Class 2: Regression and classification

Andrew Parnell, School of Mathematics and Statistics,
University College Dublin

Learning outcomes

- Be able to understand the structure of regression and classification models
- Know how to read and interpret the output of a statistical model
- Be familiar with some of the extensions to basic regression and classification models

Why regression and classification?

- t-tests are only really useful when you have a continuous outcome variable and one discrete variable with two groups (e.g. treatment vs control)
- For almost any real life situation you have multiple variables of all different types
- For these situations you need a *statistical model*
- A statistical model allows to perform *probabilistic prediction* of the outcome variable from the remaining variable, and/or to explain how the other variables are causing the outcome variable to change

Regression vs Classification: what's the difference?

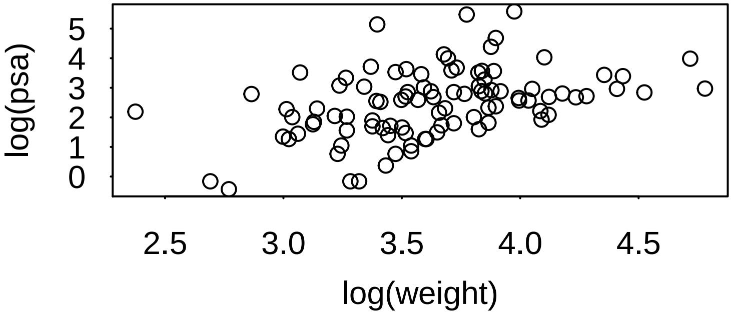
- In regression we have a single *continuous* outcome variable and lots of other variables which we think might be causing the outcome to change
- In classification we have a single *discrete* outcome variable and lots of other variables
- In the machine learning literature this is often known as *supervised learning*
- Situations where there are multiple outcome variables are beyond the scope of this course

Response and explanatory variables

- The outcome variable is more commonly known as the *response* variable
- The other variables which we think might be causing the response variable to change are called the *explanatory variables* (though be careful with causation)
- We will use these words from now on, but beware there are lots of other terms in the literature

A basic regression model

- Let's go back to the prostate cancer data
- Recall the key outcome variable is `lpsa` the log of the prostate specific antigen value. This is our response variable
- Suppose we had one explanatory variable `lweight`



Creating the model

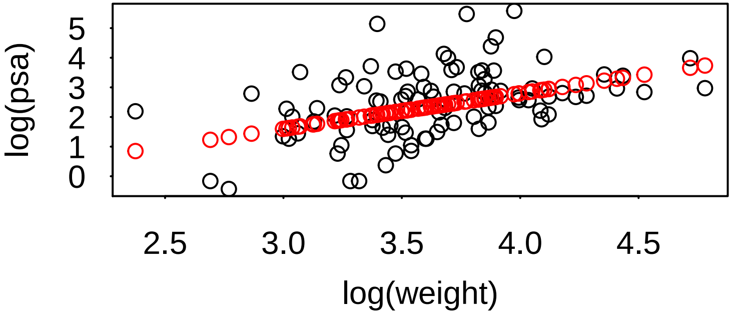
- Looking at the plot, there may be a positive, linear relationship between log(weight) and log(psa)
- Perhaps we can create a prediction model that allows us to predict log(psa) from log(weight)
- Suppose, for each patient we multiplied the log(weight) value by 1.2 and then subtracted the value 2 so:

$$prediction = 1.2 \times \log(weight) - 2$$

- If we do this repeatedly for every value in the data set we get . . .

A first model

```
par(mar=c(3,3,2,1), mgp=c(2,.7,0), tck=-.01,las=1)
prediction = 1.2 * prostate$lweight - 2
plot(prostate$lweight, prostate$lpsa, xlab = ’log(weight)’,
points(prostate$lweight, prediction, col=’red’)
```



Refining the model

- Is this model any good?
- How might we measure how close our predictions are to the truth?
- How can we choose the values (here 1.2 and -2) better?

Getting R to do the work

- Luckily the R function `lm` will do the work for us

```
model_1 = lm(formula = lpsa ~ lweight, data = prostate)
summary(model_1)
```

```
## 
## Call:
## lm(formula = lpsa ~ lweight, data = prostate)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -2.27976 -0.67507 -0.03503  0.53984  2.93649 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  -1.7586     0.9103  -1.932   0.0564 .  
## lweight       1.1676     0.2491   4.686 9.28e-06 ***
## ---
```

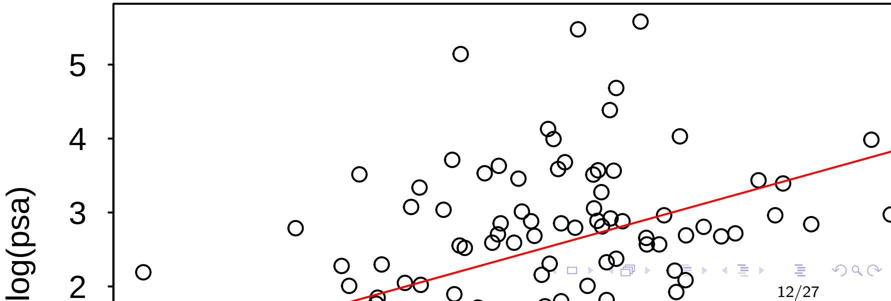
Background details

- The two values here are the *y*-intercept and the slope of the line
- R chooses these values by minimising the vertical distances between the black and the red points (called least squares)
- A key assumption in the model is that these vertical distances (known as *residuals*) are normally distributed
- R uses this assumption to run t-tests on the parameters, which you can see the results of in the `summary` output

Plotting the fit

- One way is to type `plot(model_1)` but this perhaps gives too much info. Better:

```
par(mar=c(3,3,2,1), mgp=c(2,.7,0), tck=-.01,las=1)
plot(prostate$lweight, prostate$lpsa, xlab = ’log(weight)’
abline(model_1, col=’red’)
```



Reading the output of the model

Expanding the model with two explanatory variables

Expanding the fit even more

Regularisation and shrinkage

Lasso; Ridge and Elastic Net

Dealing with interactions

Even more advanced regression approaches

Intro to classification models

The logit transformation

Example: SA Heart rate

Extending the model

Understanding the output

Plotting the fitted model

Regularisation and shrinkage for classification

More advanced classification approaches