

# Linear Models, Marked Practical

Week 4, MT 2023

- This practical sheet contains two sections. **Write a report on the Exercise in Section 2 only.**
- **The report has a word limit of 2000 words.** This word limit is on the main body of the report. Equations, tables, figures, captions, appendices to your report and computer code do not contribute to the word count.
- **You should use your anonymous practical ID (of the form P123, and not your name)** for the cover page of the report, and you should name the PDF file you upload using that same ID (e.g. P123.pdf).
- **You should submit your report via the Canvas system.** There will be instructions about how to do this on the Canvas Practicals page next to this practical handout.

You are welcome to ask questions about the examples in Section 1 during the practical session. If this was an assessed practical I would not be able to answer questions regarding the exercise in Section 2, with the sole exception of questions relating to a limited number of programming issues. In order to mimic the setup for an assessed practical, only limited help will be available for the exercise in Section 2.

## 1 Examples for practice, NOT MARKED

### (a) A quadratic term

This came up in the first practical. If you want e.g.  $y = \beta_0 + \beta_1 x + \beta_2 x^2 + \epsilon$ , then you should use `y ~ x + I(x^2)`. It is important to use `I()` here.

```
## one of the introductory examples
plot(dist ~ speed, data = cars)

cars0.lm <- lm(dist ~ speed, data = cars)
cars1.lm <- lm(dist ~ speed + I(speed^2), data = cars)
cars2.lm <- lm(dist ~ speed + speed^2, data = cars)

summary(cars0.lm)
summary(cars1.lm)
summary(cars2.lm)
## cars2.lm is the same as cars0.lm and is probably not what was intended
```

### (b) Box-Cox transformation

Suppose a normal linear model applies not to  $y$ , but to some power of  $y$ , say to  $y^\lambda$ . We can use the Box-Cox method to find the best value of  $\lambda$ . Where possible we might hope for an interpretable value of  $\lambda$ . Faraway (2015): “If explaining the model is important, you should round  $\lambda$  to the nearest interpretable value.”

As  $\lambda$  varies in the range  $(-2, 2)$  we get the inverse transformation ( $\lambda = -1$ ), square and cube roots ( $\lambda = \frac{1}{2}, \frac{1}{3}$ ), the original scale ( $\lambda = 1$ ), as well as the squared case ( $\lambda = 2$ ). We want a sensible  $\lambda = 0$  case as well, so the method actually works with the transformation to  $y^{(\lambda)}$  where

$$y^{(\lambda)} = \begin{cases} \frac{y^\lambda - 1}{\lambda} & \lambda \neq 0 \\ \log y & \lambda = 0. \end{cases}$$

Note this is consistent because  $\lim_{\lambda \rightarrow 0} \left( \frac{y^\lambda - 1}{\lambda} \right) = \log y$ .

We assume all  $y_i$  values satisfy  $y_i > 0$  (if not we could add a small constant to all  $y_i$ s).

We can treat  $\lambda$  as a parameter and find the MLE: see Davison (2003, p389–390), or Faraway (2015, p134–137) for details.

### Example (i)

```
## the boxcox() function is in the MASS package
library(MASS)

trees0.lm <- lm(Volume ~ log(Height) + log(Girth), data = trees)
par(mfrow = c(2, 2))
plot(trees0.lm)

## see ?boxcox
boxcox(Volume ~ log(Height) + log(Girth), data = trees,
       lambda = seq(-0.25, 0.25, length = 10))
abline(v = 0, col = "red")

trees1.lm <- lm(log(Volume) ~ log(Height) + log(Girth), data = trees)
```

```
par(mfrow = c(2, 2))
plot(trees1.lm)
```

## Examples (ii)

```
lmod <- lm(sr ~ pop15 + pop75 + dpi + ddpi, data = LifeCycleSavings)
boxcox(lmod)
boxcox(lmod, lambda = seq(0.5, 1.5, by = 0.1))
## Faraway: "no good reason to transform"

## need the faraway package for the gala data
library(faraway)

lmod <- lm(Species ~ Area + Elevation + Nearest + Scrub + Adjacent, data = gala)
boxcox(lmod, lambda = seq(-0.25, 0.75, by = 0.05))
## Faraway:
## "... perhaps a cube root transformation might be best here.
## A square root is also a possibility ..."
## Certainly there is a strong need to transform."
```

## (c) Further practice

- See the Introductory Examples ([IntroExs.html](#)). Which model would you fit to the advertising data? (with sales as the response variable). Or to the Oxford house price data? (with price as the response variable).
- For examples of interpretations, see [AdvFitInterp.html](#) or [Gas.html](#).  
See also the many examples in the books recommended for the course.
- Weighted regression and Box-Cox transformation (see (b) above) were treated relatively briefly in lectures. For more see [Weighted.html](#) and [BoxCox.html](#).
- Which model would you fit to the `poisons` data in the `boot` package? (with time as the response variable).

## 2 MARKED EXERCISE

The data in `swim.csv` are the competitors' times in some swimming races. The times are from the finals of individual events at the 2016 Olympics, and from the finals of similar events at the 2016 World Championships. The Olympic events were “long course” events – swum in a 50 metre pool; the World Championships were “short course” – swum in a 25 metre pool. The strokes swum in these events were freestyle, backstroke, breaststroke, or butterfly, and also medley. In a medley race, all four of the other strokes are swum, an equal number of lengths of each stroke.

For each event, the times of the finalists are recorded as well as some other information about the event. The variables recorded are:

- `event`, the name of the event, e.g. “50 m Freestyle”
- `dist`, the length of the event, in metres
- `stroke`, the stroke swum in the event
- `sex`, to indicate whether an event is women's or men's
- `course`, to indicate whether an event is short course or long course
- `time`, the time of one of the swimmers in the final, in seconds.

The file `swim.csv` is on Canvas.

You can load the data using something like:

```
swim <- read.csv("swim.csv")
```

*Write a report on the exercise below.*

### Exercise:

You are asked to investigate how race times depend on the other variables.

The primary aim of the analysis is: (i) to obtain an interpretable model that explains how time depends on the other variables; and (ii) to interpret the model you obtain.

You are also asked, using the same model, to predict times for four additional races.

1. Perform an exploratory analysis of the data and summarise your findings. You may wish to consider some numerical summaries as well as some exploratory plots.
2. Model the relation between time and the other variables that are available. Carry out model selection and outlier analysis. Remember when selecting your model that the main aim here is interpretation. (Stick to normal linear models, with fixed effects.)
3. Interpret your final normal linear model carefully – and preferably on the original scale rather than on a transformed scale should your model involve transformed quantities. What is the effect of each of distance/sex/course/stroke on time, and how does this effect vary with the other variables?

Aim to produce one or two plots which illustrate your findings graphically.

4. Using the model obtained above, obtain predicted times and prediction intervals for the four additional races below (prediction intervals, not confidence intervals).

Comment on the predictions you obtain.

name	dist	stroke	sex	course
RaceA	400	Freestyle	F	Long
RaceB	50	Backstroke	F	Long
RaceC	400	Butterfly	F	Long
RaceD	100	Medley	F	Long