Workshop 1: solutions

1. (a) The residual sum of squares is given by

$$\mathcal{S}(\beta) = \sum_{i=1}^{n}(y_i - \beta)^2,$$

which implies that

$$\frac{\mathrm{d}}{\mathrm{d}\beta}S(\beta) = -2\sum_{i=1}^{n}(y_i - \beta).$$

Setting this derivative to zero and solving with respect to β gives the least squares estimate of β, $\hat{\beta}$:

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}\beta}S(\beta) = 0 &\Rightarrow -2\sum_{i=1}^{n}(y_i - \beta) = 0\\ &\Rightarrow \sum_{i=1}^{n}(y_i - \beta) = 0\\ &\Rightarrow \hat{\beta} = \frac{\sum_{i=1}^{n} y_i}{n} = \bar{y}.\end{aligned} \tag{1}$$

The residuals in this model are simply given by

$$\hat{\epsilon}_i = y_i - \hat{\beta}, \quad i = 1, \ldots, n.$$

By construction, it holds that (see Equation (1))

$$\sum_{i=1}^{n}(y_i - \hat{\beta}) = 0 \Leftrightarrow \sum_{i=1}^{n}\hat{\epsilon}_i = 0.$$

(b) The residual sum of squares for this model is

$$\mathcal{S}(\alpha, \beta) = \sum_{i=1}^{n}(y_i - \alpha - \beta x_i)^2,$$

and therefore the partial derivatives with respect to α and β are given by

$$\begin{aligned}\frac{\partial}{\partial\alpha}\mathcal{S}(\alpha, \beta) &= -2\sum_{i=1}^{n}(y_i - \alpha - \beta x_i),\\ \frac{\partial}{\partial\beta}\mathcal{S}(\alpha, \beta) &= -2\sum_{i=1}^{n} x_i(y_i - \alpha - \beta x_i).\end{aligned}$$

Setting the two partial derivatives to zero leads to

$$\begin{cases}\sum_{i=1}^{n}(y_i - \alpha - \beta x_i) = 0,\\ \sum_{i=1}^{n} x_i(y_i - \alpha - \beta x_i) = 0.\end{cases} \tag{2}$$

From the first equation in (2) we have that

$$\sum_{i=1}^{n} y_i - n\alpha - \beta \sum_{i=1}^{n} x_i = 0 \Rightarrow \alpha = \bar{y} - \beta \bar{x}.$$

Replacing α by $\bar{y} - \beta\bar{x}$ in the second equation in (2) leads to

$$\begin{aligned}
&\sum_{i=1}^{n} x_i y_i - (\bar{y} - \beta\bar{x}) n\bar{x} - \beta \sum_{i=1}^{n} x_i^2 = 0, \\
&\Rightarrow \sum_{i=1}^{n} x_i y_i - n\bar{x}\bar{y} - \beta \left(\sum_{i=1}^{n} x_i^2 - n\bar{x}^2 \right) = 0, \\
&\Rightarrow \hat{\beta} = \frac{\sum_{i=1}^{n} x_i y_i - n\bar{x}\bar{y}}{\sum_{i=1}^{n} x_i^2 - n\bar{x}^2}.
\end{aligned}$$

Also, $\hat{\alpha} = \bar{y} - \hat{\beta}\bar{x}$. Note that for this model the residuals are

$$\hat{\epsilon}_i = y_i - \hat{\alpha} - \hat{\beta} x_i.$$

The fact that $\sum_{i=1}^{n} \hat{\epsilon}_i = 0$ for this model follows from the first Equation in (2) and by definition of least squares estimates.

(c) For this model the residual sum of squares is

$$\mathcal{S}(\beta) = \sum_{i=1}^{n} (y_i - \beta x_i)^2,$$

which implies that

$$\frac{\mathrm{d}}{\mathrm{d}\beta} \mathcal{S}(\beta) = -2 \sum_{i=1}^{n} x_i (y_i - \beta x_i) = 0, \tag{3}$$

leading to

$$\hat{\beta} = \frac{\sum_{i=1}^{n} x_i y_i}{\sum_{i=1}^{n} x_i^2}.$$

The residuals in this model are given by $\hat{\epsilon}_i = y_i - \hat{\beta} x_i$. From Equation (3) we must have

$$\sum_{i=1}^{n} x_i \hat{\epsilon}_i = 0.$$

Clearly this does not imply that the residuals all sum to zero, although it does not say that they do not. A simple example suffices to demonstrate that they certainly do not always sum to zero. Let $x_1 = 0$, $x_2 = 1$ and $y_1 = y_2 = 1$. A simple calculation shows that for these data the sum of the residuals is 1.

(d) Generalising part (b), we have that for any model with an intercept, setting the derivative w.r.t. to the intercept to zero yields the equation

$$\sum_{i=1}^{n} \hat{\epsilon}_i = 0.$$

Without the intercept term, this does not happen. So, if the model has an intercept term then the residuals will always sum to zero. For models with no intercept term, this is generally not the case.

2. None of these. See section 4 of the notes where unbiasedness is proven, and you will see that none of the give assumptions is used in the proof. The only assumption we have used in the proof is that $\mathbb{E}(\boldsymbol{\epsilon}) = \mathbf{0}$, which implies that $\mathbb{E}(\mathbf{y}) = \mathbf{X}\boldsymbol{\beta}$.

3. We have that

$$\begin{aligned}\|\boldsymbol{\mathcal{Q}}^{\mathsf{T}}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})\|^2 = [\boldsymbol{\mathcal{Q}}^{\mathsf{T}}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})]^{\mathsf{T}}[\boldsymbol{\mathcal{Q}}^{\mathsf{T}}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})] &= (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})^{\mathsf{T}}\boldsymbol{\mathcal{Q}}\boldsymbol{\mathcal{Q}}^{\mathsf{T}}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) \\ &= (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})^{\mathsf{T}}\mathbf{I}_n(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) \\ &= \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|^2\end{aligned}$$

4. We know that the fitted value vector when we regress $\mathbf{y}$ on $\mathbf{X}$ is $\hat{\boldsymbol{\mu}}_{\mathbf{y}} = \mathbf{X}\hat{\beta}_{\mathbf{y}} = \mathbf{Q}\mathbf{Q}^{\mathsf{T}}\mathbf{y}$, where we are using the index $\mathbf{y}$ to emphasise that the response vector in this case is $\mathbf{y}$. Now, if $\hat{\boldsymbol{\epsilon}}$ is treated as the response vector and $\mathbf{X}$ as the design matrix, we have that the fitted value vector is

$$\begin{aligned}\hat{\boldsymbol{\mu}}_{\boldsymbol{\epsilon}} &= \mathbf{X}\hat{\beta}_{\boldsymbol{\epsilon}} \\ &= (\mathbf{Q}\mathbf{R})(\mathbf{R}^{-1}\mathbf{Q}^{\mathsf{T}}\hat{\epsilon}) \\ &= \mathbf{Q}\mathbf{Q}^{\mathsf{T}}\hat{\epsilon} \qquad (4)\end{aligned}$$

But note that

$$\hat{\epsilon} = \mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}_{\mathbf{y}} = \mathbf{y} - \mathbf{Q}\mathbf{Q}^{\mathsf{T}}\mathbf{y}$$

Returning to (4), we have that

$$\begin{aligned}\hat{\boldsymbol{\mu}}_{\boldsymbol{\epsilon}} &= \mathbf{Q}\mathbf{Q}^{\mathsf{T}}(\mathbf{y} - \mathbf{Q}\mathbf{Q}^{\mathsf{T}}\mathbf{y}) \\ &= \mathbf{Q}\mathbf{Q}^{\mathsf{T}}\mathbf{y} - \mathbf{Q}\mathbf{Q}^{\mathsf{T}}\mathbf{Q}\mathbf{Q}^{\mathsf{T}}\mathbf{y} \\ &= \mathbf{Q}\mathbf{Q}^{\mathsf{T}}\mathbf{y} - \mathbf{Q}\mathbf{Q}^{\mathsf{T}}\mathbf{y} \\ &= \mathbf{0}\end{aligned}$$

5. The first two results still hold (at least provided we do not change the definition of the sum of squares being minimized to find $\hat{\boldsymbol{\beta}}$), but the remaining results are no longer true. This is because these results rely on the elements of $\mathcal{Q}^{\mathsf{T}}\mathbf{y}$ being independent with constant variance, but they will be neither if the elements of $\boldsymbol{\epsilon}$, and consequently the elements of $\mathbf{y}$, do not have constant variance.

6. In section 4.3.1 of the notes we have derived that $\mathbf{r} \sim N(\mathbf{0}, \sigma^2\mathbf{I}_{n-p})$. Therefore we have that $\mathbb{E}(r_i) = 0$ and $\text{var}(r_i) = \sigma^2$, for $i = 1, \ldots, n-p$. Therefore, by definition of variance, we must have that, $\mathbb{E}(r_i^2) = \sigma^2$, for $i = 1, \ldots, n-p$. Let us workout now the expectation of $\hat{\sigma}^2$:

$$\begin{aligned}\mathbb{E}(\hat{\sigma}^2) = \mathbb{E}\left(\frac{\|\mathbf{r}\|^2}{n-p}\right) = \mathbb{E}\left(\frac{\sum_{i=1}^{n-p} r_i^2}{n-p}\right) &= \frac{1}{n-p}\sum_{i=1}^{n-p}\mathbb{E}(r_i^2) \\ &= \frac{1}{n-p}(n-p)\sigma^2 \\ &= \sigma^2\end{aligned}$$

7. We start by noting that we will be using the version of the dataset that removes observations 3 and 15 (although for the sake of the argument made in this exercise it does not really matter). In this case we would then be fitting the following model

$$y_i = \alpha + \beta x_i + \epsilon_i, \quad \epsilon_i \overset{\text{iid}}{\sim} N(0, \sigma^2),$$

where y_i is the recessional velocity of galaxy i and x_i is the distance to Earth of such galaxy, for $i = 1, \ldots, 22$ (because we are excluding the two outliers). Let us then fit this model and inspect the information provided by the call to the `summary`.

```
library(gamair)
data(hubble)
fit_int <- lm(y ~ x, data = hubble[-c(3, 15), ])
summary(fit_int)

##
## Call:
## lm(formula = y ~ x, data = hubble[-c(3, 15), ])
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -312.83 -140.91  -13.79  139.54  269.31
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -16.460     88.899  -0.185    0.855
## x             78.812      6.863  11.484 2.95e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 184.8 on 20 degrees of freedom
## Multiple R-squared:  0.8683,Adjusted R-squared:  0.8617
## F-statistic: 131.9 on 1 and 20 DF,  p-value: 2.946e-10
```

From the summary we see that that the p-value for the null hypothesis $H_0 : \alpha = 0$ (against the alternative $H_1 : \alpha \neq 0$) is very high (0.855) and so there is strong evidence *to not reject* H_0. Also, the 95% confidence interval for α is very wide.

```
beta_hat <- coef(fit_int)
sigma_beta_hat <- sqrt(diag(vcov(fit_int)))
beta_hat[1] + qt(c(0.025, 0.975),
                 df = (nrow(hubble[-c(3, 15),]) - 2))*sigma_beta_hat[1]

## [1] -201.8996  168.9791
```

We could have obtained this directly from the `confint` function.

```
confint(fit_int)[1,]

##     2.5 %    97.5 %
## -201.8996  168.9791
```

Note that because of the duality between hypothesis tests and confidence intervals, because this 95% confidence interval contains the value zero, we would have not rejected the null hypothesis $H_0 : \alpha = 0$ at a significance level of 0.05.

This dataset has a very small sample size. The standard error for the estimated parameter β in this model is quite inflated compared to the model that does not contain the intercept (shown below): 6.9 against 3.

```
fit <- lm(y ~ x - 1, data = hubble[-c(3, 15), ])
summary(fit)

##
## Call:
## lm(formula = y ~ x - 1, data = hubble[-c(3, 15), ])
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -304.3 -141.9  -26.5  138.3  269.8
##
## Coefficients:
##   Estimate Std. Error t value Pr(>|t|)
## x    77.67       2.97   26.15   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 180.5 on 21 degrees of freedom
## Multiple R-squared:  0.9702,Adjusted R-squared:  0.9688
## F-statistic: 683.8 on 1 and 21 DF,  p-value: < 2.2e-16
```

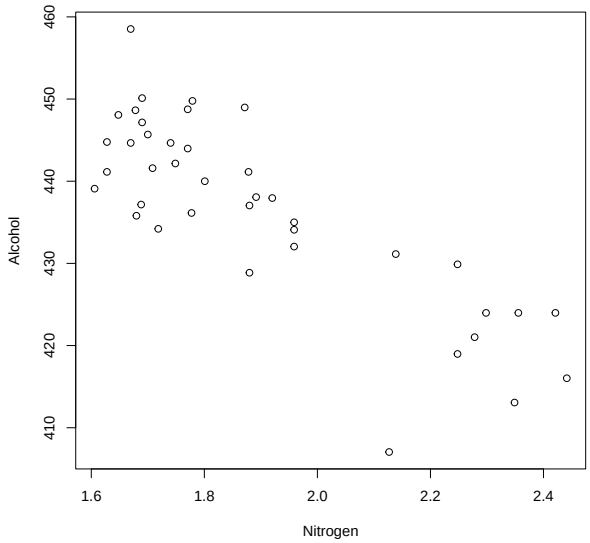
8. (a) Independent, with zero expected value and constant variance, σ^2. We also assume a normal distribution.

   (b) First we need to read the dataset in R.

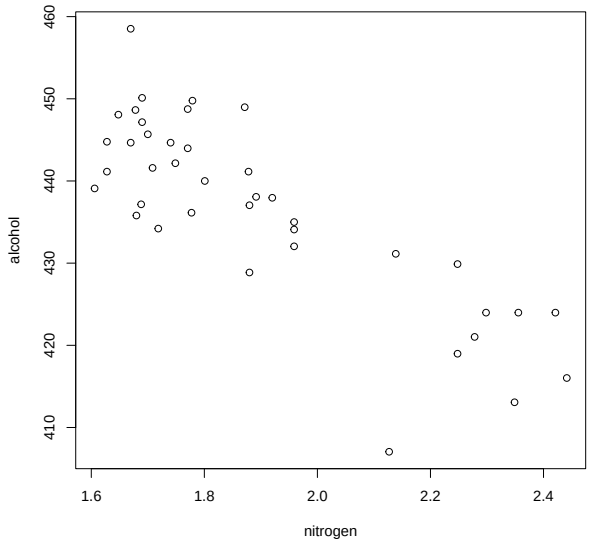
   ```
   grain <- read.table("grain.dat")
   ```

   There are many ways we can produce such a plot. I will show two. Nicer plots can be made in `ggplot2` but I prefer base R.

   ```
   # One way
   plot(grain$nitrogen, grain$alcohol,
        xlab = "Nitrogen", ylab = "Alcohol")
   ```




```
# Another way
with(grain, plot(nitrogen, alcohol))
```



(c) Let us start by fitting the model.

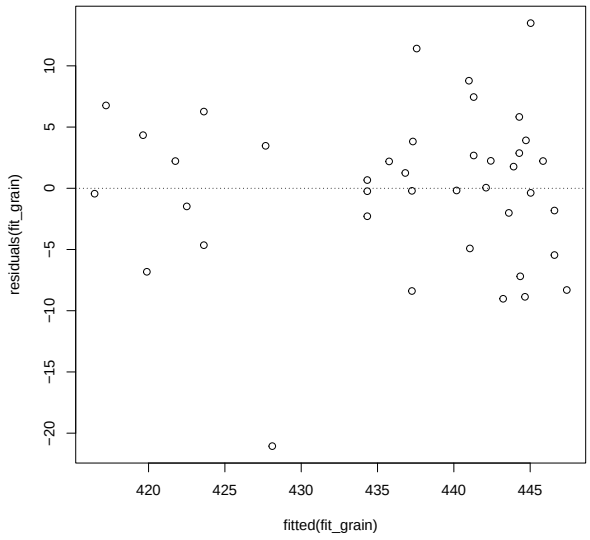
```
fit_grain <- lm(alcohol ~ nitrogen, data = grain)
summary(fit_grain)

##
## Call:
## lm(formula = alcohol ~ nitrogen, data = grain)
##
## Residuals:
```

```
##      Min       1Q   Median       3Q      Max
## -21.0588  -2.8736   0.3611   3.5560  13.4889
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  506.871      7.839  64.661  < 2e-16 ***
## nitrogen     -37.034      4.096  -9.041 5.21e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.482 on 38 degrees of freedom
## Multiple R-squared:  0.6826,Adjusted R-squared:  0.6743
## F-statistic: 81.74 on 1 and 38 DF,  p-value: 5.208e-11
```

We will see next week more about model checking and, more specifically, about residuals plots. Nonetheless, the most crucial check is the plot of the fitted values against the residuals and, in such plot, we should expect to see no patterns. Let us see if this is indeed the case.

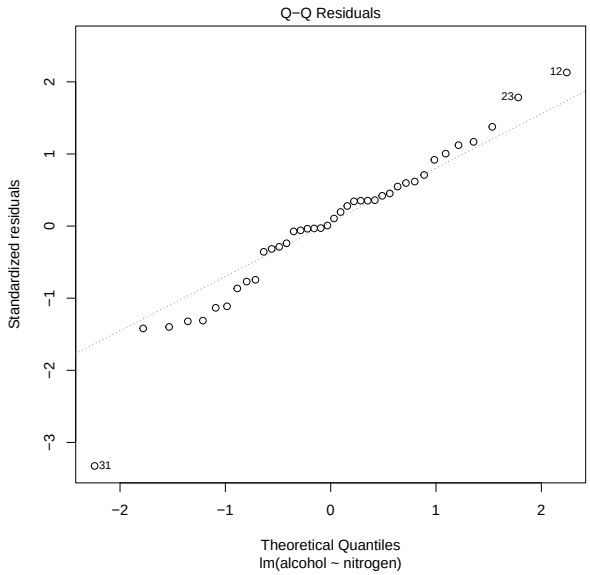
```
plot(fitted(fit_grain), residuals(fit_grain))
abline (h = 0, lty = 3)
```




```
#Alternative way of extracting fitted values and residuals
#plot(fit_grain$fitted.values,  fit_grain$residuals)
#abline (h = 0, lty = 3)
```

There is no trend in residual mean or variability. As we will see next week, the assumption of normality of the error terms is not as important as the other assumptions (due to the central limit theorem). However, because the sample size is quite small for this dataset, before proceeding, I feel more comfortable looking at a qqplot of the (standardised) residuals.

```r
plot(fit_grain, which = 2)
```



It is fine. As in the fitted values against residuals plot, we can observe that there is one observation which has a large, in absolute value, residual but this not necessarily means that it will affect our results (but it can!). More next week (when we will learn about the concept of *leverage*)

(d) The required 95% confidence interval for β is

```r
confint(fit_grain)

##                 2.5 %    97.5 %
## (Intercept) 491.00147 522.73969
## nitrogen    -45.32595 -28.74117

beta_hat <- coef(fit_grain)
sd_beta_hat <- sqrt(diag(vcov(fit_grain)))
beta_hat[2] + qt(c(0.025, 0.975), df = (nrow(grain) - 2))*sd_beta_hat[2]

## [1] -45.32595 -28.74117

#check with confint
confint(fit_grain)[2,]

##     2.5 %    97.5 %
## -45.32595 -28.74117
```

Since the interval is a long way from including zero, there is strong evidence that increasing nitrogen reduces alcohol yield (see also the p-value from the summary).
Just for the sake of curiosity, below is the code to find the estimated covariance matrix of $\hat{\boldsymbol{\beta}}$, $\hat{\mathbf{V}}_{\hat{\boldsymbol{\beta}}} = \mathbf{R}^{-1}\mathbf{R}^{-T}\hat{\sigma}^2$, using the QR decomposition directly.

```r
X <- model.matrix( ~ nitrogen, data = grain)
qrx <- qr(X)
```

```
R <- qr.R(qrx)
Q_orth <- qr.Q(qrx, complete = TRUE)
y <- as.matrix(grain$alcohol, ncol = 1)
p <- 2
n <- nrow(grain)
aux <- t(Q_orth)%*%y
r <- as.matrix(aux[(p + 1): n], ncol = 1)
sigma_hat <- (t(r)%*%r)/(n-p)
V.beta_alt <- solve(R)%*%t(solve(R))*as.numeric(sigma_hat)
V.beta_alt
```

```
##             (Intercept)  nitrogen
## (Intercept)    61.44905 -31.83450
## nitrogen      -31.83450  16.77912
```

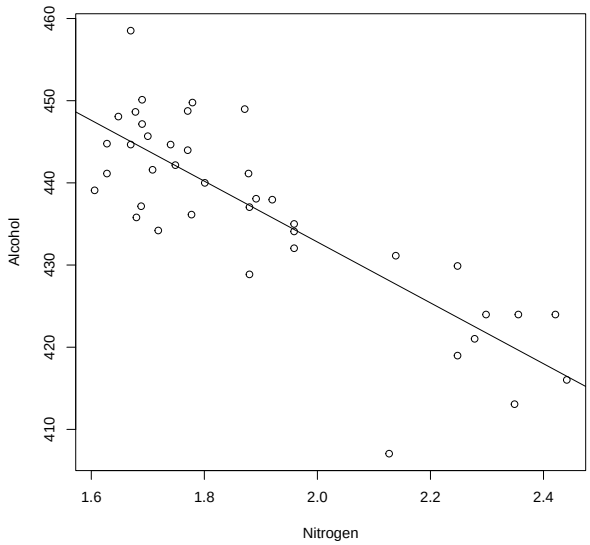
```
vcov(fit_grain)
```

```
##             (Intercept)  nitrogen
## (Intercept)    61.44905 -31.83450
## nitrogen      -31.83450  16.77912
```

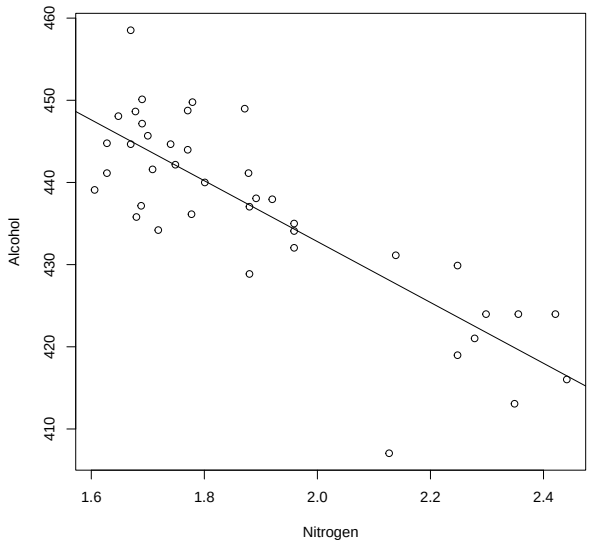
(e) The estimate of β is -37.034. The units of the nitrogen variable is in $\%$ (by weight). The interpretation of β is as follows: it is the expected change in alcohol yield (in Litres per Tonne) when nitrogen is increased by 1%. When nitrogen is increased by 0.1%, the expected alcohol yield changes by $-37.034 \times 0.1 = -3.7$, i.e., it drops by 3.7 Litres per Tonne.

(f) Again, there a few ways of doing it. I will show three.

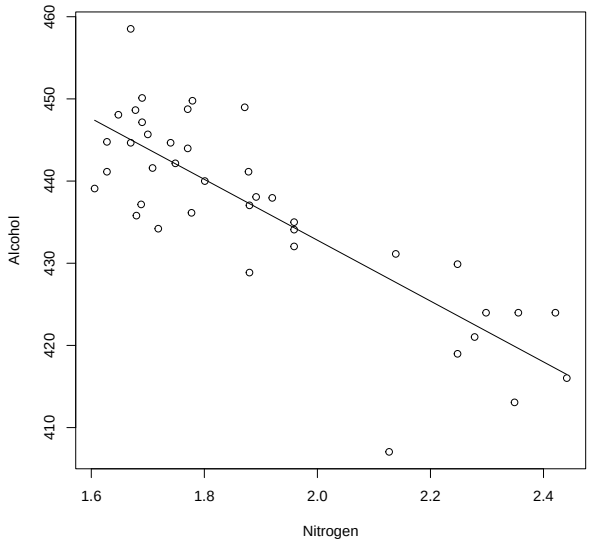
```
plot(grain$nitrogen, grain$alcohol,
     xlab = "Nitrogen", ylab = "Alcohol")
abline(fit_grain)
```



```r
plot(grain$nitrogen, grain$alcohol,
     xlab = "Nitrogen", ylab = "Alcohol")
abline(a = beta_hat[1], b = beta_hat[2])
```




```r
plot(grain$nitrogen, grain$alcohol,
     xlab = "Nitrogen", ylab = "Alcohol")
lines(sort(grain$nitrogen), beta_hat[1] + beta_hat[2]*sort(grain$nitrogen))
```



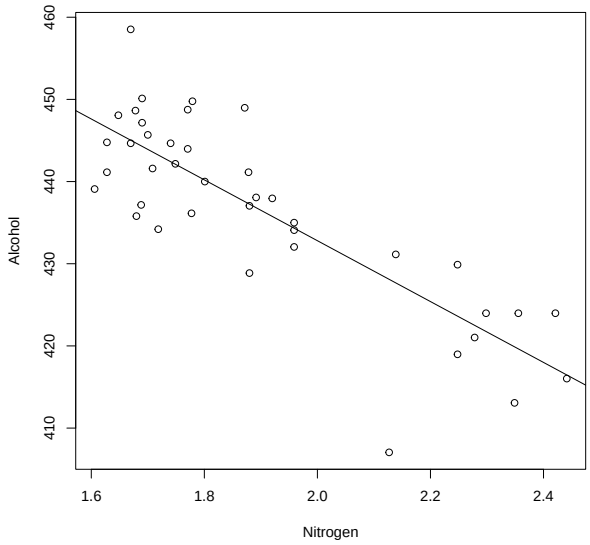
(g) *The following is suitable as the sort of brief, to the point, summary of an analysis that you might provide for a short statistical consultancy. Notice that all the important details of the analysis are there, and another statistician would have no problem reproducing what you have done, but the report also tries to use plain language, where ever possible. Also note that it is well short of a full*

report: it assumes that the reader knows the context and data already. We are not done with this example, and will return to it later (either in the lecture notes or in one workshop), once we have covered, even if briefly, confounding.

Data consisting of alcohol distillation yield and percentage nitrogen in the grain used for distillation were analysed to investigate the relationship, if any, of yield and nitrogen. This was done by using least squares to fit the model

$$\texttt{alcohol}_i = \alpha + \beta \texttt{nitrogen}_i + \epsilon_i$$

to the data, where α and β are parameters (estimated by fitting) and the ϵ_i are random error terms, assumed to have zero mean, constant variance, and to be independent of each other. Model fitting was performed in R, and residual checking plots were examined to check model plausibility, and that the assumptions about ϵ_i were not obviously violated. The checking plots looked good, so the model appears to be reasonable, as is also evident from the following figure.



The least squares estimated of β is -37.0, indicating that each 1% rise in percentage nitrogen content would be expected to produce a drop of 37 units of alcohol yield. Standard linear model theory was also used to compute a 95% confidence interval for β. This is approximately (-45,-29). The interval is well away from zero, indicating that the relationship between alcohol and yield is highly statistically significant (i.e. it is highly implausible that $\hat{\beta}$ could be as low as -37, if there were really no relationship between yield and nitrogen).

9. (a) Running the code with this seed leads to the following results.

```
## [1] 0.940 0.953 0.953
```

Of course random error (often termed Monte Carlo error) will mean that by changing the seed, the empirical coverage probabilities for the three parameters may be a little different, but the key point is that the coverages are close to the nominal level of 0.95 (we computed 95% confidence intervals.

(b) Let us adapt the code and check the empirical coverages we obtain.

```r
n <- 100 # sample size
b.true <- c(0.5, 1, 10)
ct <- qt(.975, df = n-3)
cp <- b.true*0
n.rep <- 1000
set.seed(1)
for (i in 1:n.rep) {
  x <- runif(n)
  mu <- b.true[1] + b.true[2]*x + b.true[3]*x^2
  y <- rpois(n,mu)
  m1 <- lm(y ~ x + I(x^2))
  b <- coef(m1)
  sig.b <- sqrt(diag(vcov(m1)))
  cp <- cp + as.numeric(b - ct*sig.b <= b.true &
                        b + ct*sig.b >= b.true)
}
cp/n.rep

## [1] 1.000 0.975 0.939
```

So now the coverages are far from nominal (1 is really very far from 0.95, for example). This happens because a Poisson random variable does not have constant variance. The variance increases with the mean, which undermines the theoretical variance calculation for the model coefficients. Of course, the errors are also not normally distributed but because the sample size is already decent, the QQ plot does not look terribly wrong. For instance, let us check it for the last generated dataset and corresponding model fit from the previous simulation.

```r
library(ecostats)
qqenvelope(m1)
```