An introduction to linear models in R

R Club

Jacinta Kong

20/10/2021

Linear models

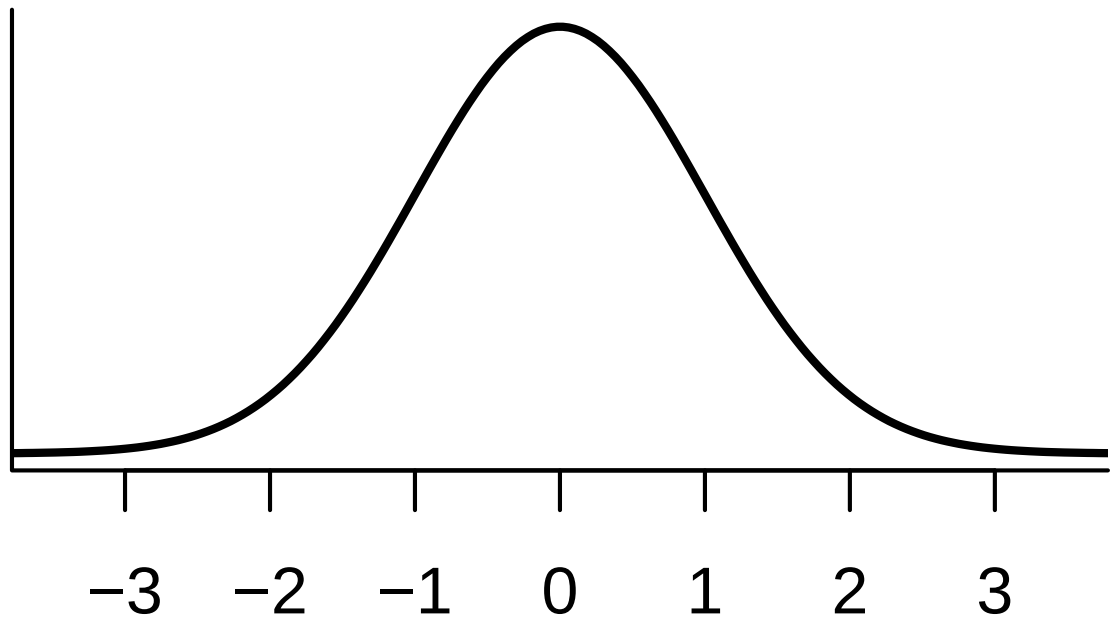
$$Y = \beta_0 + \beta_1 X + \epsilon$$

- Introduction to the fundamentals of linear modelling in R.
- Limited to Type I models or fixed effect models using base `lm` function.
- Other regression packages (e.g. `nlme` and `lmer`) are not covered.

Concepts covered

Gaussian family linear regressions with continuous and/or categorical variables:

- 1 predictor variable (simple regression, ANOVA)
- 2 predictor variables (multiple regression: additive/multiplicative, ANCOVA, two-way ANOVA)



Concepts not covered

- Theory underlying linear regression
- Random effects, including mixed effects models
- Nested or block designs
- Other probability distributions (generalised linear regression)
- Non-linear regression
- Bayesian approaches

Resources

- Lecture notes. Online and downloadable
- Interactive tutorial (this session)

Installing the tutorial

1. Install `learnr`, `remotes` and `rmarkdown`.
   - `learnr` - package for tutorial (will also install `shiny`)
   - `remotes` - easy install from Github
   - `rmarkdown` - to render tutorial

```
install.packages(c("learnr", "remotes", "rmarkdown))
```

2. Download tutorial

```
remotes::install_github("jacintak/biostats",
build_vignettes = TRUE)
```

3. Profit

Open tutorial

- Tutorial tab
- Run tutorial
- Open in new window
- Stop icon

Linear model

The core function is:

```
lm(Y ~ X, data)
```

- `Y` is response variable
- `X` is predictor variable(s)
- `data` is name of dataset

Variables fitted in **alphabetical order**.

$$\text{Height} = \beta_0 + \beta_1(\text{Girth}) + \epsilon \tag{1}$$

```
lm(Height ~ Girth, trees)
```

```
Call:
lm(formula = Height ~ Girth, data = trees)

Coefficients:
(Intercept)        Girth
     62.031        1.054
```

- `(Intercept)` is intercept
- `Girth` is slope

$$\widehat{\text{Height}} = 62.03 + 1.05(\text{Girth}) \tag{2}$$

Analysis of Variance

The core function is `anova`. Uses `lm`:

```
anova(lm(Y ~ X, data))
```

To mess with you, an alternative method is `aov`. Uses `summary`:

```
summary(aov(Y ~ X, data))
```

Tooth growth

Additive model (+) with two categorical variables (Two way ANOVA).

```
anova(lm(len ~ factor(dose) + supp, ToothGrowth))
```

```
Analysis of Variance Table

Response: len
              Df  Sum Sq Mean Sq F value    Pr(>F)
factor(dose)   2 2426.43 1213.22  82.811 < 2.2e-16 ***
supp           1  205.35  205.35  14.017 0.0004293 ***
Residuals     56  820.43   14.65
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 '
```

Multiplicative model (*) with one categorical variable (supp) and one continuous variable (dose) (ANCOVA).

```
anova(lm(len ~ dose * supp, ToothGrowth))
```

```
Analysis of Variance Table

Response: len
           Df  Sum Sq Mean Sq  F value    Pr(>F)
dose        1 2224.30 2224.30 133.4151 < 2.2e-16 ***
supp        1  205.35  205.35  12.3170 0.0008936 ***
dose:supp   1   88.92   88.92   5.3335 0.0246314 *
Residuals  56  933.63   16.67
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 '
```

Summary

`summary` shows more information about the linear model:

- Estimated parameters
- One-sample t-test on estimates (t value and P value)
- R^2 values

Can call individual elements, e.g. `summary(lm(Height ~ Girth, trees))$r.squared`.

```
summary(lm(len ~ dose * supp, ToothGrowth))
```

```
Call:
lm(formula = len ~ dose * supp, data = ToothGrowth)

Residuals:
    Min      1Q  Median      3Q     Max 
-8.2264 -2.8462  0.0504  2.2893  7.9386 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   11.550      1.581   7.304 1.09e-09 ***
dose           7.811      1.195   6.534 2.03e-08 ***
suppVC        -8.255      2.236  -3.691 0.000507 ***
dose:suppVC    3.904      1.691   2.309 0.024631 *  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.083 on 56 degrees of freedom
Multiple R-squared:  0.7296,    Adjusted R-squared:  0.7151 
F-statistic: 50.36 on 3 and 56 DF,  p-value: 6.521e-16
```

The parameterised equation

Cheat by using the `equatiomatic` package for automatic formatting[1].

$$\widehat{\text{len}} = 11.55 + 7.81(\text{dose}) - 8.26(\text{supp}_{\text{VC}}) + 3.9(\text{dose} \times \text{supp}_{\text{VC}}) \tag{3}$$

`broom` is also handy.

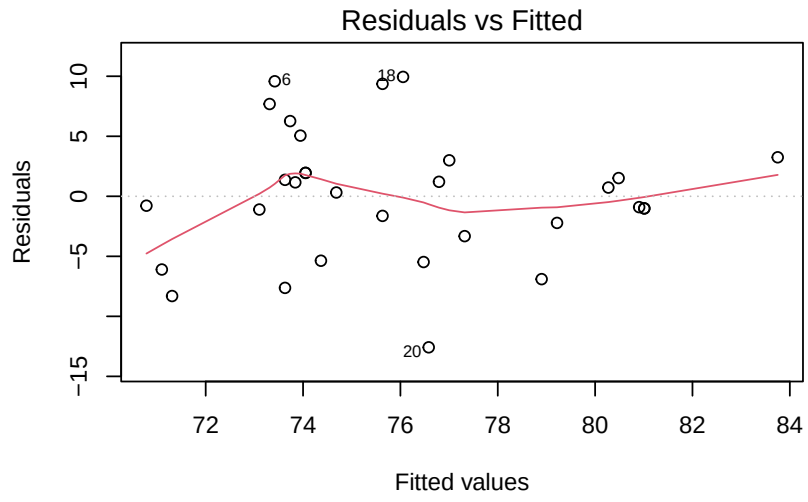
[1] IMHO packages or shiny apps for automatic visualisation of linear regressions and exploratory analyses (e.g. `summarytools`) defeats the point of R's transparency.

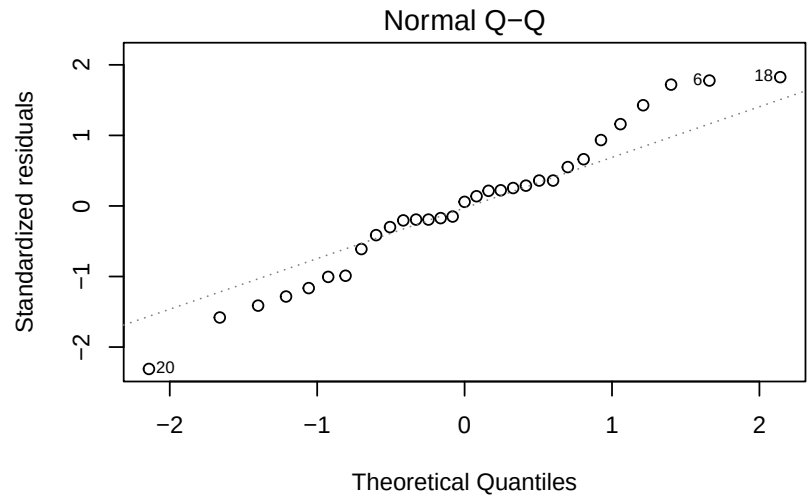
Residual plots

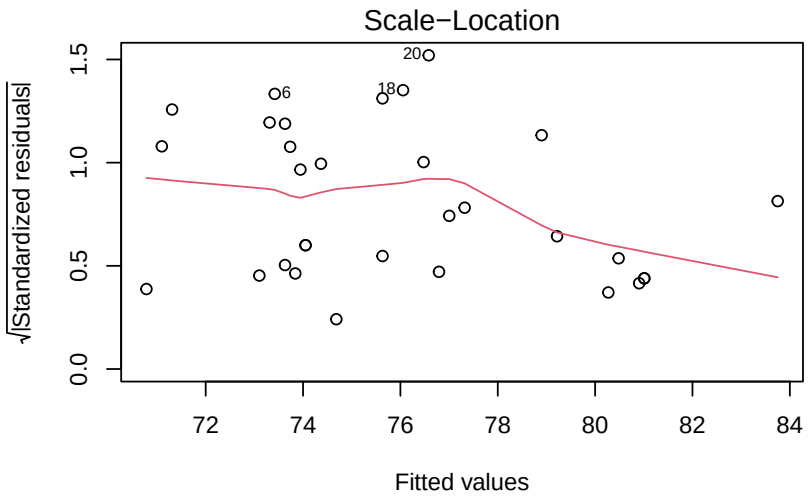
`plot(lm(...))` shows residual plots. 4 plots[2]:

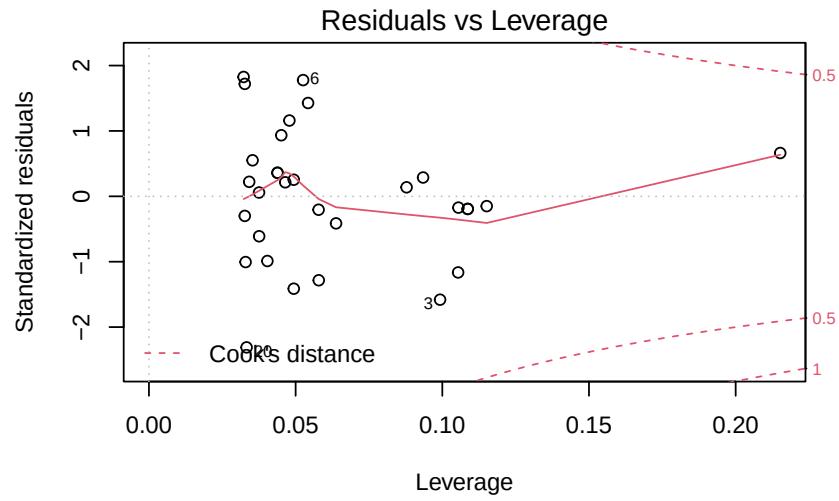
- Residuals vs fitted values (Homogeneity of variance)
- Normal quantile-quantile plot (Normally distributed errors)
- Scale-location plot of standardised residuals (Homogeneity of variance but fancy)
- Residual vs leverage plot (Outliers)

[2] Can use `par(mfrow = c(2,2))` to plot all of them in a 2x2 grid.


Residuals vs Fitted
Residuals
Fitted values
6
18
20
10
5
0
−5
−15
72
74
76
78
80
82
84


Normal Q−Q
Standardized residuals
Theoretical Quantiles
6
18
20
2
1
0
−1
−2


Scale−Location
√|Standardized residuals|
Fitted values
6
18
20
1.5
1.0
0.5
0.0
72
74
76
78
80
82
84


Residuals vs Leverage
Standardized residuals
Leverage
Cook's distance
3
6
20
0.5
1
2
1
0
−1
−2
0.00
0.05
0.10
0.15
0.20

`lm` too normal?[3]

The GLM equivalent is:

```
glm(Y ~ X, data, family = "gaussian")
```

[3] See what I did there?