STATS544.2 Applied multiple linear regression

Gunnar Stefansson

August 28, 2013

Contents

1 Multiple linear regression background **3**

1.1 The basics . . . 3

1.2 Plotting . . . 3

1.3 Data sets . . . 3

1.4 MULREG case studies . . . 4

1.5 association vs causation . . . 5

2 Multiple linear regression **5**

2.1 Datasets . . . 5

2.2 Point estimation (with R) . . . 7

2.3 Interpreting coefficients . . . 8

2.4 R^2 . . . 9

2.5 P-values . . . 10

2.6 Building a model . . . 10

2.7 Stepwise linear regression . . . 11

2.8 Thou shalt not seek a model with a shotgun (the multiplicity issue) . . . 12

2.9 Multiplicity corrections . . . 13

2.10 Comparing nested models . . . 13

2.11 Comparing non-nested models . . . 13

3 Deviations from assumptions **14**

3.1 Plotting residuals . . . 14

3.2 Common deviations from assumptions . . . 14

3.3 Regression diagnostics . . . 14

3.4 Transformations to normality . . . 15

3.5 Weighting . . . 15

3.6 Serial correlations . . . 15

3.7 Errors in variables . . . 15

4 Extensions to the multiple linear regression model **15**

4.1 Dummy variables . . . 15

4.2 Factors . . . 16

4.3 ANOVA . . . 16

4.4 ANCOVA 16
4.5 Poisson regression, via case studies 18
4.6 Logistic regression, via case studies 19
4.7 Survival analysis (Coxph), via case studies 20

5 Case studies **20**

5.1 Writing a report 20
5.2 Case studies for students 20

1 Multiple linear regression background

1.1 The basics

Want to describe or predict a dependent y variable from several independent/exploratory x variables.
Main package for this course: R
Examples will be from engineering, biology, economics, education etc.

For info on fields which use R, see http://cran.r-project.org/web/views/

Example (economics): The icecream data (economics) plotted in the figure is available at http://tgax14.rhi.hi.is/html/data/icec
or in the Ecdat package.

For more economic data sets, see e.g. http://www.micecon.org/ or http://eu.wiley.com/legacy/wileychi/verbeek2ed/datasets.ht

For economic applications with R see e.g. http://cran.r-project.org/web/views/Econometrics.html

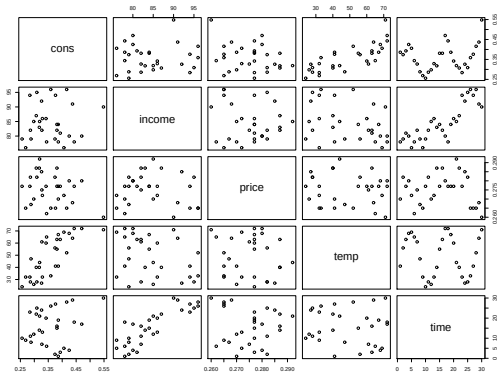


Figure 1: Icecream data.

1.2 Plotting

The first step in analysing data should always consist of plots

Example (education): Comparing scores from several exams.

1.3 Data sets

Data are stored in files, which are then read into data frames. Data files (or data sets) should be structured as simple rectangular tables before they are read into R
The data set is commonly just a table of numbers, possibly with a single header line.
We think of the data frame in the same way - usually as a table of numbers.
Many data sets are available as built-in data sets in R
Data can also be read directly from a URL

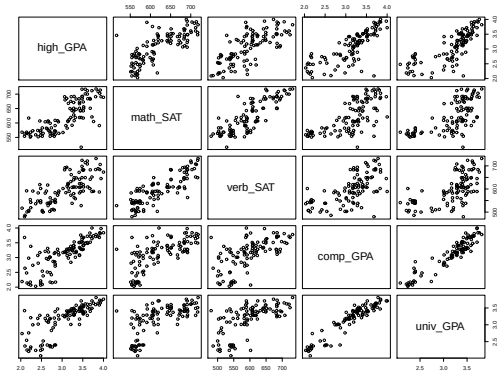


Figure 2: **Example (education):** Typical first plot using the pairs command. Comparing scores from several exams.

Example (medicine): R is commonly used in medicine.

A typical data set preloaded into R is the ais data set in the DAAG library, accessible as follows:

```
library(DAAG)
data(ais)
pairs(ais[,c("rcc","wcc","bmi","ht","wt","pcBfat","lbm")])
```

Figure 3: Data set ais from the DAAG library (see http://cran.r-project.org/web/packages/DAAG/DAAG.pdf) contains several measurements on characteristics of how blood varied with sport body size and sex of athletes.

1.4 MULREG case studies

In a MULREG course you will be asked to do analyze certain data sets (case studies). Depending on the course, these may be assigned by the instructor or you may need to find your own.

Example data sets:

http://www.stats4stem.org/data-sets.html

http://tgax14.rhi.hi.is/html/data/biol/shsamples

UCI Machine Learning Repository http://archive.ics.uci.edu/ml/

Competitions: https://www.kaggle.com/c/informs2010/

Info on fields which use R, see http://cran.r-project.org/web/views/

For more economic data sets, see e.g. http://www.micecon.org/ or http://eu.wiley.com/legacy/wileychi/verbeek2ed/datasets.ht

Economic applications http://cran.r-project.org/web/views/Econometrics.html

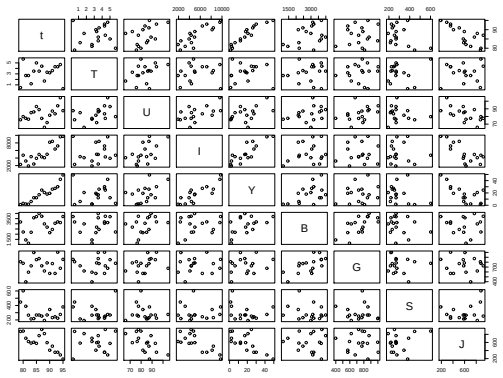
1.5 association vs causation

Evidence of a relationship does not show causation

Example (biology): A data set of several quantities from Icelandic waters can be found at http://tgax14.rhi.hi.is/html/data/biol/
This can be read into R using

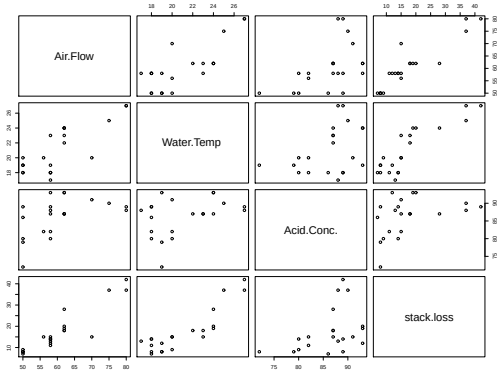
```
b<-read.table("http://tgax14.rhi.hi.is/html/data/biol/borecol.txt",header=T)
```

Just typing "b" shows the content..



2 Multiple linear regression

2.1 Datasets



We will mostly be working with two datasets in the lecture, **stackloss** and **LifeCycleSavings**, both part of the **datasets** package that comes with your installation of R.

The stackloss data

We have operational data of a plant for the oxidation of ammonia to nitric acid. The data were obtained from 21 days of operation of a plant for the oxidation of ammonia (NH3) to nitric acid (HNO3). The nitric oxides produced are absorbed in a countercurrent absorption tower.

- Air Flow: the rate of operation of the plant.
- Water Temp: the temperature of cooling water circulated through coils in the absorption tower.
- Acid Conc.: the concentration of the acid circulating, minus 50, times 10: that is, 89 corresponds to 58.9 per cent acid.
- stack.loss: 10 times the percentage of the ingoing ammonia to the plant that escapes from the absorption column unabsorbed; that is, an (inverse) measure of the over-all efficiency of the plant.

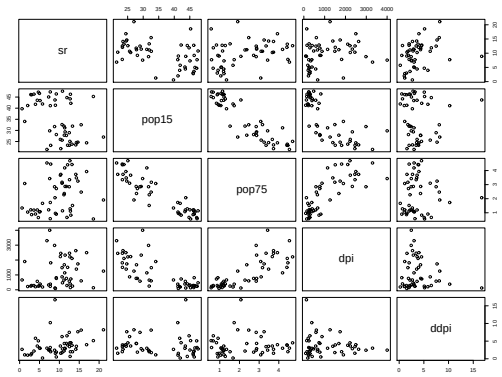
Source: Brownlee, K. A. (1960, 2nd ed. 1965) Statistical Theory and Methodology in Science and Engineering. New York: Wiley. pp. 491–500.

The LifeCycleSavings data

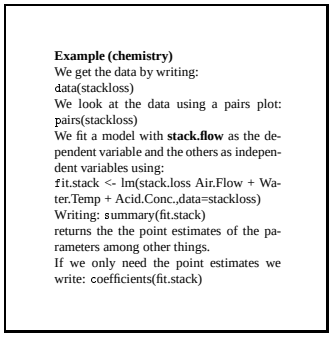
Under the life-cycle savings hypothesis as developed by Franco Modigliani, the savings ratio (aggregate personal saving divided by disposable income) is explained by per-capita disposable income, the percentage rate of change in per-capita disposable income, and two demographic variables: the percentage of population less than 15 years old and the percentage of the population over 75 years old. The data are averaged over the decade 1960–1970 to remove the business cycle or other short-term fluctuations.

- sr: aggregate personal savings
- pop15: % of population under 15
- pop75: % of population over 75
- dpi: real per-capita disposable income
- ddpi: growth rate of dpi

Source: The data were obtained from Belsley, Kuh and Welsch (1980). They in turn obtained the data from Sterling (1977).



2.2 Point estimation (with R)



Example (chemistry)

We get the data by writing:

```
data(stackloss)
```

We look at the data using a pairs plot:

```
pairs(stackloss)
```

We fit a model with **stack.flow** as the dependent variable and the others as independent variables using:

```
fit.stack <- lm(stack.loss~Air.Flow + Water.Temp + Acid.Conc.,data=stackloss)
```

Writing:

```
summary(fit.stack)
```

returns the the point estimates of the parameters among other things.

If we only need the point estimates we write:

```
coefficients(fit.stack)
```

Example (economics)

We get the data by writing:

```
data(LifeCycleSavings)
```

We look at the data using a pairs plot:

```
pairs(LifeCycleSavings)
```

We fit a model with **sr** as the dependent variable and the others as independent variables using:

```
fit.life <- lm(sr ~ pop15 + pop75 + dpi + ddpi ,data=LifeCycleSavings)
```

Writing:

```
summary(fit.life)
```

returns the the point estimates of the parameters among other things.

If we only need the point estimates we write:

```
coefficients(fit.life)
```

2.3 Interpreting coefficients

Example (chemistry)

We continue working with the **stackloss** data.

We can use our model to predict y, (10 times) the percentage of the ingoing ammonia to the plant that escapes from the absorption column unabsorbed using the model (the following model can possibly be reduced, see the slide on building a model).

A typical R session could run as follows:

```
> data(LifeCycleSavings)
> fit.life <- lm(sr ~ pop15 + pop75 + dpi + ddpi ,data=LifeCycleSavings)
> summary(fit.life)

Call:
lm(formula = sr ~ pop15 + pop75 + dpi + ddpi, data = LifeCycleSavings)

Residuals:
    Min      1Q  Median      3Q     Max
-8.2422 -2.6857 -0.2488  2.4280  9.7509

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 28.5660865  7.3545161   3.884 0.000334 ***
pop15       -0.4611931  0.1446422  -3.189 0.002603 **
pop75       -1.6914977  1.0835989  -1.561 0.125530
dpi         -0.0003369  0.0009311  -0.362 0.719173
ddpi         0.4096949  0.1961971   2.088 0.042471 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.803 on 45 degrees of freedom
Multiple R-squared: 0.3385,Adjusted R-squared: 0.2797
F-statistic: 5.756 on 4 and 45 DF,  p-value: 0.0007904
```

$$\hat{y} = -39.920 + 0.7156x_1 + 1.295x_2 - 0.152x_3$$

where x_1 is the air flow , x_2 is the water temperature and x_3 is the acid concentration.

This means that increasing the air flow by one unit increases $\hat{y}$ by 0.7156 units (holding other variables constant), increasing the water temperature by one unit increases $\hat{y}$ by 1.295 units (holding other variables constant) and increasing acid concentration by one unit decreases (note the sign) $\hat{y}$ by 0.152 units (holding other variables constant).

Example (economics)

We continue working with the **LifeCycleSavings data.**

We can use our model to predict y, the savings ratio, using the model (the following model can possibly be reduced, see the slide on building a model):

$$\hat{y} = 28.5661 - 0.4612x_1 - 1.6915x_2 - 0.0003x_3 + 0.4107x_4$$

where x_1 is the % of population under 15, x_2 is the % of population over 75, x_3 is the real per-capita disposable income and x_4 is the % growth rate of dpi

This means that increasing the % of population under 15 by one unit decreases $\hat{y}$ by 0.4612 units (holding other variables constant), increasing the % of population over 75 by one unit decreases $\hat{y}$ by 1.6915 units (holding other variables constant), increasing the real per-capita disposable income by one unit decreases $\hat{y}$ by 0.0003 units (holding other variables constant) and increasing the growth rate of dpi by one unit increases $\hat{y}$ by 0.4107 units (holding other variables constant),

2.4 R^2

The summary command gives the R^2 value.
It can also be extracted directly, for use in computations.
As usual, this is $1 - SSE/SSTOT$.
It is also the squared correlation between y and $\hat{y}$.

In R, the summary() command gives the R^2 value.

It can also be extracted directly, for use in computations.

As usual, this is $1 - SSE/SSTOT$, i.e. R^2 denotes the proportion of variation explained by the model.

It is also the squared correlation between y and $\hat{y}$.

Example (chemistry)

We continue working with the **stackloss** data.

We can get the R^2 along with other things writing:

```
summary(fit.stack)
```

The R^2 can be extracted with

```
summary(fit.stack)$r.squared
```

Example (economics)

We continue working with the **LifeCycleSavings data.**

We can get the R^2 along with other things writing:

```
summary(fit.life)
```

The R^2 can be extracted with

```
summary(fit.life)$r.squared
```

2.5 P-values

Example (chemistry)

We continue working with the **stackloss** data.

By writing

```
summary(fit.stack)
```

we get some statistics and informations about out model. In the last column of the Coefficients table, marked $Pr(>|t|)$ we get the p-values for the tests of the individual coefficients.

On the last line of the output we get the p-value for the overal test that all the coefficients of the model are equal to zero.p

In general we reject the null hypothesis that the coefficient(s) is equal to zero if the p-value is smaller than the α-value used.

Example (economics)

We continue working with the **LifeCycleSavings data.**

By writing

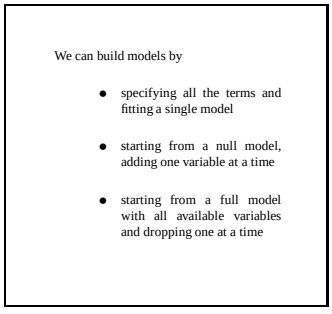
```
summary(fit.life)
```

we get some statistics and informations about out model. In the last column of the Coefficients table, marked $Pr(>|t|)$ we get the p-values for the tests of the individual coefficients.

On the last line of the output we get the p-value for the overal test that all the coefficients of the model are equal to zero.

In general we reject the null hypothesis that the coefficient(s) is equal to zero if the p-value is smaller than the α-value used.

2.6 Building a model



We can build models by

- specifying all the terms and fitting a single model
- starting from a null model, adding one variable at a time

- starting from a full model with all available variables and dropping one at a time

Example (chemistry)

We continue working with the **stackloss** data.

If we use backward selection method building our stackloss model we start by including all the explanatory variables as we did before:

```
fit.stack <- lm(stack.loss~Air.Flow + Water.Temp + Acid.Conc.,data=stackloss)
summary(fit.stack)
```

We look at the p-values for the individual coefficients and see if some of them are larger than the α-value used. The p-value when testing if the parameter for the acid concentration is equal to zero is 0.34405 and therefore the variable is removed from the model. That can be done in a convenient manner by writing

```
fit.stack.2<-update(fit.stack,~.-Acid.Conc.)
summary(fit.stack.2)
```

Note the notation; the "." means use same explanatory variables as in the original model (fit.stack) and then we remove the Acid.Conc. variable by wring "-Acid.Conc.". Now all the p-values are small so we stop.

Example (economics)

We continue working with the **LifeCycleSavings data.**

If we use backward selection method building our LifeCycleSavings model we start by including all the explanatory variables as we did before:

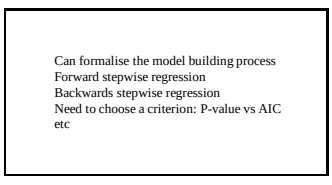
```
fit.life <- lm(sr ~ pop15 + pop75 + dpi + ddpi ,data=LifeCycleSavings)
summary(fit.life)
```

We look at the p-values for the individual coefficients and see if some of them are larger than the α-value used. The p-value when testing if the parameter for the dpi is equal to zero is 0.719173 (the largest one) and therefore the variable is removed from the model. That can be done in a convenient manner by writing

```
fit.life.2<-update(fit.life,~.-dpi)
summary(fit.life.2)
```

Note the notation; the "." means use same explanatory variables as in the original model (fit.life) and then we remove the dpi variable by wring "-dpi". Looking at the p-values for the coefficients in our new model we can see that the largest p-value is the one for the pop75 variable (0.072473). Although we might consider removing that one as well, it is probably best left in the model since it is so close to being significant.

2.7 Stepwise linear regression



Example (chemistry)

We continue working with the **stackloss** data.

It is easy to perform stepwise regression in R using the step() function. The AIC criteria is used. We can choose backward, forward or both directions using the direction argument (both being the default).

```
step(fit.stack)
```

The step method returns the same model as we found using the backward selection method before.

Example (economics)

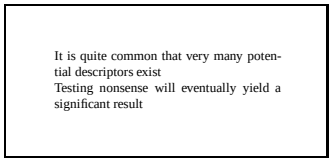
We continue working with the **LifeCycleSavings data.**

It is easy to perform stepwise regression in R using the step() function. The AIC criteria is used. We can choose backward, forward or both directions using the direction argument (both being the default).

```
step(fit.life)
```

The step method returns the same model as we found using the backward selection method before.

2.8 Thou shalt not seek a model with a shotgun (the multiplicity issue)



Example (biology): Consider again the Icelandic ecosystem data. Suppose we want to search for significance among the variables in the data set but also have 100 more variables - all of which are noise.

This can easily be simulated in R:

```
b<-read.table("http://tgax14.rhi.hi.is/html/data/biol/borecol.txt",header=T)
n<-nrows(b)
bad100<-matrix(rnorm(n*100),nrow=n)
newb<-as.data.frame(cbind(b,bad100))
```

Now check, which variable is most highly correlated with the growth in G.

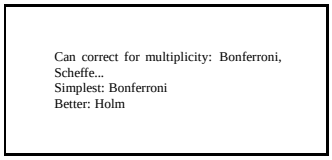
The result is that one of the simulated noise variables becomes the one with the highest correlation to the growth. As a single regression variable it also appears highly significant.

A typical such session could be

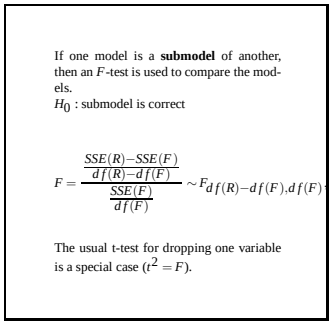
```
n<-nrow(b)
bad100<-matrix(rnorm(n*100),nrow=n)
dim(bad100)
dat<-as.data.frame(cbind(b,bad100))
dim(b)
dim(dat)
max(abs(cor(dat$G,dat[,-7])))
round(cor(dat$G,dat[,-7]),3)
```

- but actual commands will vary on the noise pattern generated.

2.9 Multiplicity corrections



2.10 Comparing nested models



If one model is a **submodel** of another, then an F-test is used to compare the models.

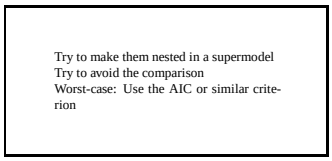
H_0 : submodel is correct

$$F = \frac{\frac{SSE(R)-SSE(F)}{df(R)-df(F)}}{\frac{SSE(F)}{df(F)}} \sim F_{df(R)-df(F),df(F)}, \text{ if } H_0 \text{ is true.}$$

The usual t-test for dropping one variable is a special case ($t^2 = F$).

Example: Check to see whether one or two straight lines are needed to explain a data set.

2.11 Comparing non-nested models



Example (biology): The case of natural mortality vs serial correlation in fish stock assessments (Myers et al).

Example (general): Variable intercept vs variable slope (more later).

3 Deviations from assumptions

3.1 Plotting residuals

Example (chemistry)

We continue working with the **stackloss** data.

As a first tool to check it the assumptions of the model are fulfilled (see more in the tutorial on diagnostics) we plot the residuals against the fitted values of our model. There should be no pattern to see in the plot. We get the fitted values using the **fitted()** function in R and the residuals using the **r**esid() function in R.

```
plot(fitted(fit.stack.2),residuals(fit.stack.2))
```

To check if the residuals follow a normal distribution we use the **q**qnorm() function and to get a line on the plot we use the **qqline()** function

```
qqnorm(residuals(fit.stack.2))
qqline(residuals(fit.stack.2))
```

Example (economics)

We continue working with the **LifeCycleSavings data.**

As a first tool to check it the assumptions of the model are fulfilled (see more in the tutorial on diagnostics) we plot the residuals against the fitted values of our model. There should be no pattern to see in the plot. We get the fitted values using the **fitted()** function in R and the residuals using the **r**esid() function in R.

```
plot(fitted(fit.life.2),residuals(fit.life.2))
```

To check if the residuals follow a normal distribution we use the **q**qnorm() function and to get a line on the plot we use the **qqline()** function

```
qqnorm(residuals(fit.life.2))
qqline(residuals(fit.life.2))
```

3.2 Common deviations from assumptions

3.3 Regression diagnostics

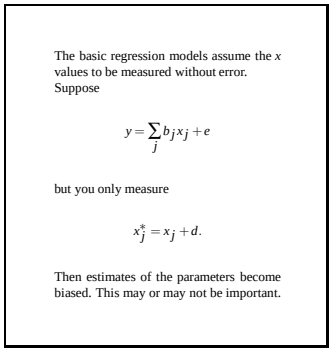
The same regression diagnostics apply as for SLR.
In addition, matrix methods give tools based on the hat matrix.

3.4 Transformations to normality

3.5 Weighting

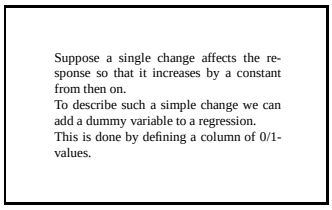
3.6 Serial correlations

3.7 Errors in variables



4 Extensions to the multiple linear regression model

4.1 Dummy variables



Suppose a single change affects the response so that it increases by a constant from then on, or it affects only some subjects.

To describe such a simple change we can add a dummy variable to a regression.

This is done by defining a new x-variable, or column in the X-matrix, consisting of 0/1-values.

A regression with only a dummy variable is equivalent to a t-test.

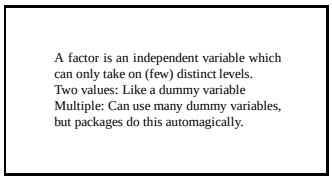
Dummy variables can be used to test whether slopes or intercepts or both are different for groups of subjects or time periods.

These dummy variables can be included in very many different ways.

It is important to understand the meaning of estimates depending on how the variables are included.

Examples: A pollution incident; different intercepts depending on sex; a medical treatment.

4.2 Factors



A factor is an independent variable which can only take on (few) distinct levels.

When a factor can only take on two values we can use a dummy variable to describe the effect.

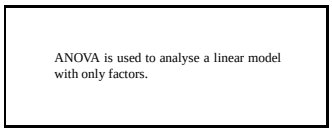
Factors can also be used to test very generally whether a straight line is appropriate using a lack-of-fit test.

A model with only a single factor will give parameter estimates which are the sample mean for each group.

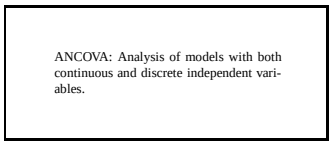
Example: Yield from a field will depend on location (farm).

tapply() can be used to compute means for each level.

4.3 ANOVA



4.4 ANCOVA



The dataset **ToothGrowth** includes the results of an experiment where the length of odontoblasts (teeth) in 10 guinea pigs was measured. The pigs got three different dose levels of Vitamin C (0.5, 1, and 2 mg) with each of two delivery methods (orange juice or ascorbic acid).

Source of data C. I. Bliss (1952) The Statistics of Bioassay. Academic Press.

We get the data by writing

```
data(ToothGrowth)
```

Lets start by plotting the data:

```
plot(ToothGrowth$dose,ToothGrowth$len,pch=20,xlab="Dose",ylab="Length")   # pch = 20 to get sol
```

A better way would be to have different colors for the two delivering methods:

```
color = as.numeric(as.factor(ToothGrowth$supp))+1  # +1 to get red and green
plot(ToothGrowth$dose,ToothGrowth$len,col=color,pch=20,xlab="Dose",ylab="Length")
legend("topleft",legend=c("OJ","VC"),fill=c(2,3))
```

Now we fit a ANCOVA model to the data using the lm() function. A model including one continuous variable and one factor variable with two levels corresponds to fitting two regression lines, one for one level of the factor and another one for the other (different slopes and intercepts), but still using only one model. This way we can test if the lines have different intercepts and/or slopes.

We fit the model writing

```
fit.tooth<-lm(len~dose+supp+supp*dose,data=ToothGrowth)
```

and get the parameter estimates and tests writing

```
summary(fit.tooth)
```

We interpret the parameter estimates in the following way:

```
(Intercept)   11.550 - This is the intercept of the line for the OJ-line
dose           7.811 - This is the slope of the line for the OJ-line
suppVC        -8.255 - This is what we need to add/withdraw from the estimate of the intercept
dose:suppVC    3.904 - This is what we need to add/withdraw from the estimate of the slope of t
```

By looking at the whole output from the summary() function we see that both the intercepts and the slopes of the lines are different.

To see if the interpretation of the parameter estimates is ok we fit two models, one for the OJ-part of the data and another for the VC-part.

```
fit.tooth.oj<-lm(len~dose,data=subset(ToothGrowth,supp=="OJ"))
summary(fit.tooth.oj)
fit.tooth.vc<-lm(len~dose,data=subset(ToothGrowth,supp=="VC"))
summary(fit.tooth.vc)
```

We can add the two lines to our plot with

```
abline(fit.tooth.oj,col=2)
abline(fit.tooth.vc,col=3)
```

To see how the parameters are estimated, we can compare fits based on the inclusion of an intercept or not. First take the case with no intercept and different slopes per factor level:

```
> fit.full<-lm(len~-1+supp+supp:dose,data=ToothGrowth)
> summary(fit.full)

Call:
lm(formula = len ~ -1 + supp + supp:dose, data = ToothGrowth)

Residuals:
     Min       1Q   Median       3Q      Max
-8.22643 -2.84625  0.05036  2.28929  7.93857

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
suppOJ        11.550      1.581   7.304 1.09e-09 ***
suppVC         3.295      1.581   2.084   0.0418 *
```

```
suppOJ:dose    7.811      1.195   6.534 2.03e-08 ***
suppVC:dose   11.716      1.195   9.800 9.44e-14 ***
---
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: 0.9622,     Adjusted R-squared: 0.9595
F-statistic: 356.2 on 4 and 56 DF,  p-value: < 2.2e-16
```

Then take the case of when we estimate an intercept and also an overall slope before adding the mixed term:

```
> fit.full<-lm(len~supp+dose+supp:dose,data=ToothGrowth)
> summary(fit.full)

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

Residuals:
     Min       1Q   Median       3Q      Max
-8.22643 -2.84625  0.05036  2.28929  7.93857

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   11.550      1.581   7.304 1.09e-09 ***
suppVC        -8.255      2.236  -3.691 0.000507 ***
dose           7.811      1.195   6.534 2.03e-08 ***
suppVC:dose    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

>
```

4.5 Poisson regression, via case studies

The dataset **ACF1** from the **DAAG** package includes the results of an experiment where the number of aberrant crypt foci (ACF) in the section 1 of the colons of 22 rats subjected to a single dose of the carcinogen azoxymethane (AOM), sacrificed at 3 different times where counted. Two variables are in the dataset **count:** The number of ACF observed in section 1 of each rat colon **endtime:** Time of sacrifice, in weeks following injection of AOM

Source: Ranjana P. Bird, Faculty of Human Ecology, University of Manitoba, Winnipeg, Canada.

We get the package and data by writing

```
install.packages("DAAG")
library(DAAG)
data(ACF1)
```

Since we are working with count data, the Poisson distribution might be appropriate. We use the glm() function in R for Poisson regression. We start by plotting the data:

```
plot(count ~ endtime, data=ACF1, pch=20)     # pch = 20 for filled dots
```

We fit the model with:

```
ACF.glm0 <- glm(formula = count ~ endtime, family = poisson, data = ACF1)
```

By looking a the plot again we can see that there seems to be a quadratic effect so we try to include that in our model.

```
ACF.glm <- glm(formula = count ~ endtime + I(endtime^2),family = poisson, data = ACF1)
```

We look at the results using the summary() function

```
summary(ACF.glm)
```

We get the estimates for the parameters along with the Wald tests. We also get the null deviance and the residual deviance. We can use the residual variance to test to overall fitness of the model.

We can use the residual deviance to perform a goodness of fit test for the overall fitness of the model. It is the difference between the deviance of the current model and and the deviance of the ideal model where the predicted values are identical to the observed. If the residual difference is small enough the goodness of fit test will not be significant indicating that the model fits the data. We get the p-value for the test using 1-pchisq(ACF.glm*deviance*,*ACF.glm*df.residual)

The p-value is larger than 0.05 indicating that the model fits the data.

4.6 Logistic regression, via case studies

In this example we are going to use a data set that involves the survival of the members of the Donner party which is the most famous tragedy in the history of the westward migration in the United States. In spring of 1846, a group of American pioneers set out for California However, they experienced a series of setbacks and did not arrive at the Sierra Nevada mountains until October. While crossing the mountains, they became trapped by an snowfall, and had to spend the winter there and almost one-half starved to death. The data include some information about each of the members of the party.

Source: Johnson, K. (1996). Unfortunate Emigrants: Narratives of the Donner Party. Logan, UT: Utah State University Press

You can get the data by writing

```
install.packages("alr3")
library(alr3)
data(donner)
```

There are five variables in the dataset:

Age: Approximate age in 1846 Outcome: 1 if survived, 0 if died Sex: Male or Female Family.name: Either a family name, hired or single Status: Family, single or hired

Since the response variable (Outcome) is binary (0 or 1) it is natural to use a logistic model. We fit a model using:

```
fit1 <- glm(Outcome~Age*Sex+Status,donner,family="binomial")
```

This model includes the main effects Age, Sex and Status along with an interaction between Age and Status (allowing for different effect of Age males and females). We look at the results using the summary() function:

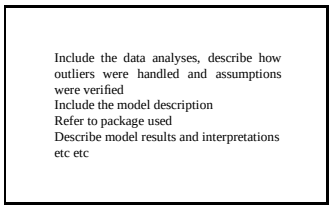
```
summary(fit1)
```

We get the parameter estimates along with the deviance. We can test our model against a model with only a mean using:

4.7 Survival analysis (Coxph), via case studies

5 Case studies

5.1 Writing a report



Include the data analyses, describe how outliers were handled and assumptions were verified

Include the model description

Refer to package used

Describe model results and interpretations

etc etc

5.2 Case studies for students