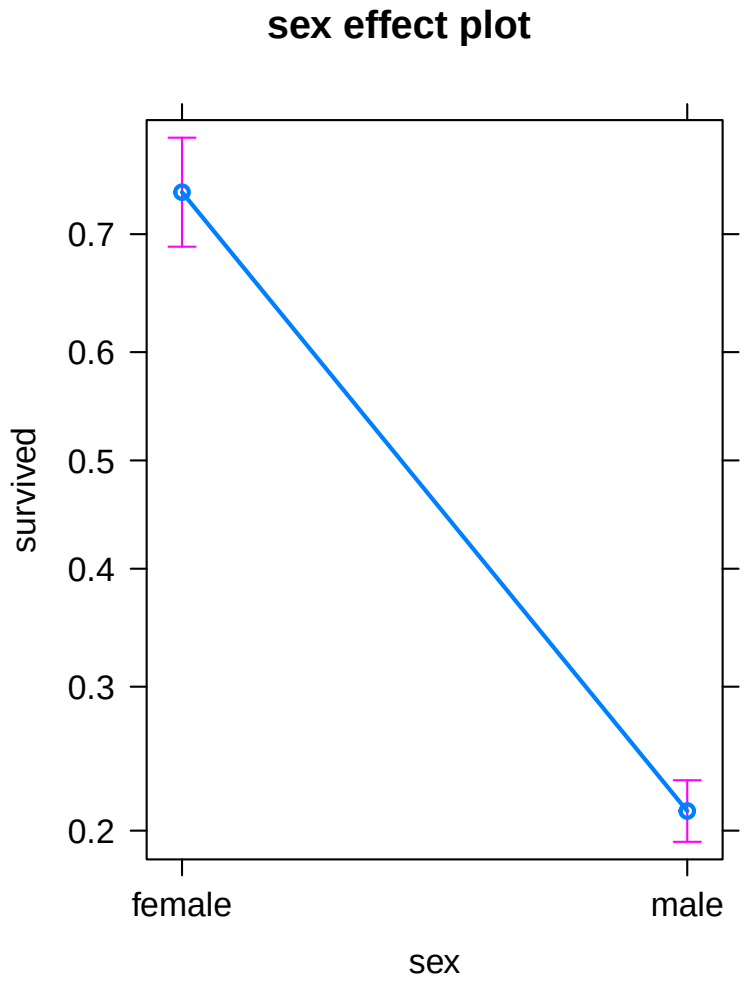


Fit model

```
Call:
glm(formula = survived ~ sex, family = binomial(link = "logit"),
    data = titanic)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.6226  -0.6903  -0.6903   0.7901   1.7613

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)   1.0044     0.1041   9.645   <2e-16 ***
sexmale      -2.3172     0.1196 -19.376   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 2769.5  on 2200  degrees of freedom
Residual deviance: 2335.0  on 2199  degrees of freedom
```

Effects

```
 model: survived ~ sex

 sex effect
sex
   female      male
0.7319149 0.2120162
```



Q: Did women have higher survival because they travelled more in first class?

Let's look at the data

```
table(titanic$class, titanic$survived, titanic$sex)
```

```
, ,  = female


          0   1
  crew    3  20
  first   4 141
  second 13  93
  third 106  90

, ,  = male


          0   1
  crew  670 192
  first 118  62
  second 154  25
  third 422  88
```

Mmmm...

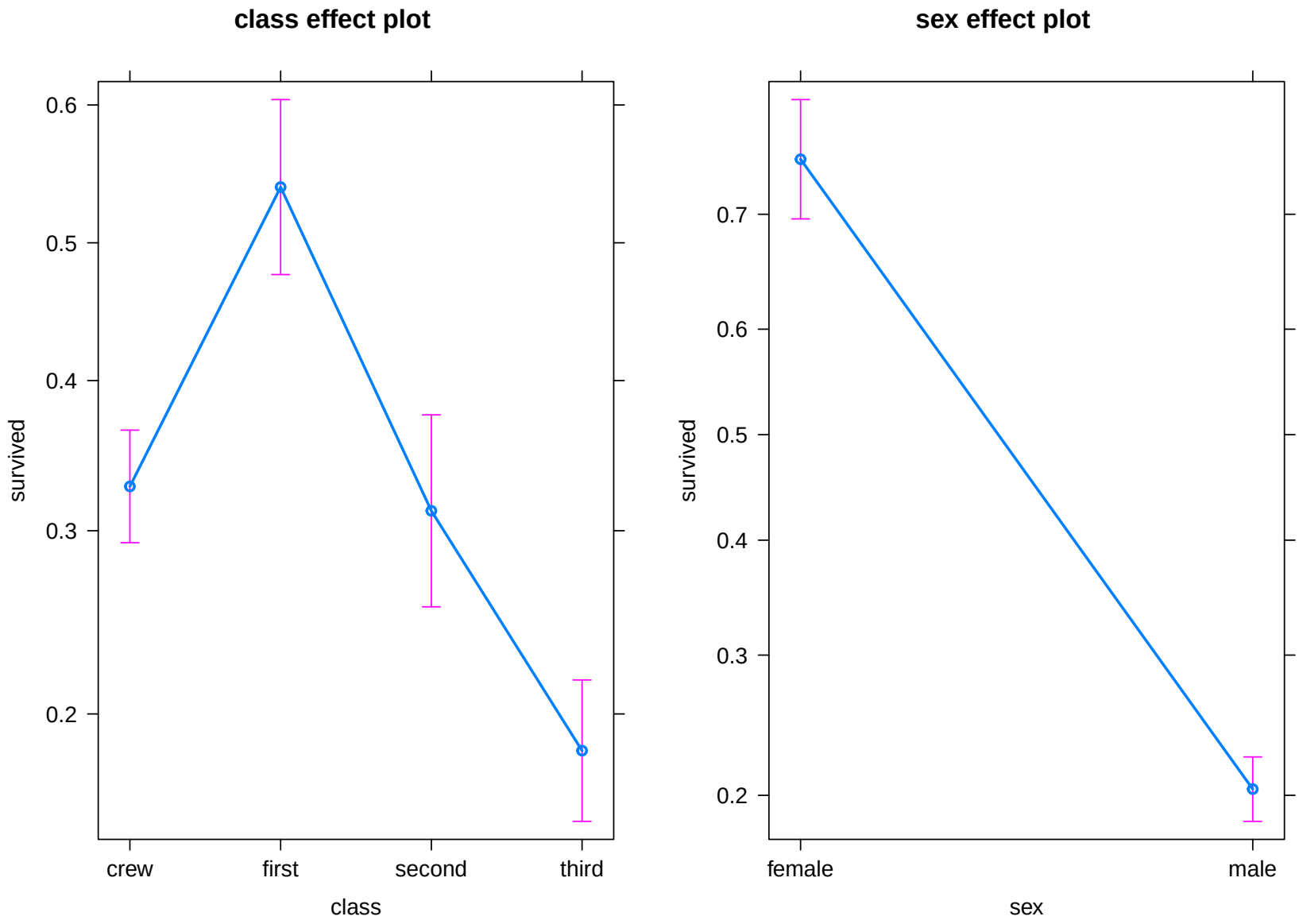
Fit additive model with both factors

```
tit.sex.class <- glm(survived ~ class + sex, data = titanic, fam
```

```
glm(formula = survived ~ class + sex, family = binomial, data =
             coef.est coef.se
(Intercept)  1.19     0.16
classfirst   0.88     0.16
classsecond -0.07     0.17
classthird  -0.78     0.14
sexmale     -2.42     0.14
---
  n = 2201, k = 5
  residual deviance = 2228.9, null deviance = 2769.5 (difference
```

Plot additive model

```
plot(allEffects(tit.sex.class))
```



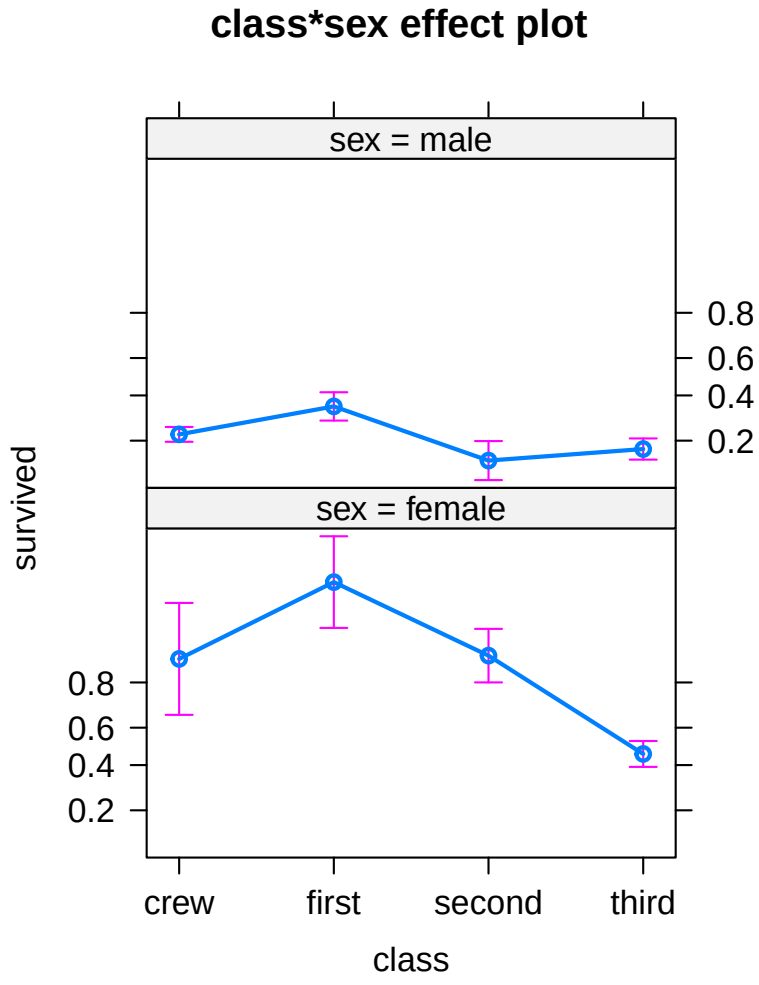
Fit model with both factors (interactions)

```
tit.sex.class <- glm(survived ~ class * sex, data = titanic, fam
```

```
glm(formula = survived ~ class * sex, family = binomial, data =
                    coef.est coef.se
(Intercept)          1.90     0.62
classfirst           1.67     0.80
classsecond          0.07     0.69
classthird          -2.06     0.64
sexmale             -3.15     0.62
classfirst:sexmale  -1.06     0.82
classsecond:sexmale -0.64     0.72
classthird:sexmale   1.74     0.65
---
  n = 2201, k = 8
  residual deviance = 2163.7, null deviance = 2769.5 (difference
```

Effects

```
model: survived ~ class * sex

class*sex effect
        sex
class         female      male
  crew    0.8695652 0.2227378
  first   0.9724138 0.3444444
  second  0.8773585 0.1396648
  third   0.4591837 0.1725490
```



So, women had higher probability of survival than men, even within the same class.

Logistic regression for proportion data

Read Titanic data in different format

Read `Titanic_prop.csv` data.

	X	Class	Sex	Age	No	Yes
1	1	1st	Female	Adult	4	140
2	2	1st	Female	Child	0	1
3	3	1st	Male	Adult	118	57
4	4	1st	Male	Child	0	5
5	5	2nd	Female	Adult	13	80
6	6	2nd	Female	Child	0	13

These are the same data, but summarized (see `Freq` variable).

Use cbind(n.success, n.failures) as response

```
prop.glm <- glm(cbind(Yes, No) ~ Class, data = tit.prop, family
```

```
Call:
glm(formula = cbind(Yes, No) ~ Class, family = binomial, data =

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-9.6404  -0.2915   1.5698   5.0366  10.1516

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)   0.5092     0.1146   4.445 8.79e-06 ***
Class2nd     -0.8565     0.1661  -5.157 2.51e-07 ***
Class3rd     -1.5965     0.1436 -11.114  < 2e-16 ***
ClassCrew    -1.6643     0.1390 -11.972  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)
```

Effects

```
 model: cbind(Yes, No) ~ Class

 Class effect
Class
      1st       2nd       3rd      Crew
0.6246154 0.4140351 0.2521246 0.2395480
```

Compare with former model based on raw data:

```
 model: survived ~ class

 class effect
class
     crew     first    second     third
0.2395480 0.6246154 0.4140351 0.2521246
```

Same results!

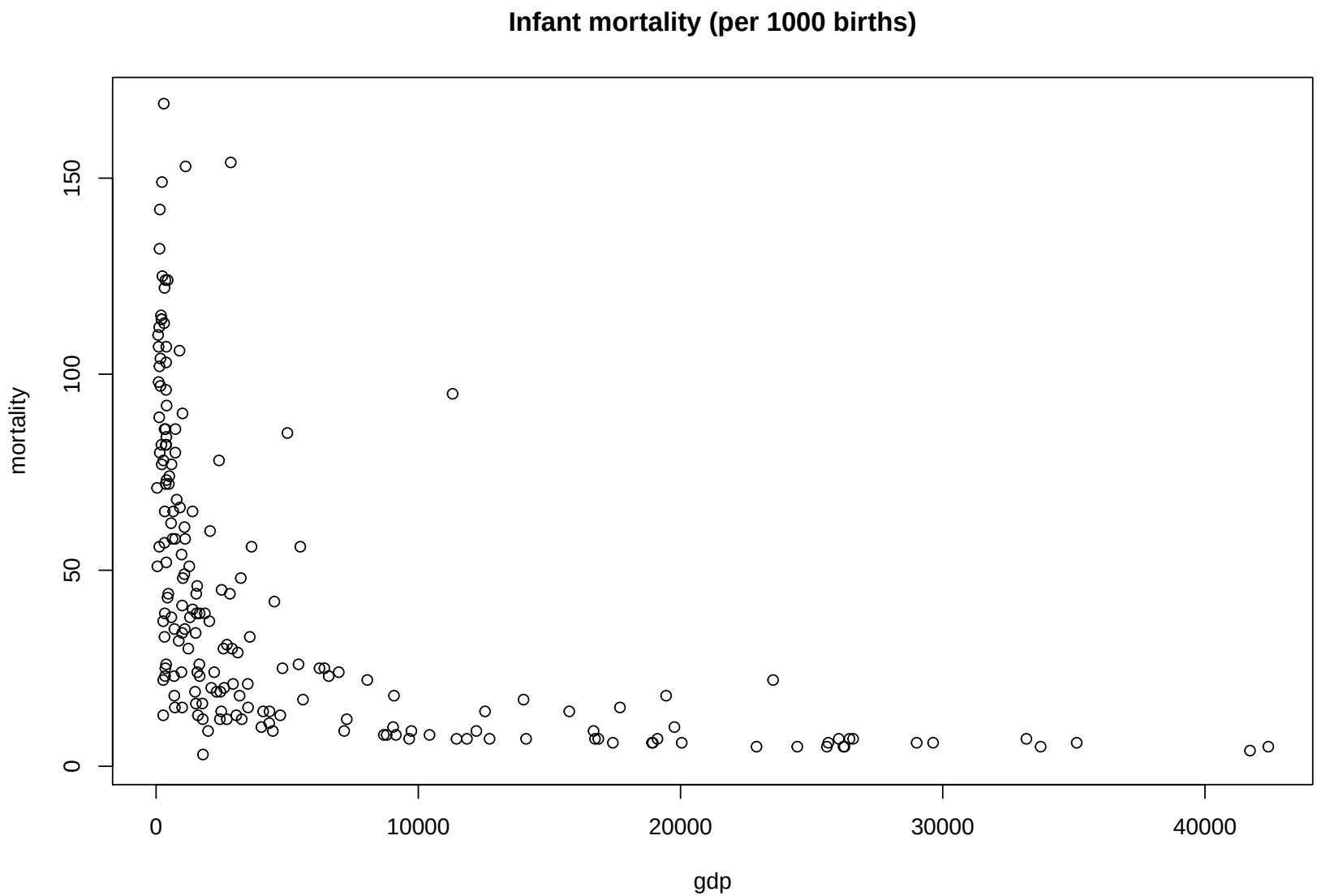
Logistic regression with continuous predictors

Example dataset: GDP and infant mortality
Read `UN_GDP_infantmortality.csv`.

```
          country       mortality           gdp
 Afghanistan   :  1   Min.   :  2.00   Min.   :   36
 Albania       :  1   1st Qu.: 12.00   1st Qu.:  442
 Algeria       :  1   Median : 30.00   Median : 1779
 American.Samoa:  1   Mean   : 43.48   Mean   : 6262
 Andorra       :  1   3rd Qu.: 66.00   3rd Qu.: 7272
 Angola        :  1   Max.   :169.00   Max.   :42416
 (Other)       :201   NA's   :6        NA's   :10
```

EDA

```
plot(mortality ~ gdp, data = gdp, main = "Infant mortality (per
```



Fit model

```
gdp.glm <- glm(cbind(mortality, 1000 - mortality) ~ gdp,
               data = gdp, family = binomial(link = "logit"))
```

```
Call:
glm(formula = cbind(mortality, 1000 - mortality) ~ gdp, family =
    data = gdp)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-9.2230  -3.5163  -0.5697   2.4284  13.5849

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -2.657e+00  1.311e-02 -202.76   <2e-16 ***
gdp         -1.279e-04  3.458e-06  -36.98   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)
```

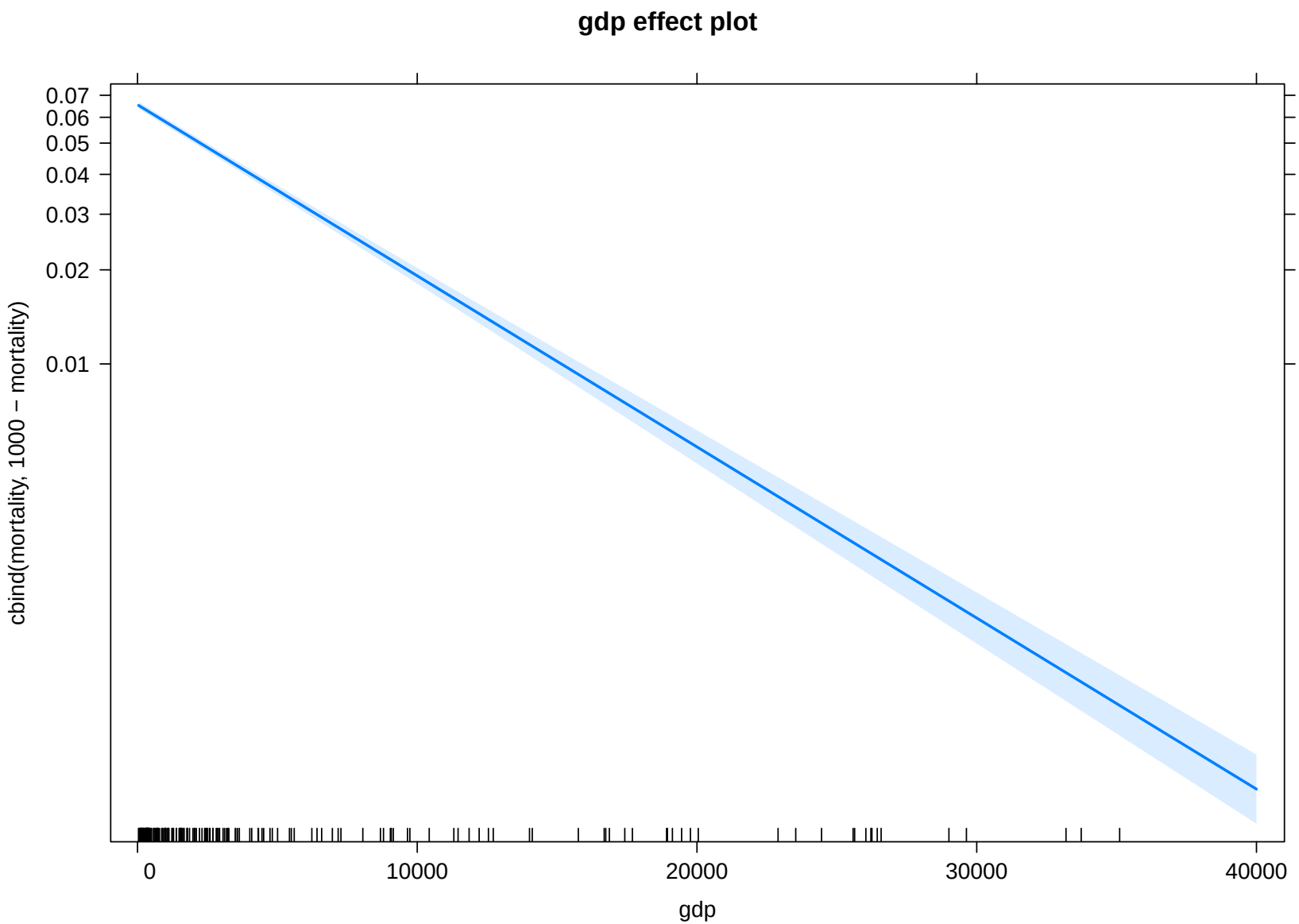
Effects

```
allEffects(gdp.glm)
```

```
 model: cbind(mortality, 1000 - mortality) ~ gdp

 gdp effect
gdp
          40         10000         20000         30000         40000
0.0652177296 0.0191438829 0.0054028095 0.0015096074 0.0004206154
```

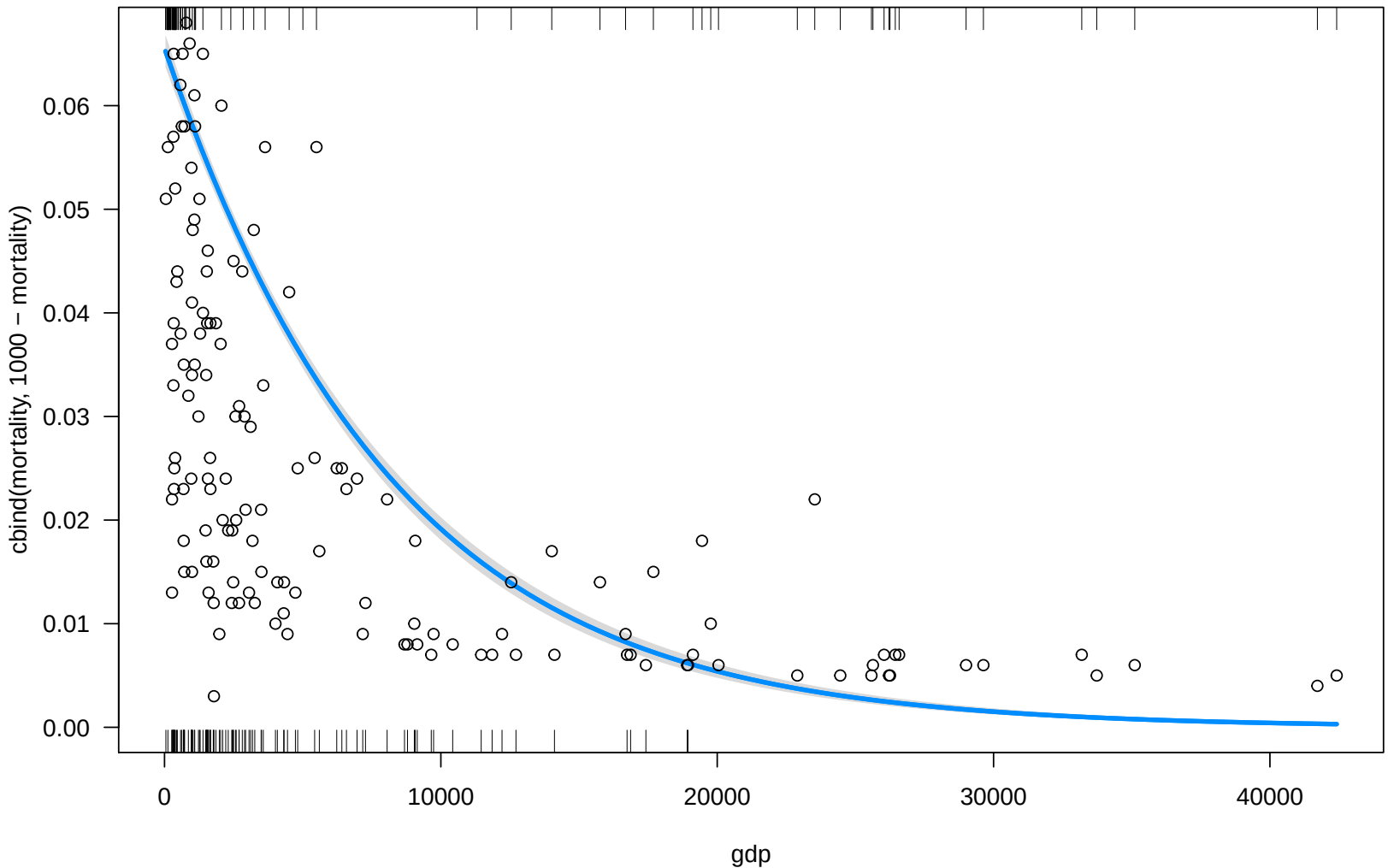
Effects plot

```
plot(allEffects(gdp.glm))
```



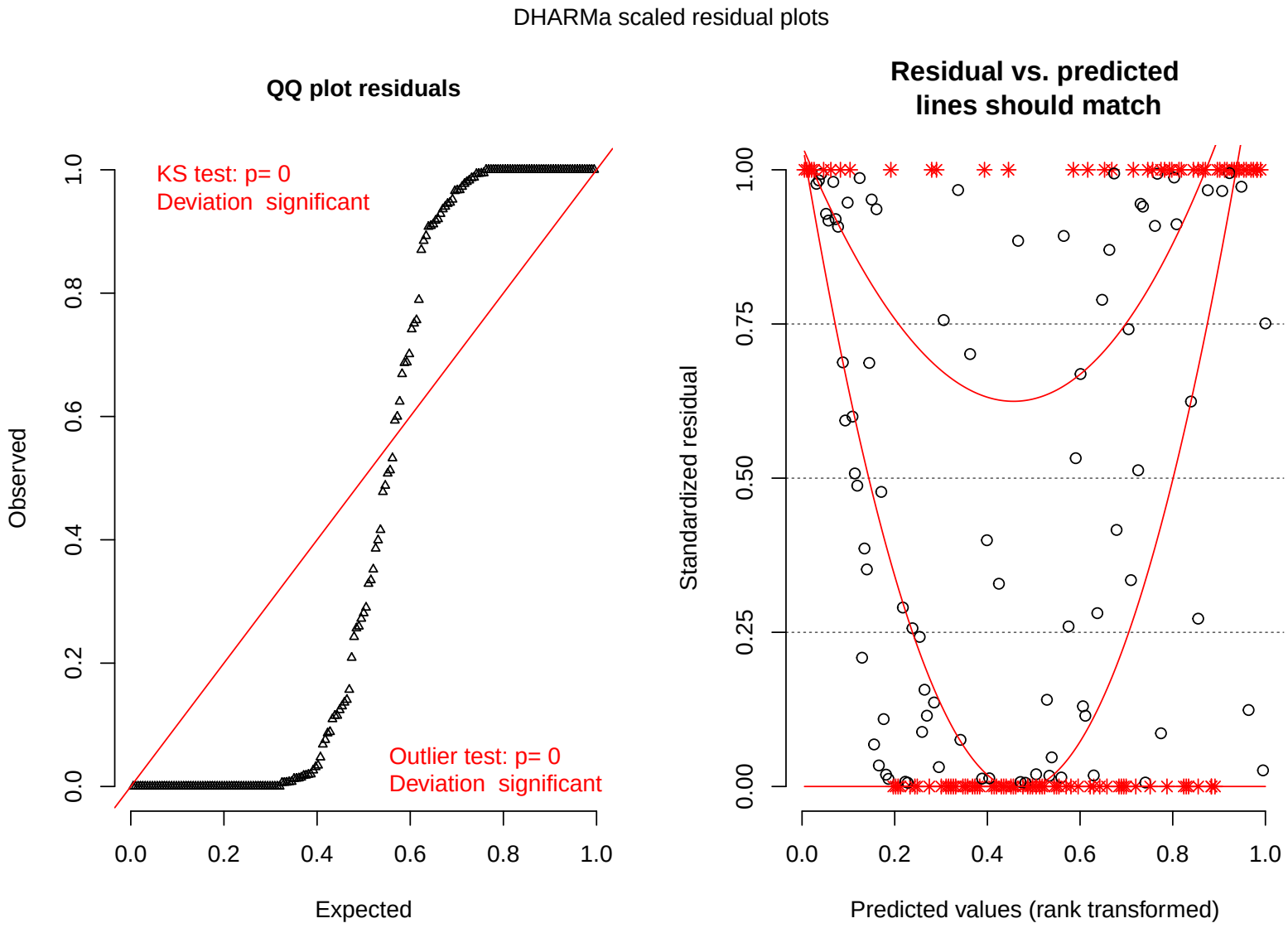
Plot model using visreg:

```
visreg(gdp.glm, scale = "response")
points(mortality/1000 ~ gdp, data = gdp)
```



Residuals diagnostics with DHARMa

```
simulateResiduals(gdp.glm, plot = TRUE)
```



Overdispersion

Testing for overdispersion (DHARMa)

```
simres <- simulateResiduals(gdp.glm, refit = TRUE)
testDispersion(simres, plot = FALSE)
```

```
    DHARMa nonparametric dispersion test via mean deviance resid
    fitted vs. simulated-refitted

data:  simres
dispersion = 21, p-value < 2.2e-16
alternative hypothesis: two.sided
```

Overdispersion in logistic regression with proportion data

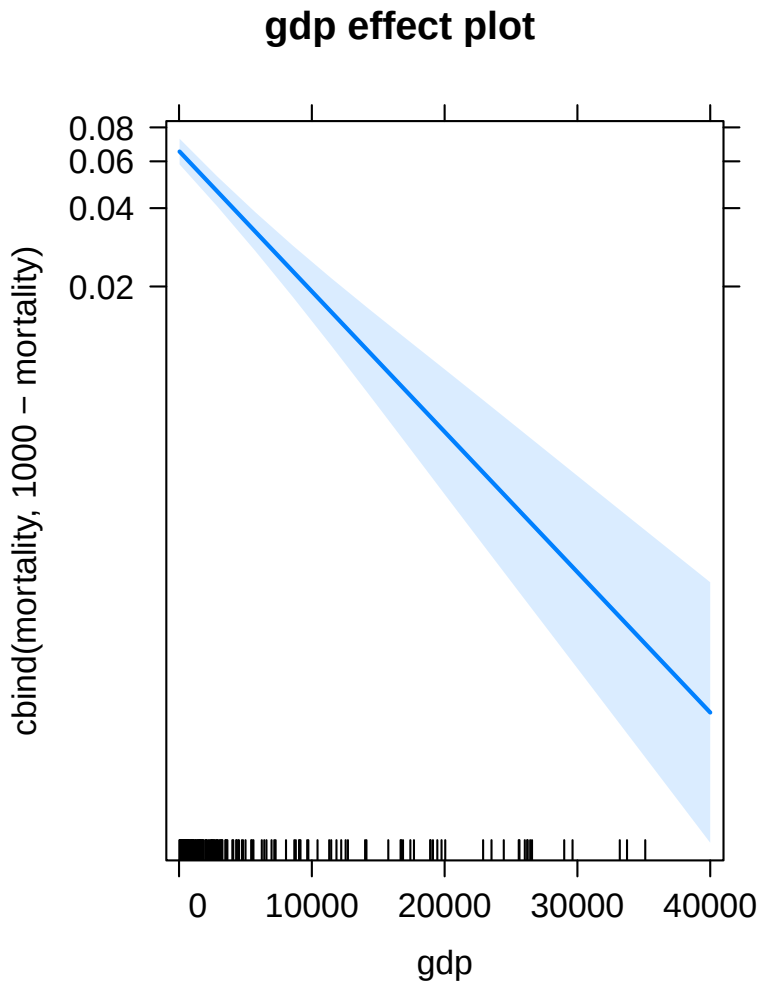
```
gdp.overdisp <- glm(cbind(mortality, 1000 - mortality) ~ gdp,
                    data = gdp, family = quasibinomial)
```

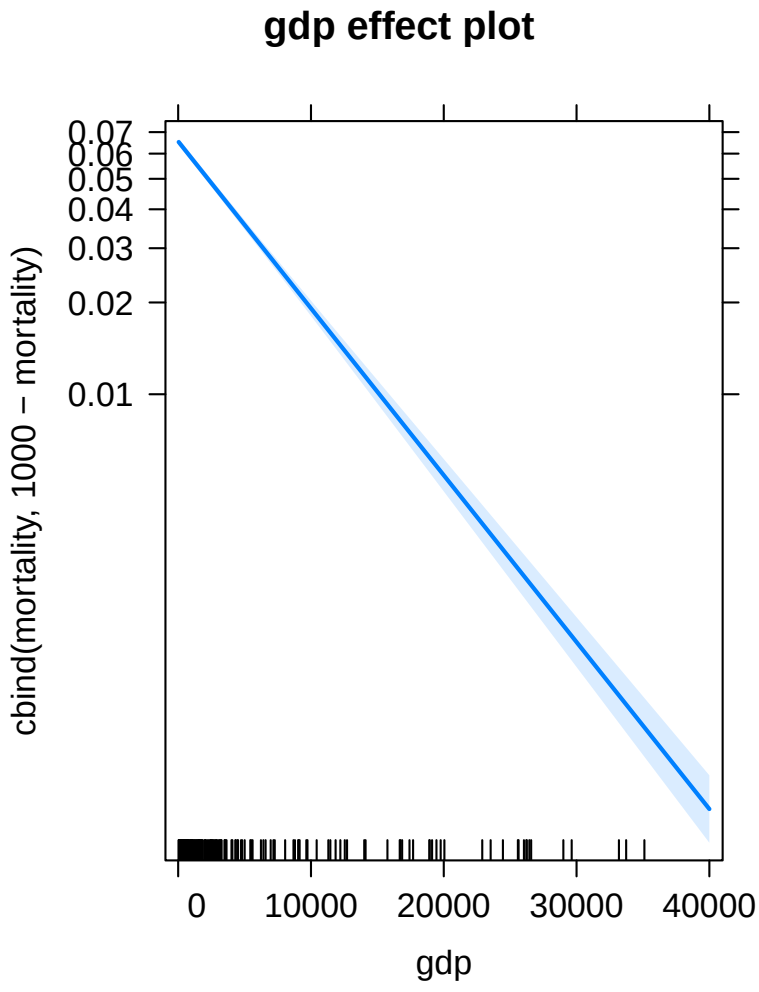
```
Call:
glm(formula = cbind(mortality, 1000 - mortality) ~ gdp, family =
    data = gdp)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-9.2230  -3.5163  -0.5697   2.4284  13.5849

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -2.657e+00  5.977e-02 -44.465  < 2e-16 ***
gdp         -1.279e-04  1.577e-05  -8.111 5.96e-14 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for quasibinomial family taken to be 20.79
```

Mean estimates do not change after accounting for overdispersion

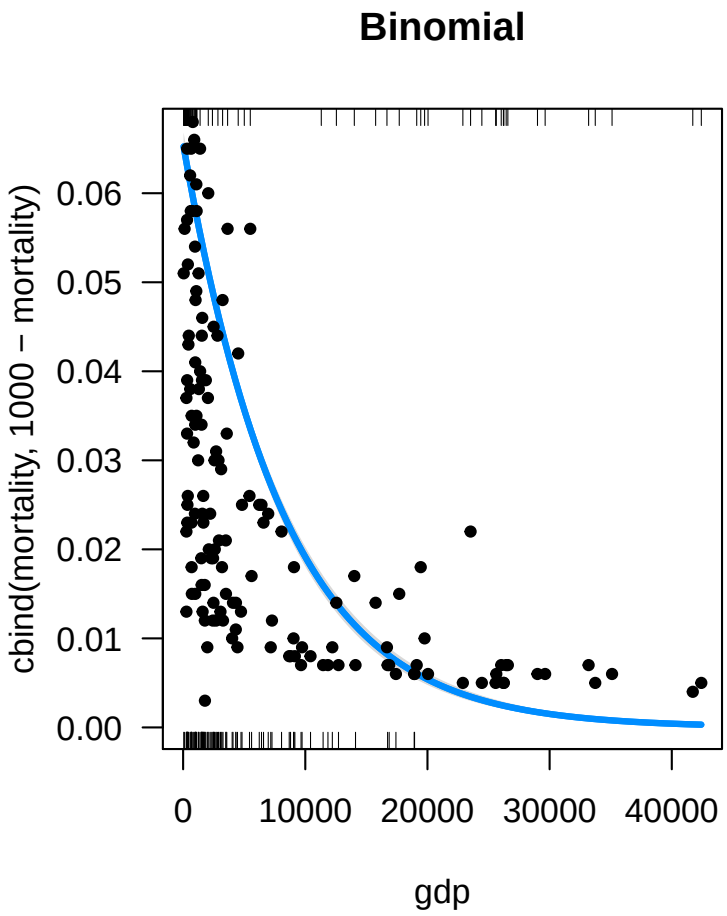
```
 model: cbind(mortality, 1000 - mortality) ~ gdp

 gdp effect
gdp
          40         10000         20000         30000         40000
0.0652177296 0.0191438829 0.0054028095 0.0015096074 0.0004206154

 model: cbind(mortality, 1000 - mortality) ~ gdp

 gdp effect
gdp
          40         10000         20000         30000         40000
0.0652177296 0.0191438829 0.0054028095 0.0015096074 0.0004206154
```

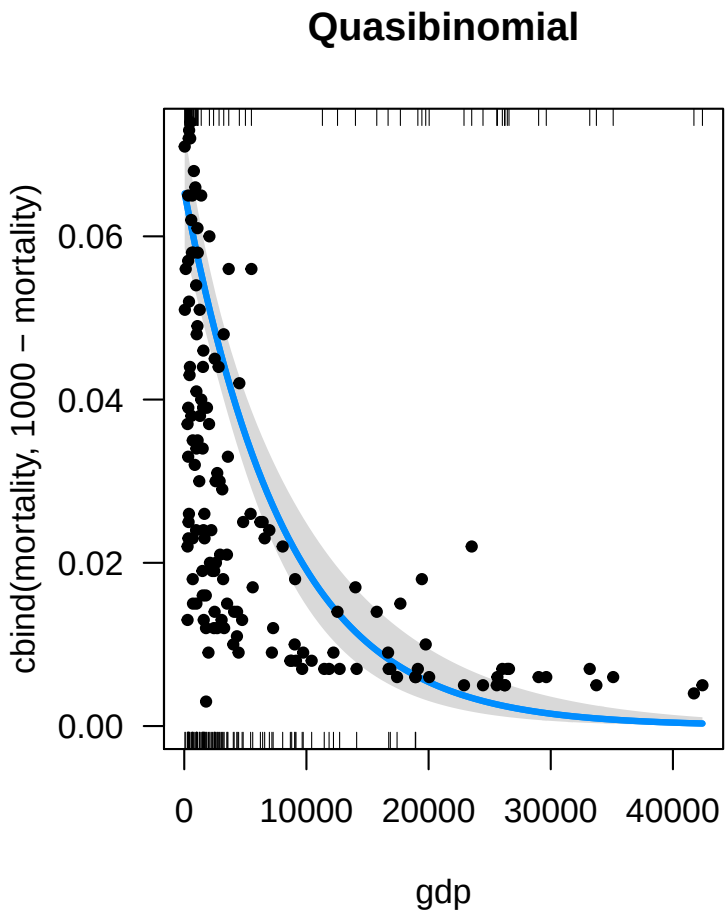
But standard errors (uncertainty) do!





Plot model and data





Overdispersion

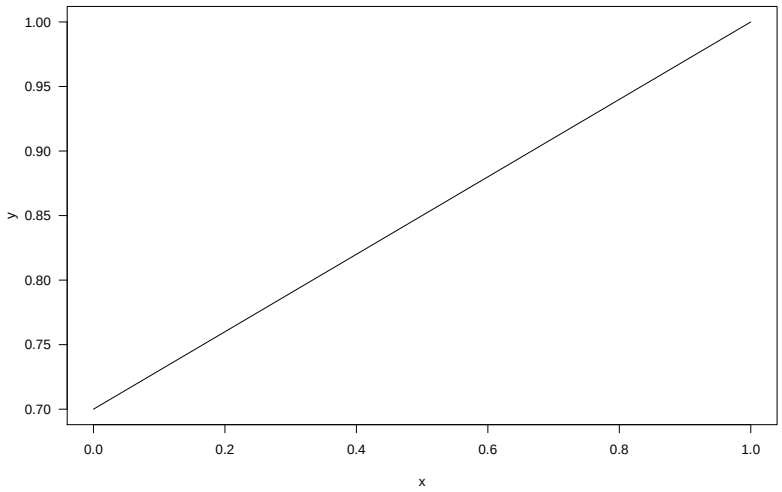
Whenever you fit logistic regression to **proportion** data, check family `quasibinomial`.

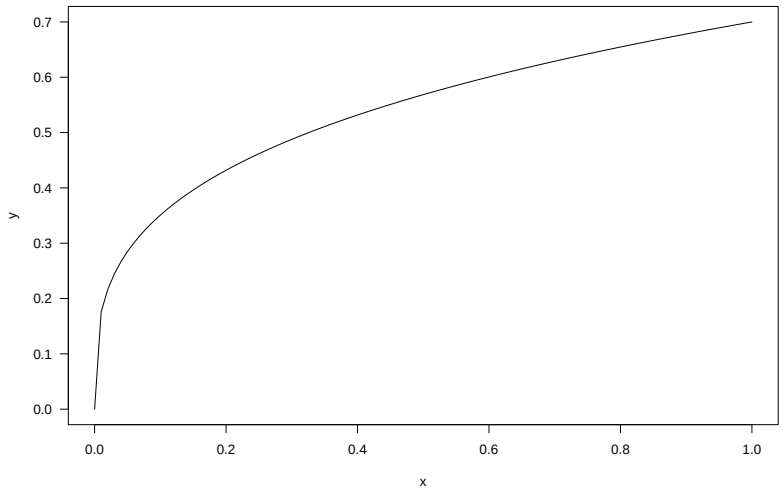
Think about the shape of relationships

$y \sim x + z$

Really? Not everything has to be linear! Actually, it often is not.

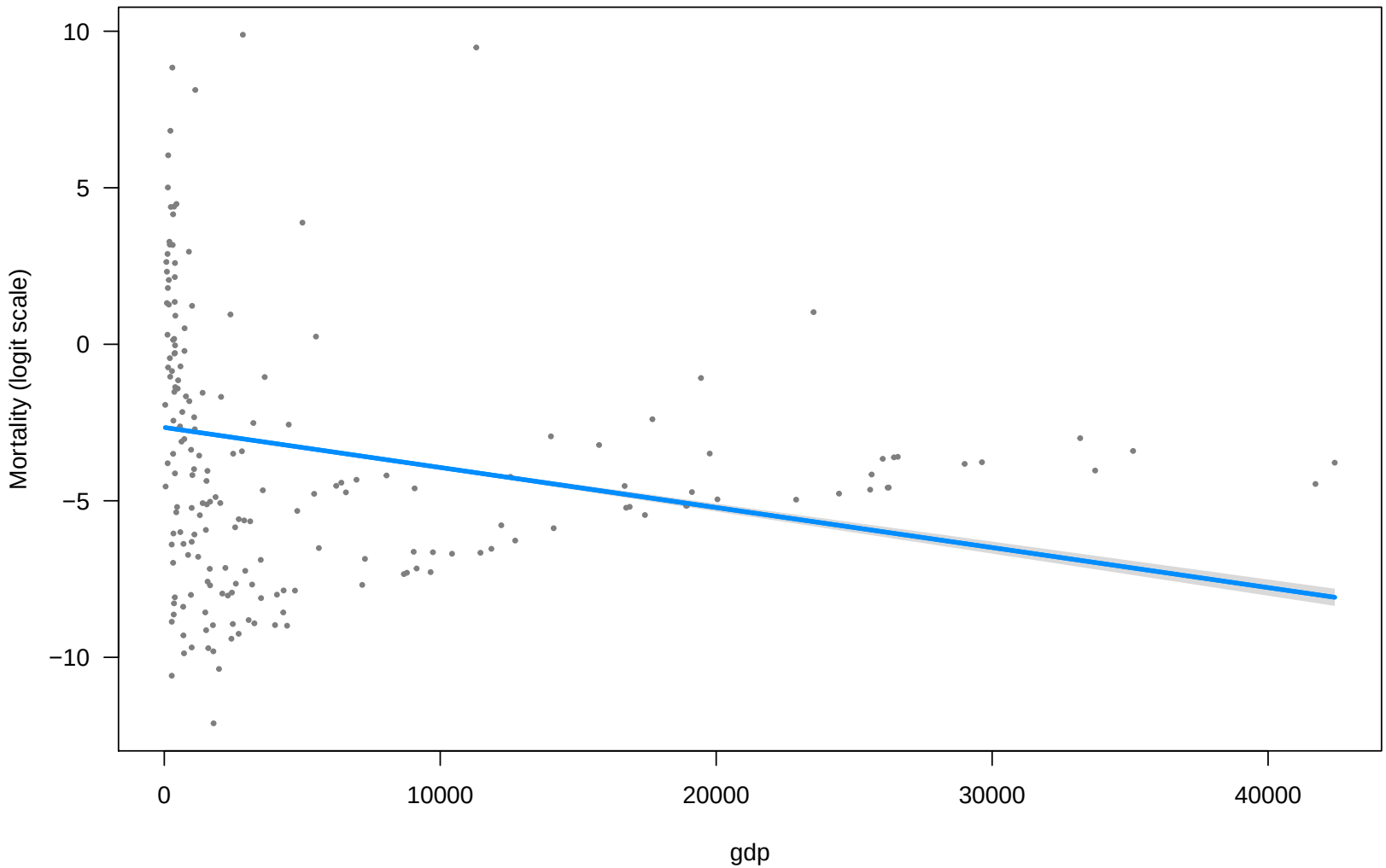
Think about shape of relationship. See chapter 3 in Bolker's book.



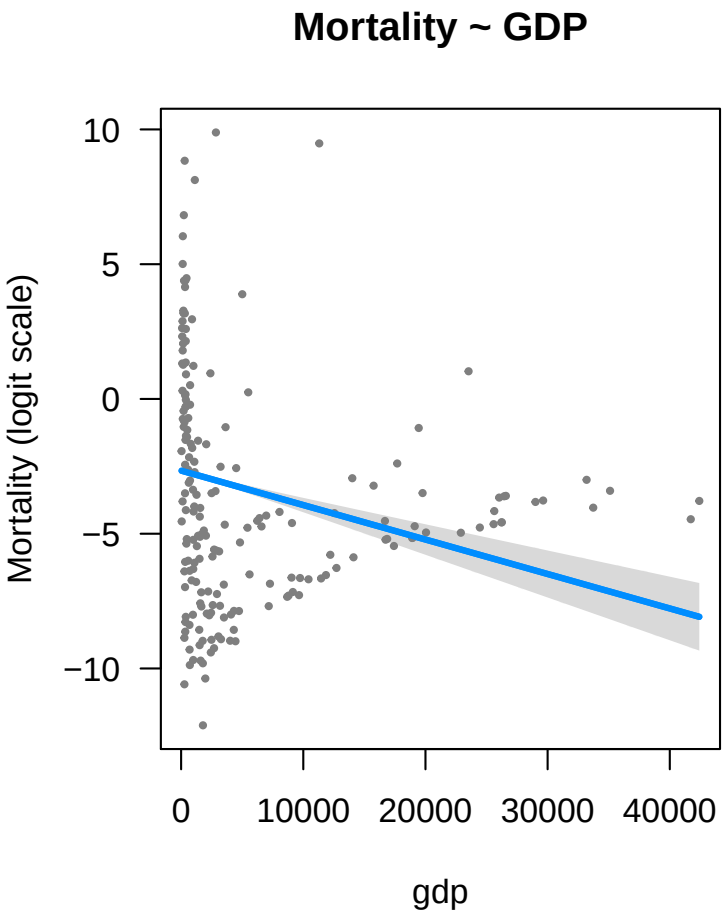


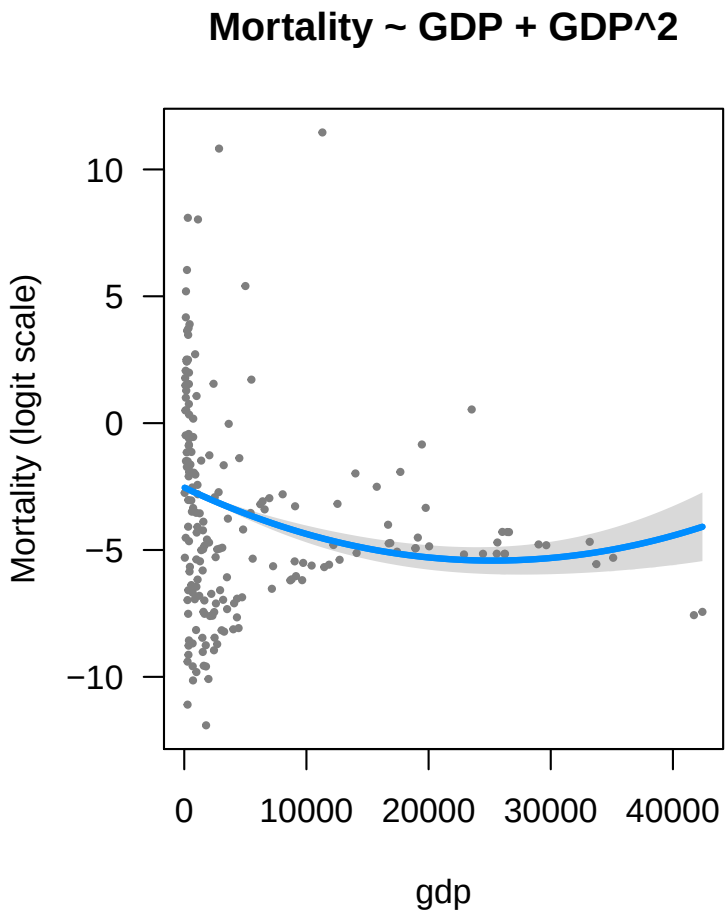
Think about the shape of relationships

```
visreg(gdp.glm, ylab = "Mortality (logit scale)")
```

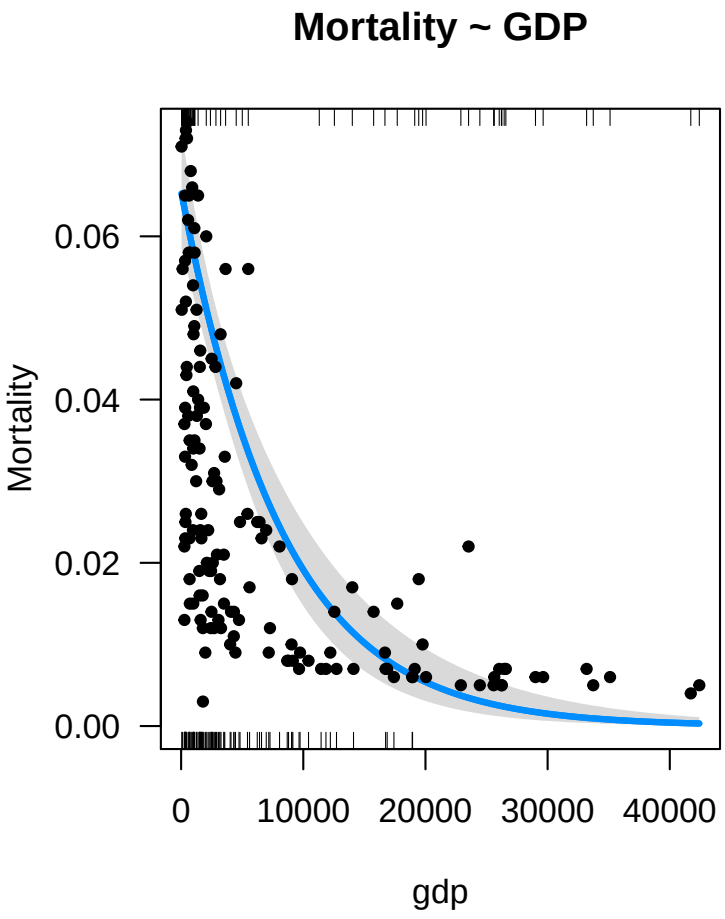


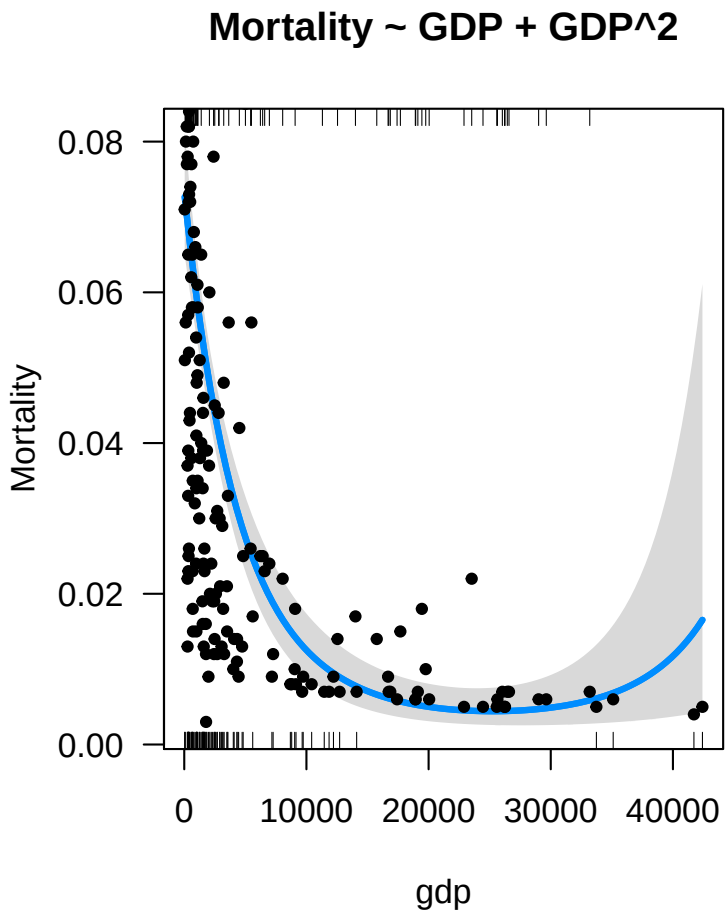
Think about the shape of relationships



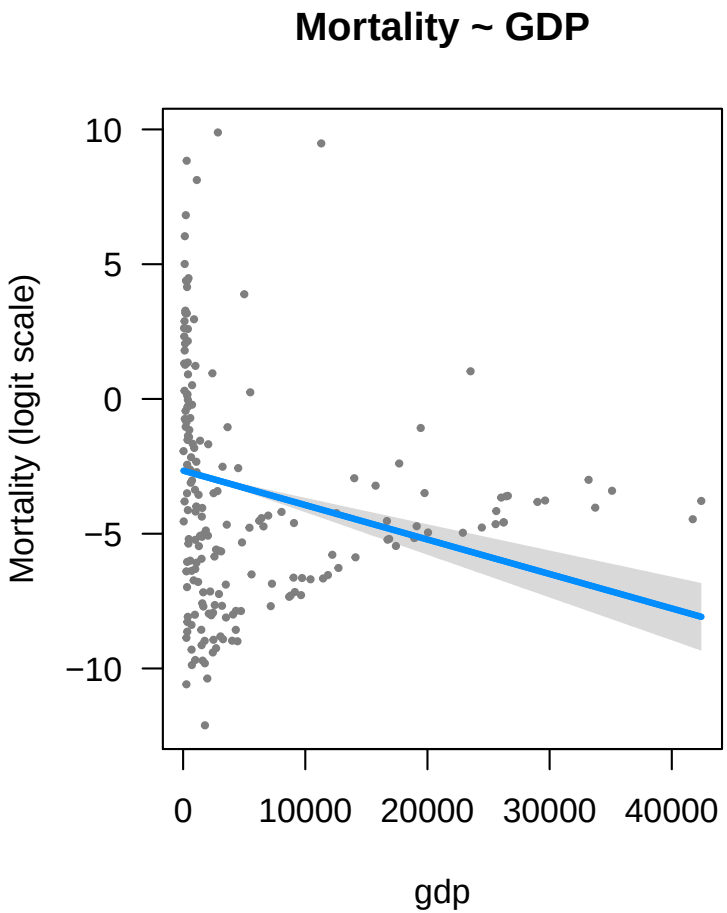


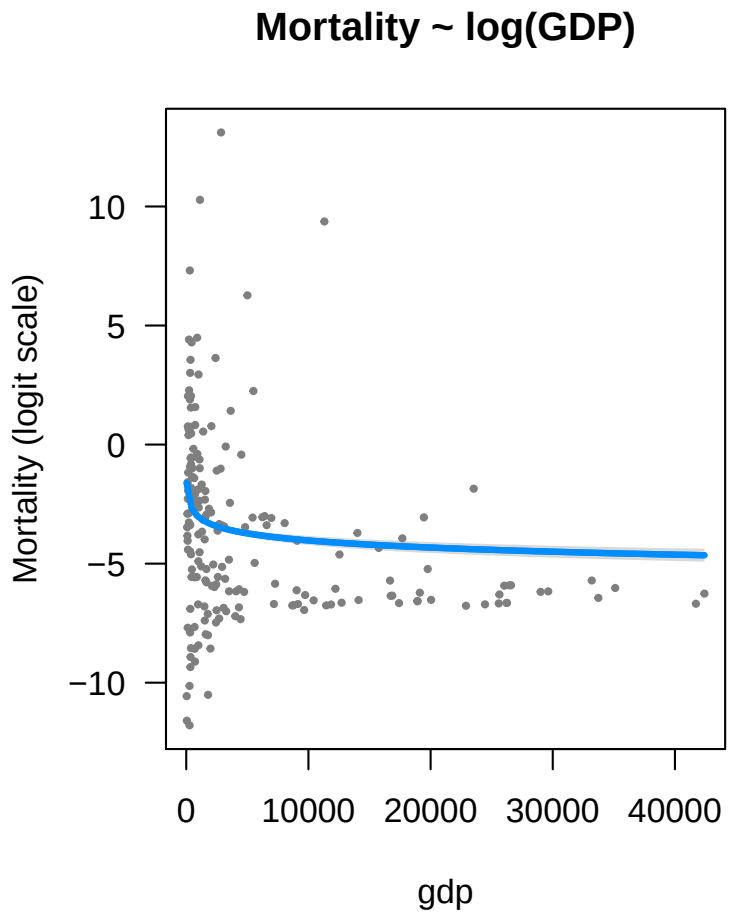
Think about the shape of relationships



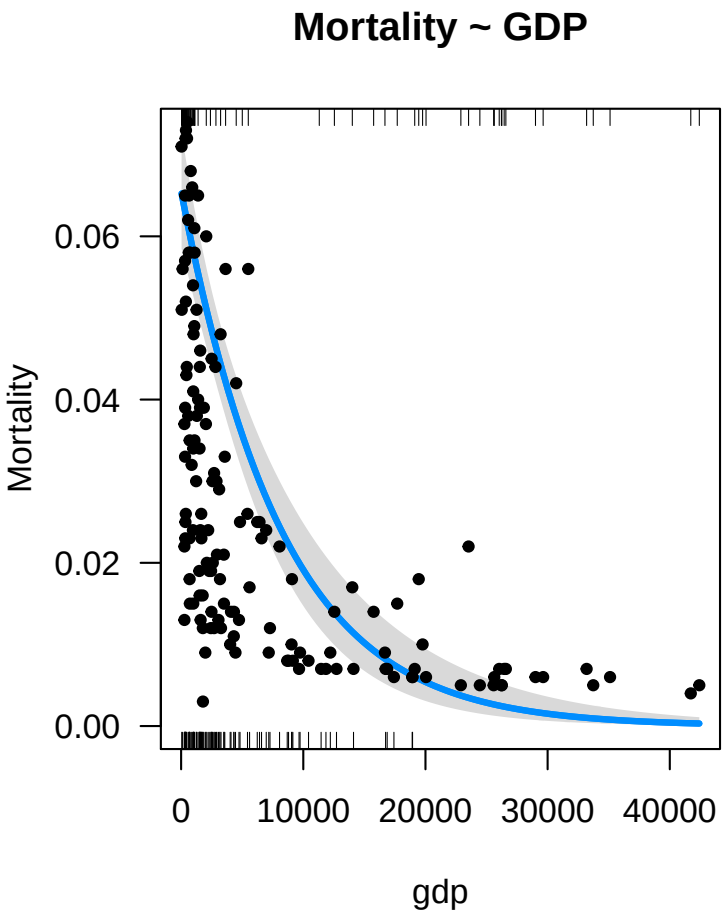


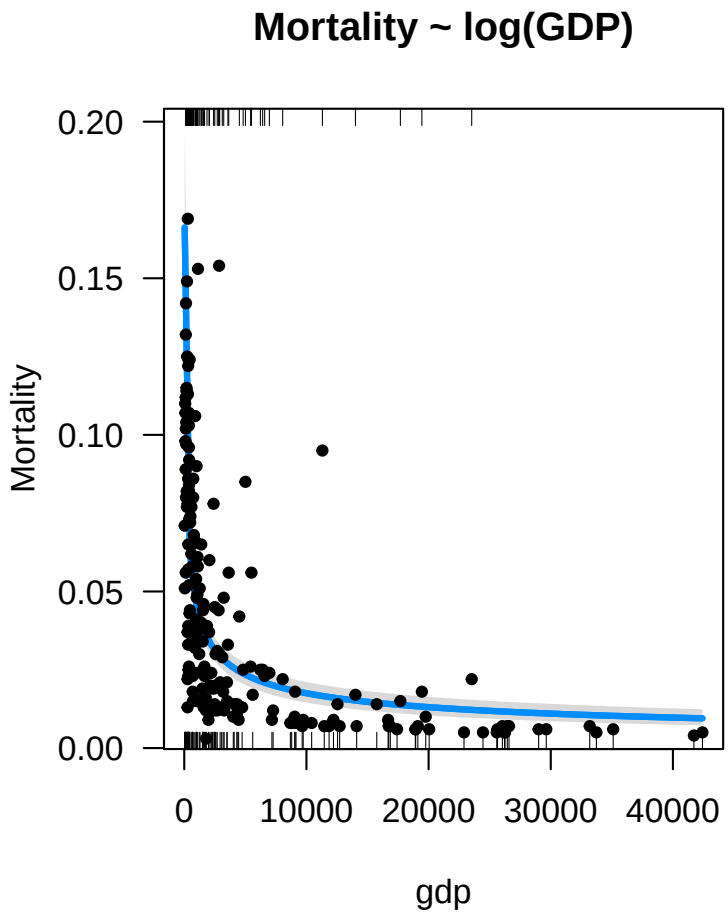
Think about the shape of relationships





Think about the shape of relationships





GLM for count data: Poisson regression

Types of response variable

- Gaussian: `lm`

Types of response variable

- Gaussian: `lm`
- Bernouilli / Binomial: `glm` (family `binomial / quasibinomial`)

Types of response variable

- Gaussian: `lm`
- Bernouilli / Binomial: `glm` (family `binomial / quasibinomial`)
- Counts: `glm` (family `poisson / quasipoisson`)

Poisson regression

- Response variable: Counts (0, 1, 2, 3...) - discrete

Then

$$log(N) = a + bx$$
$$N = e^{a+bx}$$

Poisson regression

- Response variable: Counts (0, 1, 2, 3...) - discrete
- Link function: `log`

Then

$$log(N) = a + bx$$
$$N = e^{a+bx}$$

Example dataset: Seedling counts in quadrats

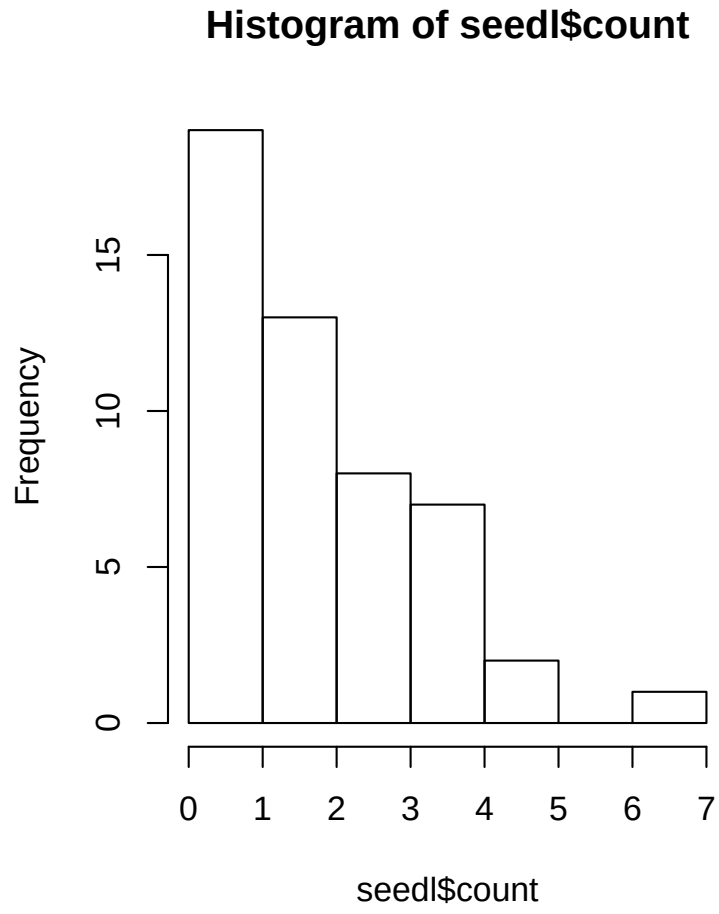
```
seedl <- read.csv("data-raw/seedlings.csv")
```

```
       X                count              row              col       
 Min.   : 1.00   Min.   :0.00   Min.   :1   Min.   : 1.0  
 1st Qu.:13.25   1st Qu.:1.00   1st Qu.:2   1st Qu.: 3.0  
 Median :25.50   Median :2.00   Median :3   Median : 5.5  
 Mean   :25.50   Mean   :2.14   Mean   :3   Mean   : 5.5  
 3rd Qu.:37.75   3rd Qu.:3.00   3rd Qu.:4   3rd Qu.: 8.0  
 Max.   :50.00   Max.   :7.00   Max.   :5   Max.   :10.0  
     light             area       
 Min.   : 2.571   Min.   :0.25  
 1st Qu.:26.879   1st Qu.:0.25  
 Median :47.493   Median :0.50  
 Mean   :47.959   Mean   :0.62  
 3rd Qu.:67.522   3rd Qu.:1.00  
 Max.   :99.135   Max.   :1.00  
```

EDA

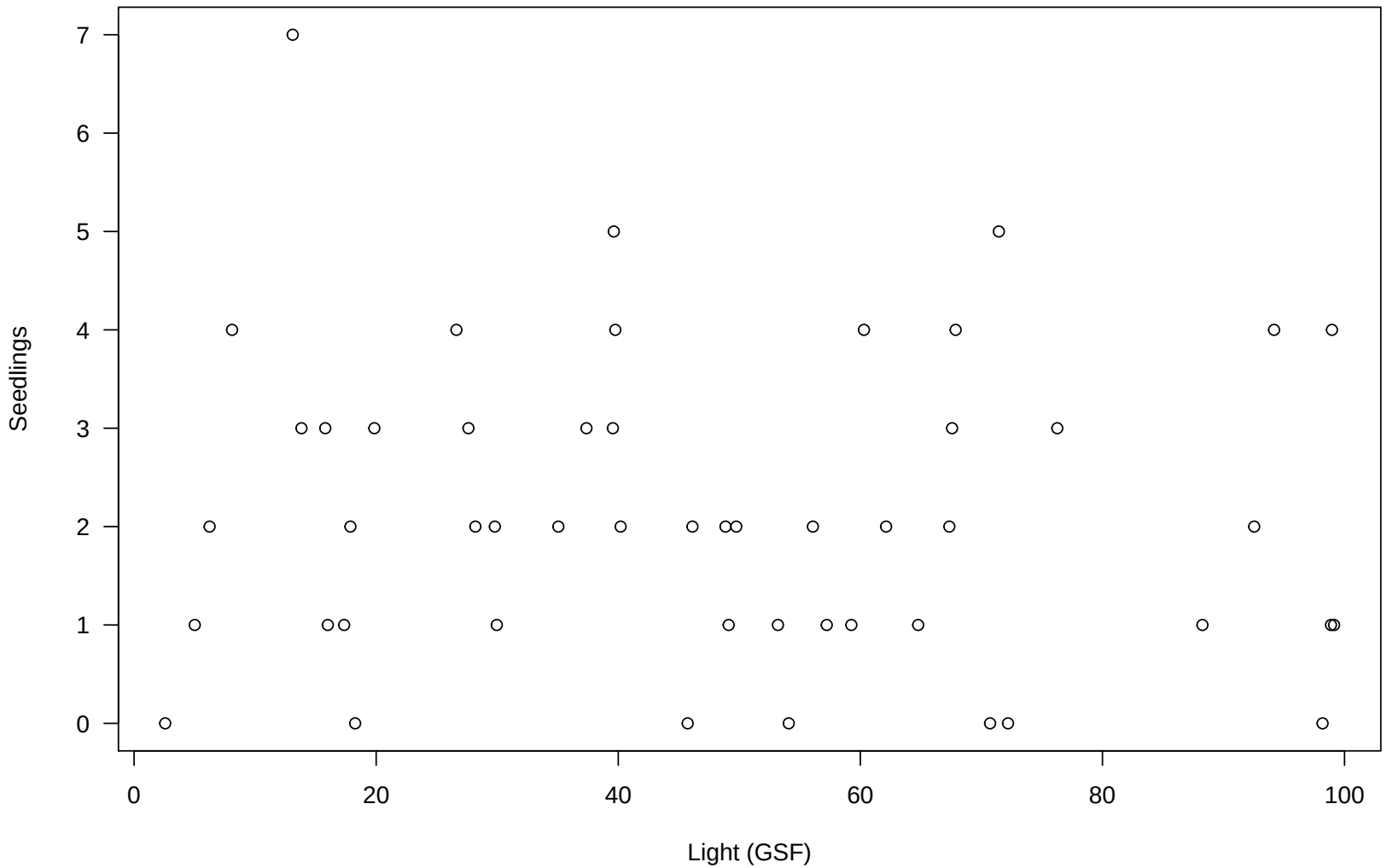
```
table(seedl$count)
```

```
 0  1  2  3  4  5  7
 7 12 13  8  7  2  1
```



Q: Relationship between Nseedlings and light?

```
plot(seedl$light, seedl$count, las = 1, xlab = "Light (GSF)", yl
```



Let's fit model (Poisson regression)

```r
seedl.glm <- glm(count ~ light, data = seedl, family = poisson(link="log"))
summary(seedl.glm)
```

```
Call:
glm(formula = count ~ light, family = poisson(link = "log"),
    data = seedl)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.1906  -0.8466  -0.1110   0.5220   2.4577

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  0.881805   0.188892   4.668 3.04e-06 ***
light       -0.002576   0.003528  -0.730    0.465
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 63.029  on 49  degrees of freedom
Residual deviance: 62.492  on 48  degrees of freedom
AIC: 182.03

Number of Fisher Scoring iterations: 5
```

Interpreting Poisson regression output

Parameter estimates (log scale):

```
coef(seedl.glm)
```

```
 (Intercept)        light
 0.881805022 -0.002575656
```

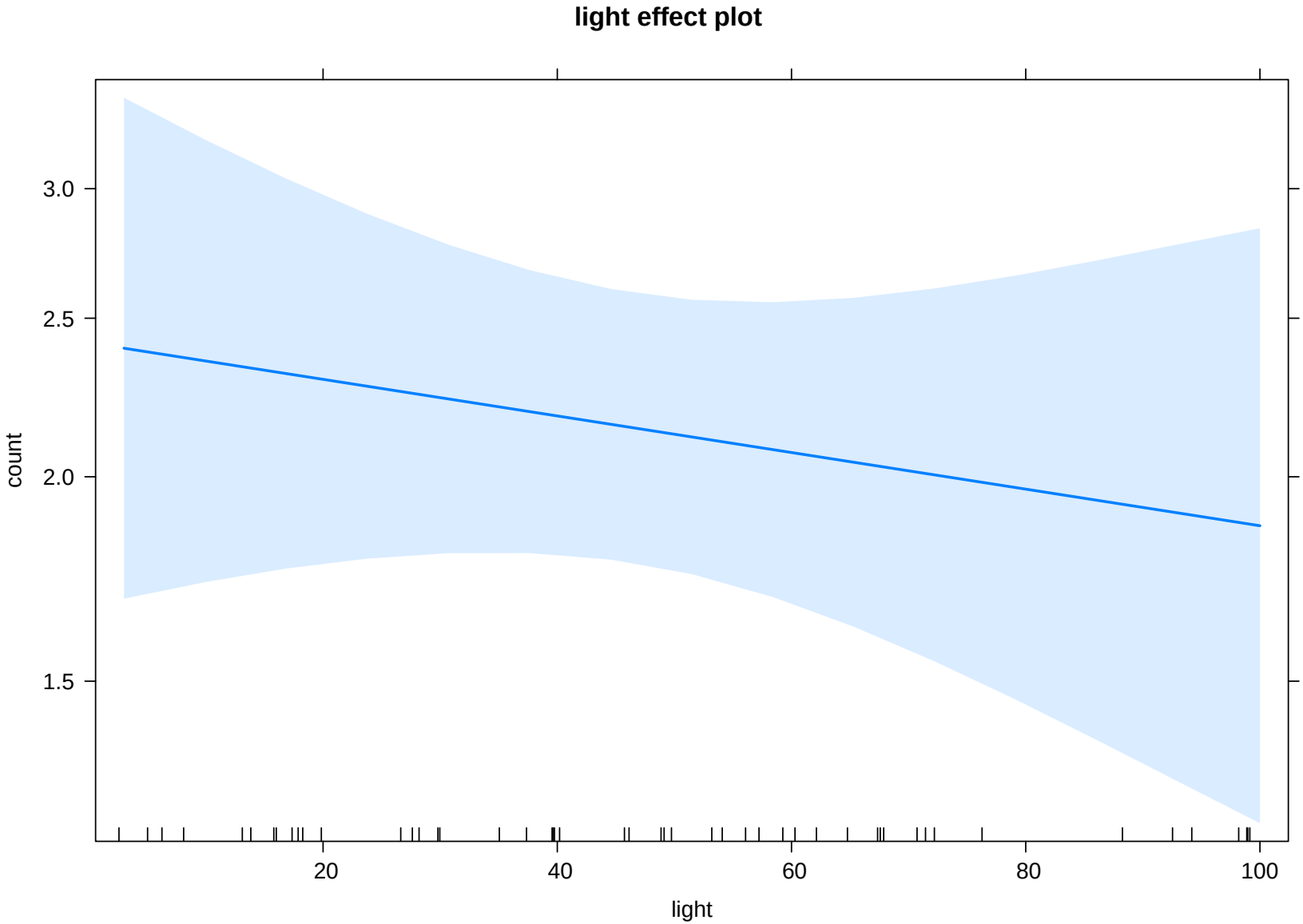
We need to back-transform: apply the inverse of the logarithm

```
exp(coef(seedl.glm))
```

```
(Intercept)       light
  2.4152554   0.9974277
```

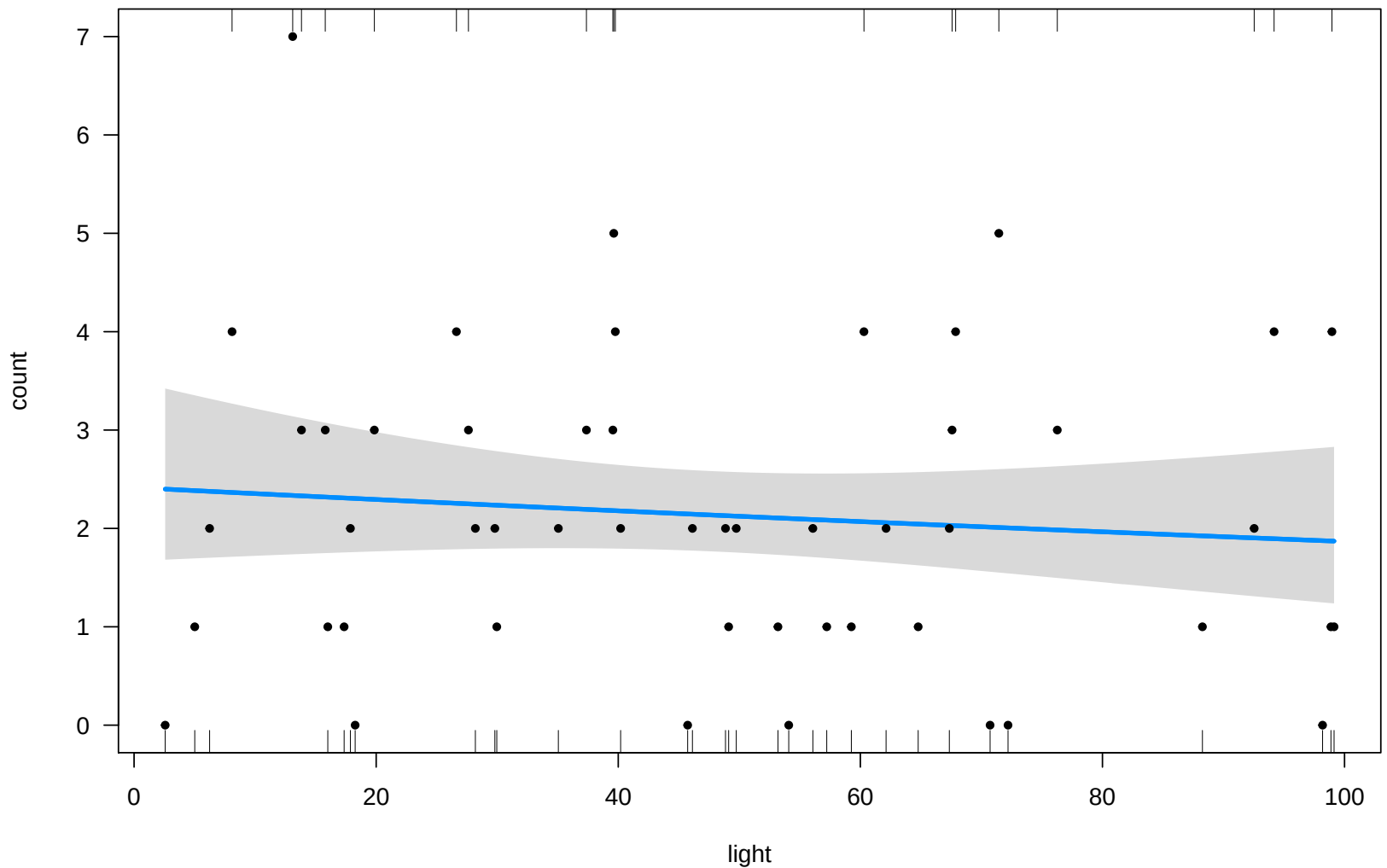
So what's the relationship between Nseedlings and light?

```
plot(allEffects(seedl.glm))
```

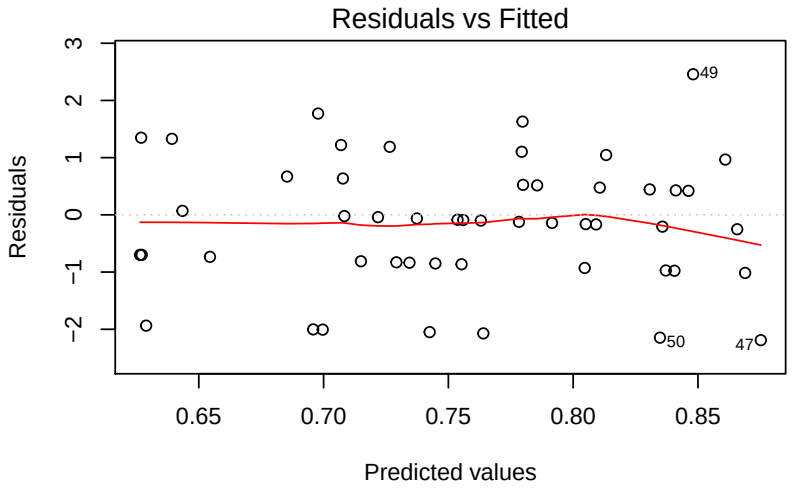


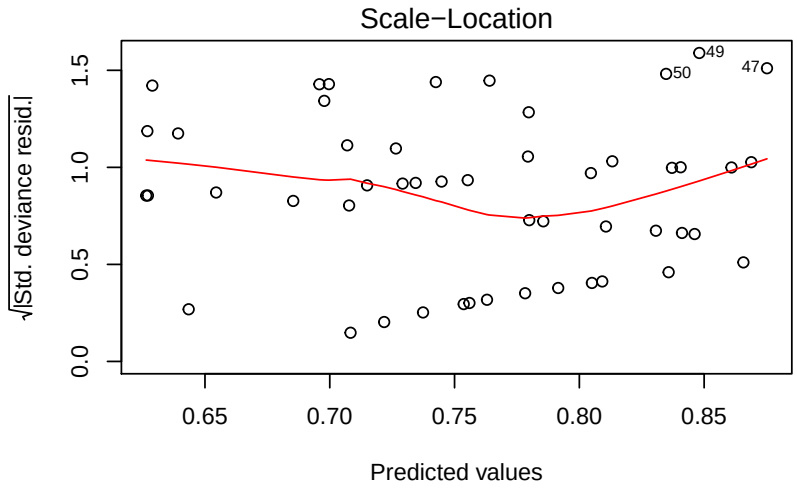
Using visreg

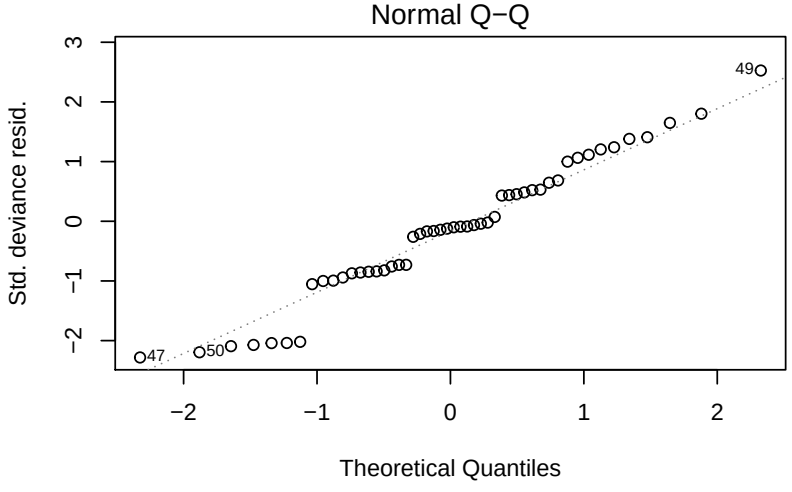
```
visreg(seedl.glm, scale = "response", ylim = c(0, 7))
points(count ~ light, data = seedl, pch = 20)
```

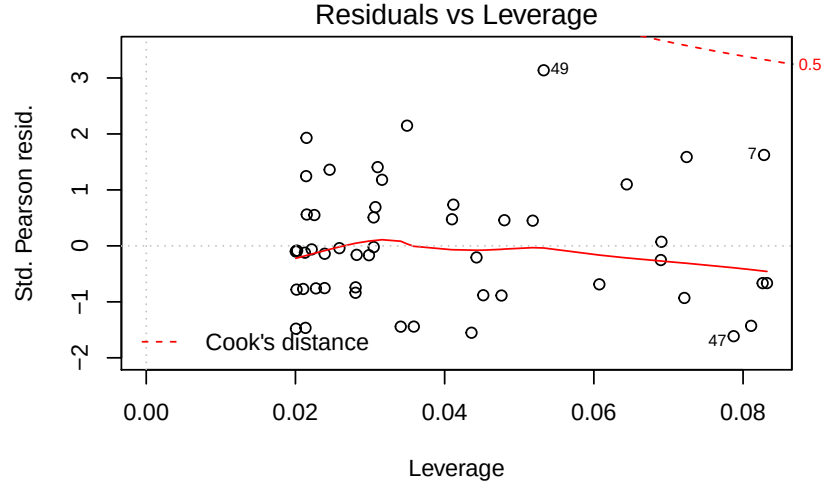


Poisson regression: model checking



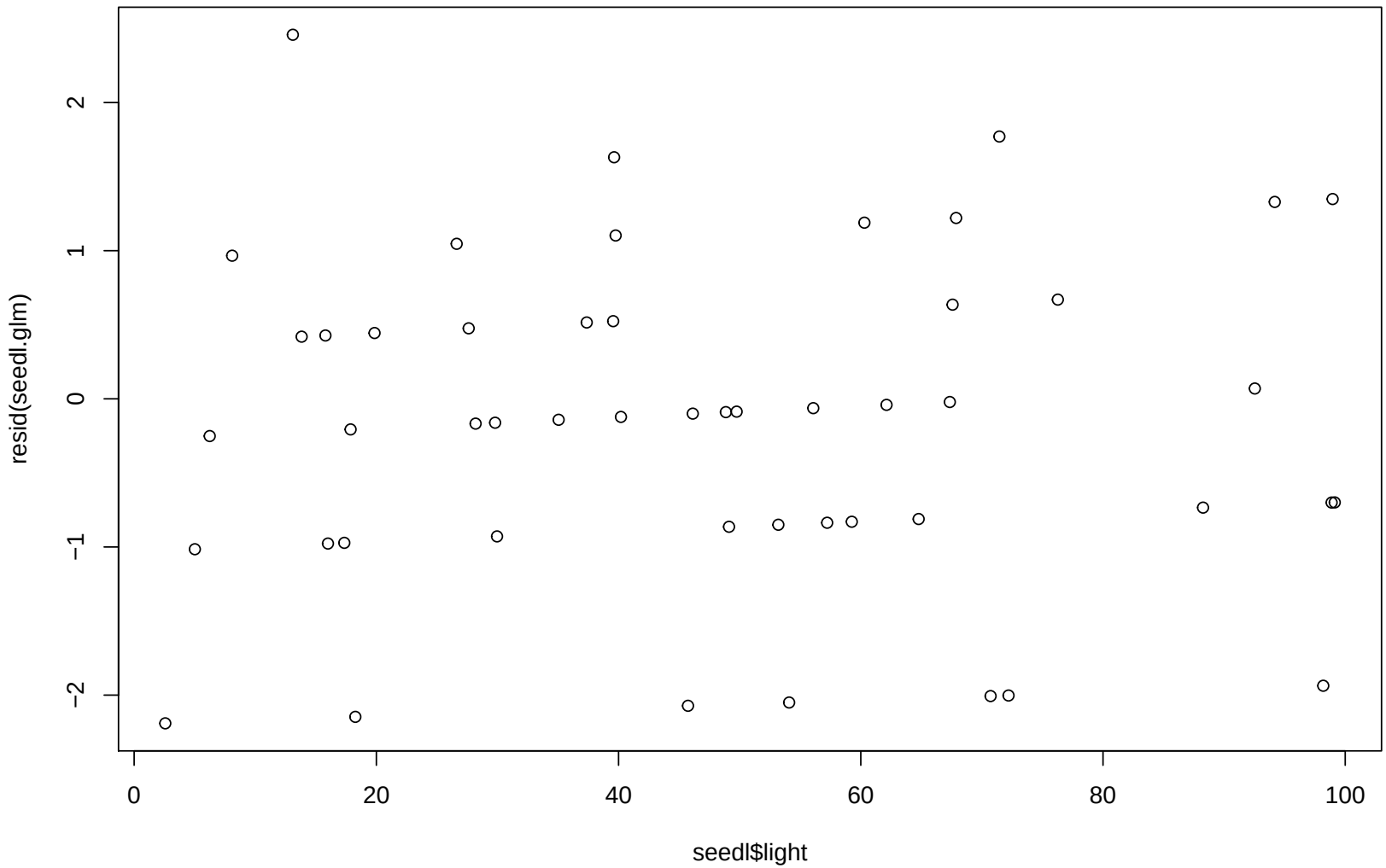






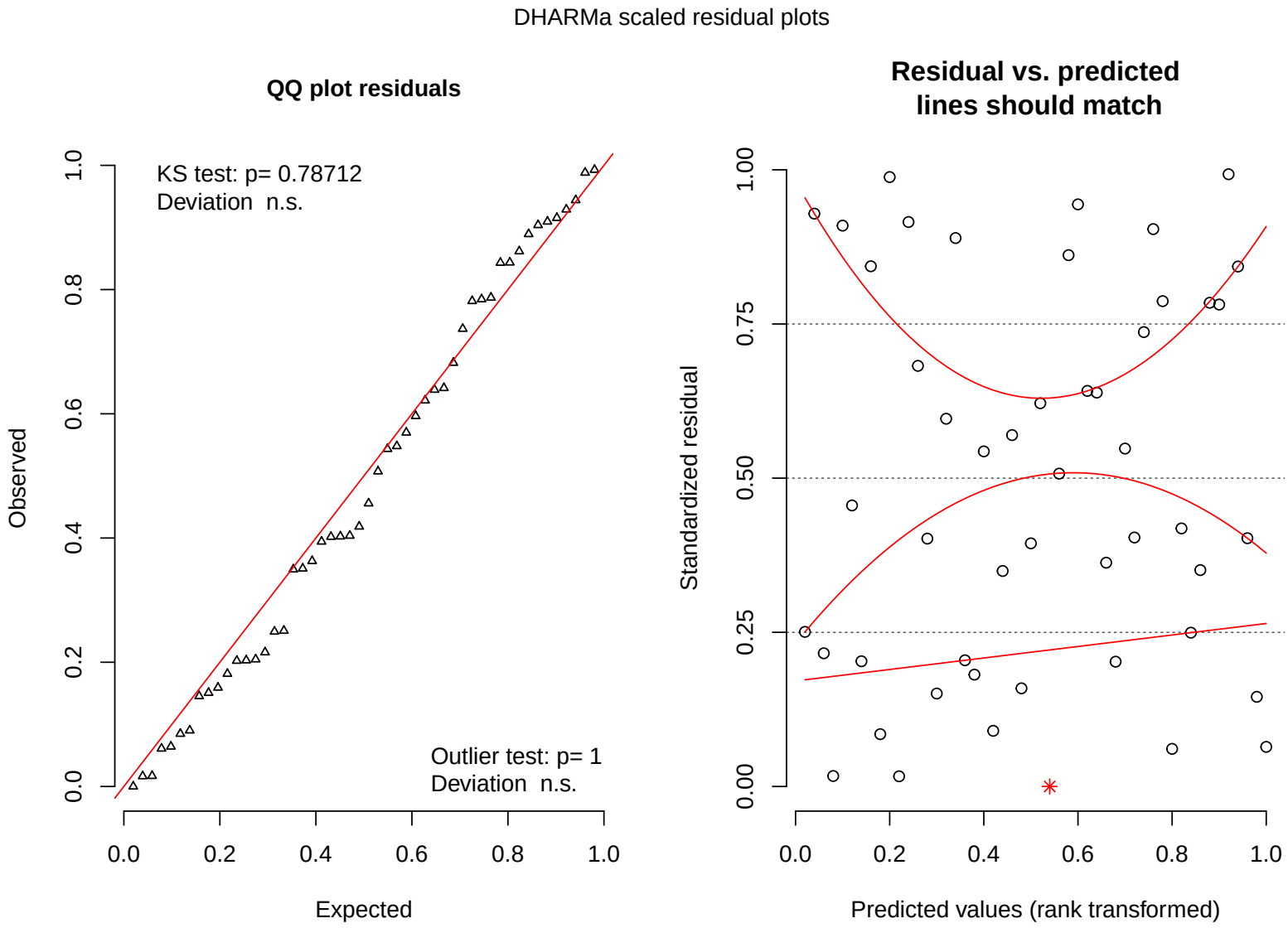
Is there pattern of residuals along predictor?

```
plot(seedl$light, resid(seedl.glm))
```



Residuals diagnostics with DHARMa

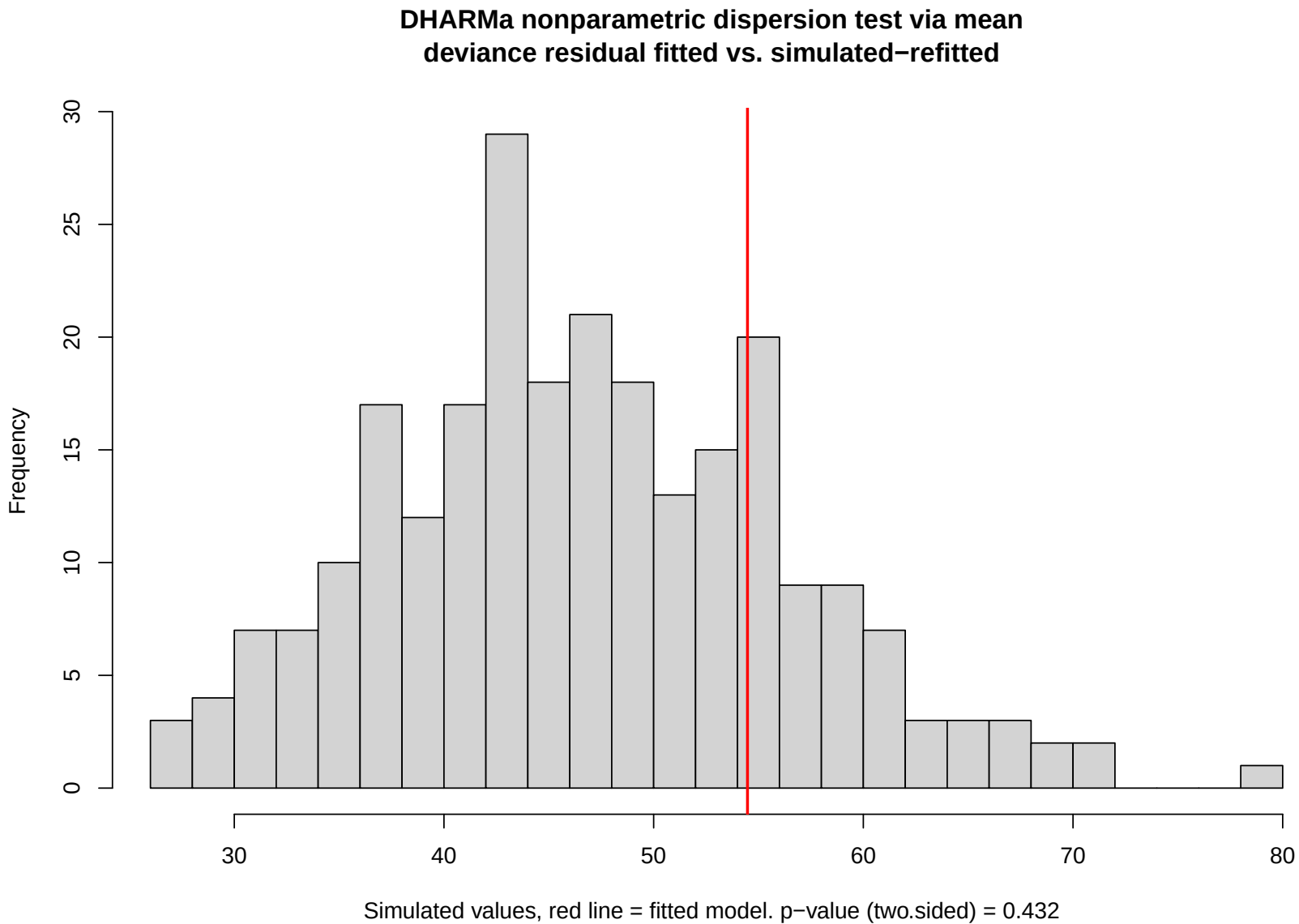
```
simulateResiduals(seedl.glm, plot = TRUE)
```



Poisson regression: Overdispersion

Always check overdispersion with count data

```
simres <- simulateResiduals(seedl.glm, refit = TRUE)
testOverdispersion(simres)
```



Accounting for overdispersion in count data

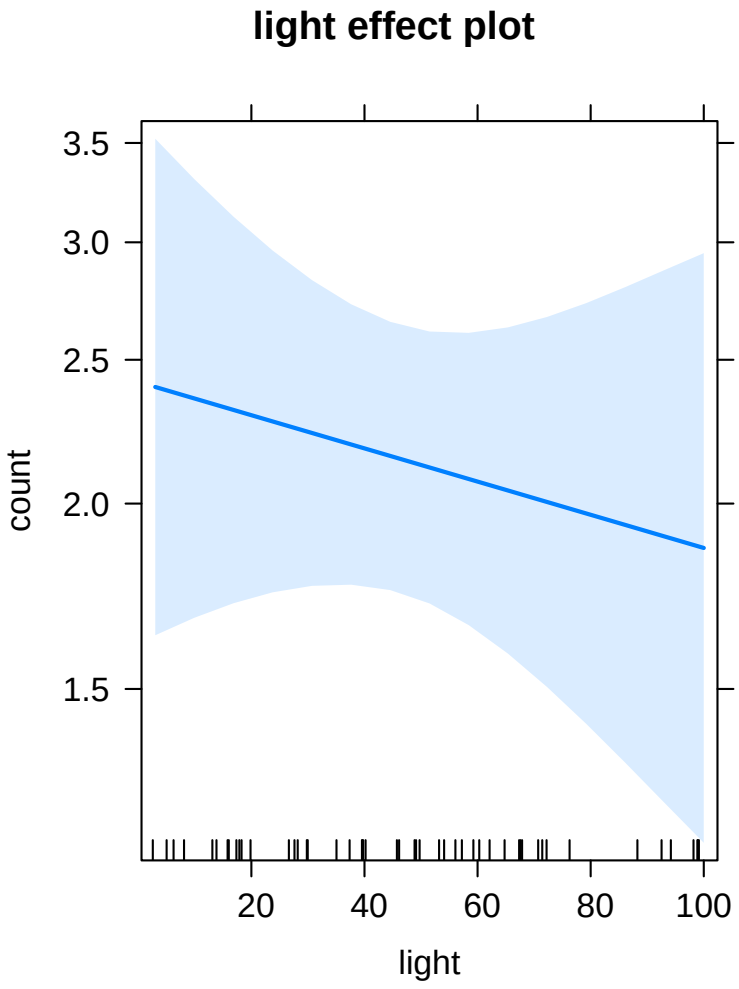
Use family quasipoisson

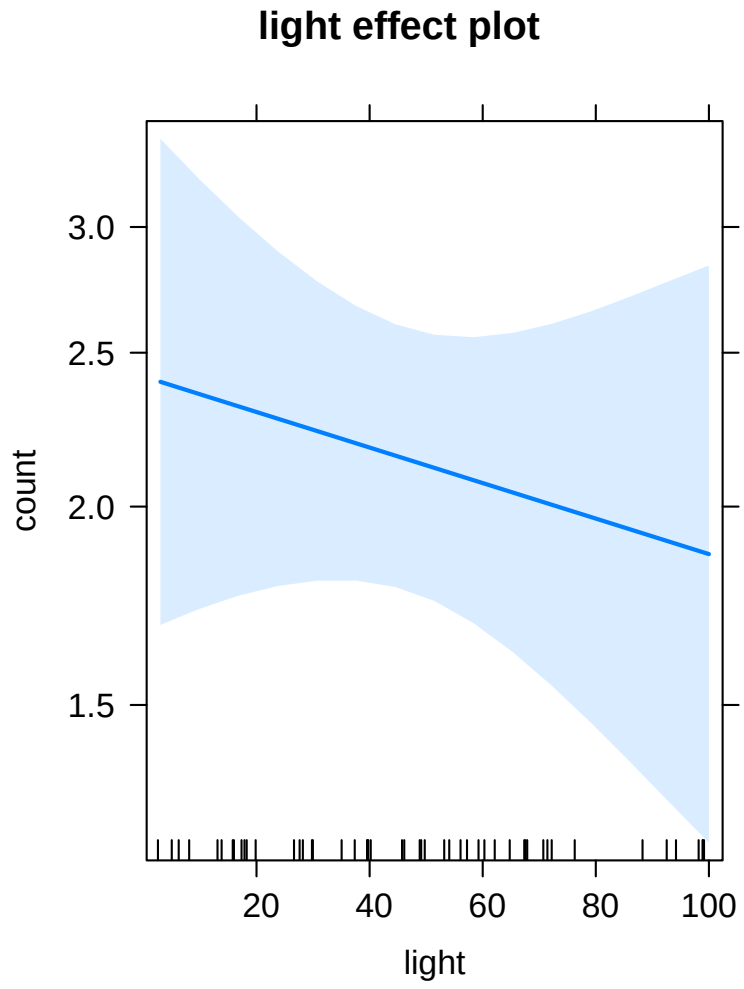
```
Call:
glm(formula = count ~ light, family = quasipoisson, data = seedl

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.1906  -0.8466  -0.1110   0.5220   2.4577

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.881805   0.201230   4.382 6.37e-05 ***
light       -0.002576   0.003758  -0.685    0.496
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for quasipoisson family taken to be 1.1349

    Null deviance: 63.029  on 49  degrees of freedom
Residual deviance: 62.492  on 48  degrees of freedom
```

Mean estimates do not change after accounting for overdispersion

```
 model: count ~ light

 light effect
light
       3        30        50        70       100
2.396665 2.235657 2.123408 2.016794 1.866826

 model: count ~ light

 light effect
light
       3        30        50        70       100
2.396665 2.235657 2.123408 2.016794 1.866826
```

But standard errors may change





What if survey plots have different area?

Avoid regression of ratios

seedlings/area $\sim$ light

J. R. Statist. Soc. A (1993)
156, *Part* 3, *pp.* 379–392

Spurious Correlation and the Fallacy of the Ratio Standard Revisited

By RICHARD A. KRONMAL†

Use offset to standardise response variables in GLMs

```
seedl.offset <- glm(count ~ light, offset = seedl$area, data = s
summary(seedl.offset)
```

```
Call:
glm(formula = count ~ light, family = poisson, data = seedl,
    offset = seedl$area)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.6926  -0.8532   0.1491   0.5211   3.1051

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  0.299469   0.185468   1.615    0.106
light       -0.004498   0.003441  -1.307    0.191

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 70.263  on 49  degrees of freedom
```

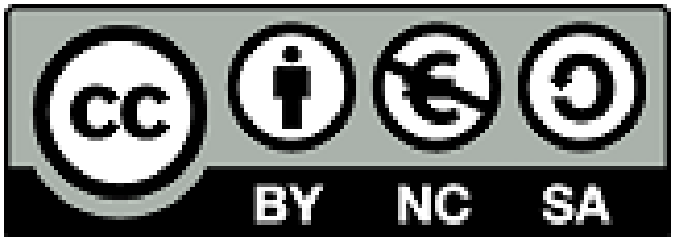
Note estimates now referred to area units

```
exp(coef(seedl.offset))
```

```
(Intercept)       light
  1.3491422   0.9955123
```

Mixed / Multilevel models

END



Source code and materials:
https://github.com/Pakillo/LM-GLM-GLMM-intro