MAST90104: Introduction to Statistical Learning

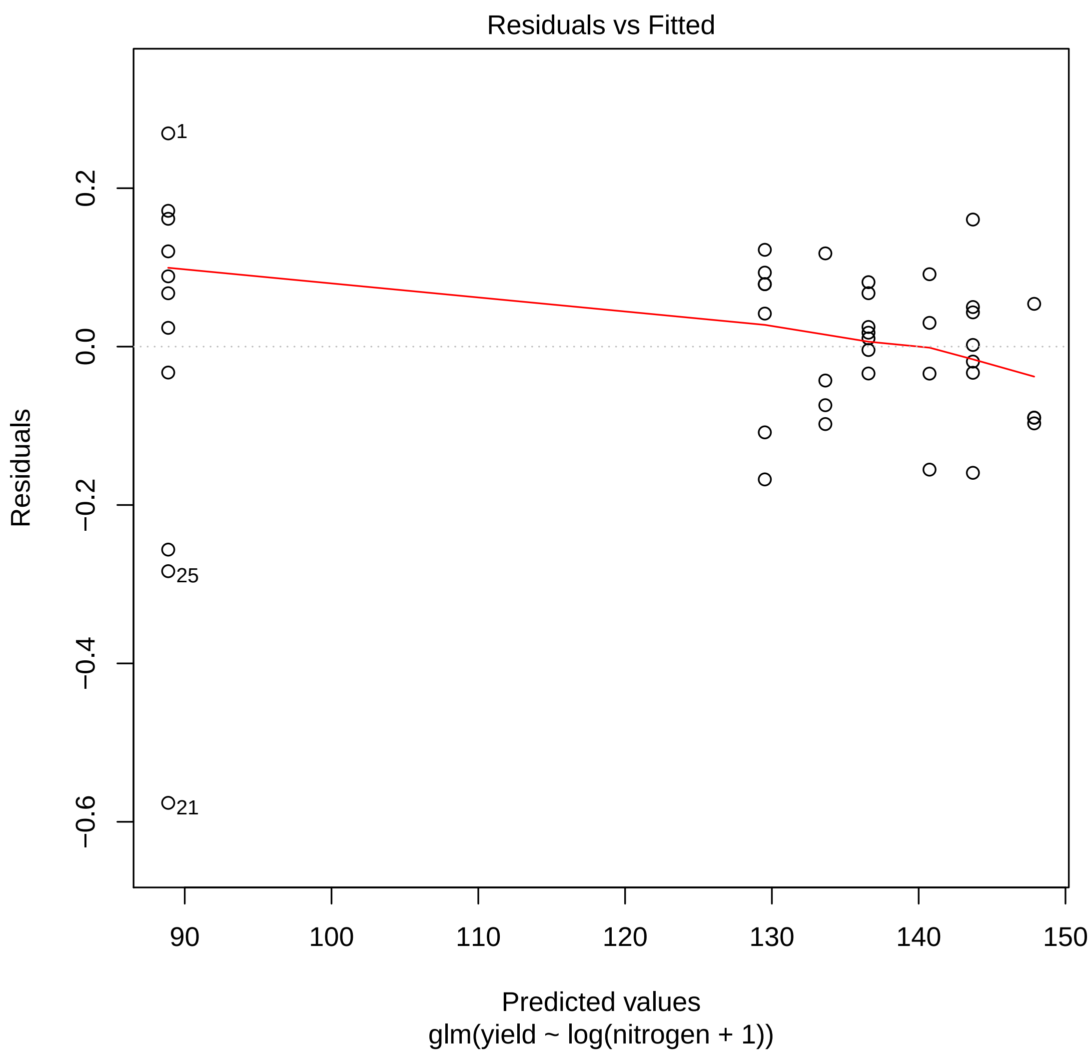
Week 10 Lab and Workshop Solutions

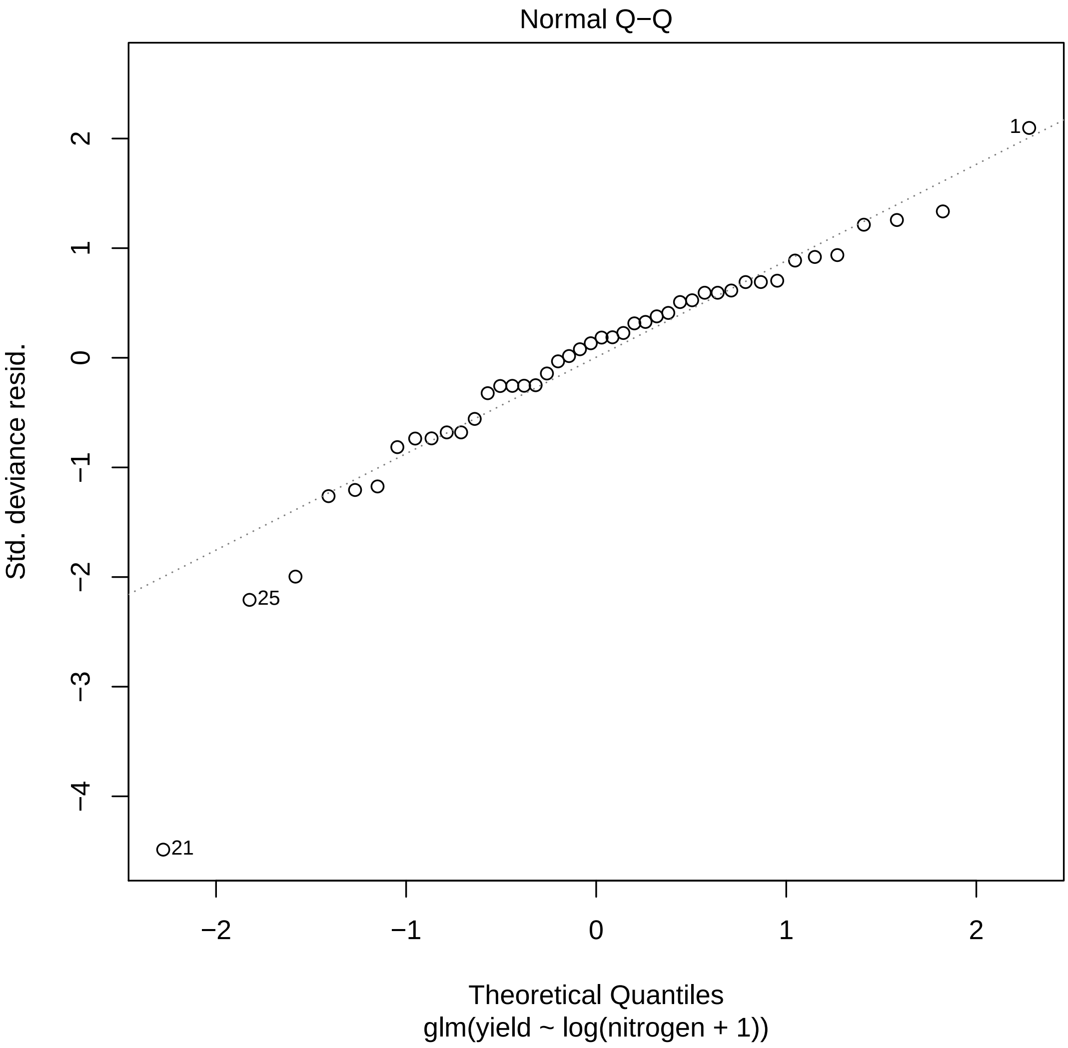
1 Lab

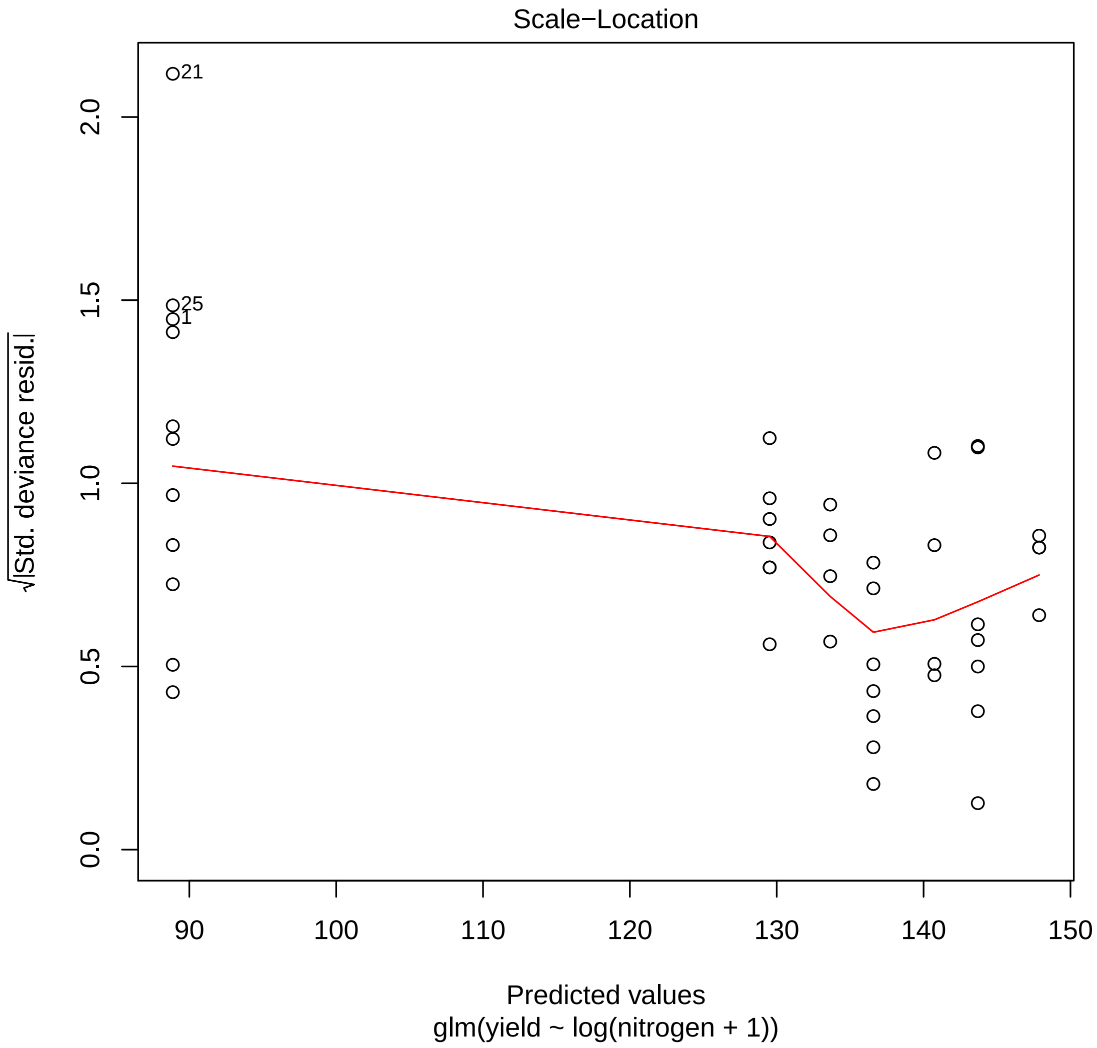
1. In question 5 in Lab sheet 9, re-do the gamma and linear model diagnostic plots with the standard R diagnostic plots and comment.

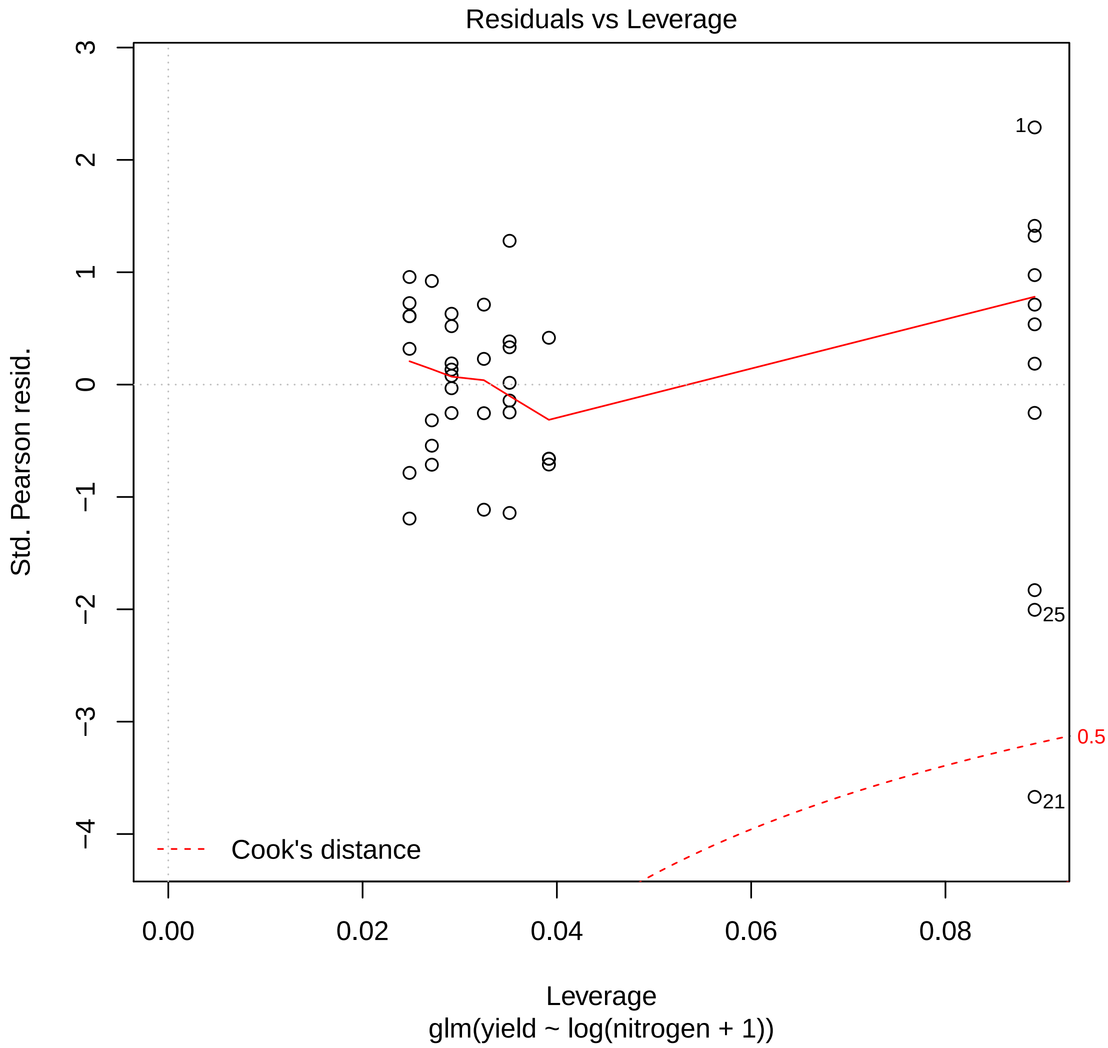
   Solution:

```
library(faraway)
gmod3 <- glm(yield ~ log(nitrogen+1), data=cornnit, family=Gamma(link="identity"))
plot(gmod3)
```

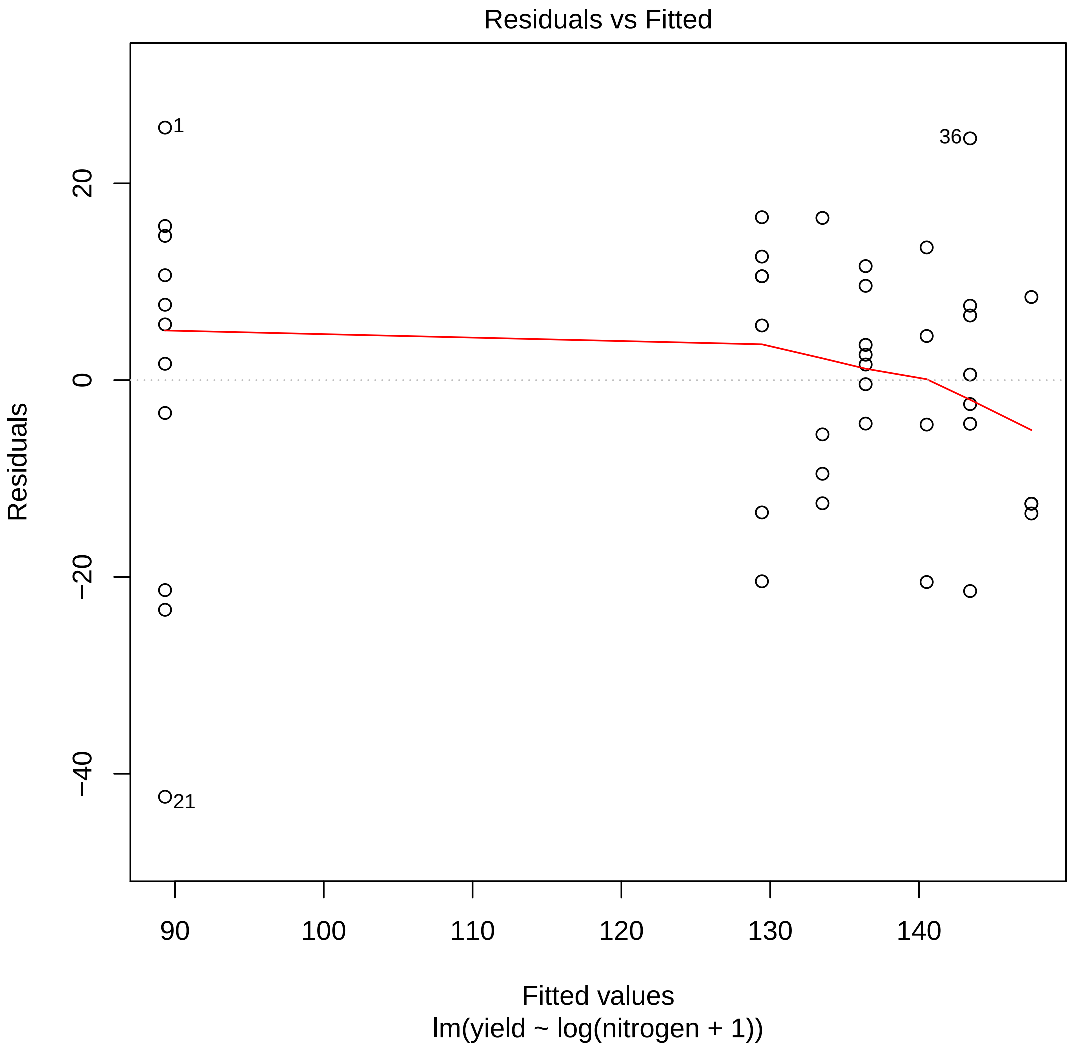


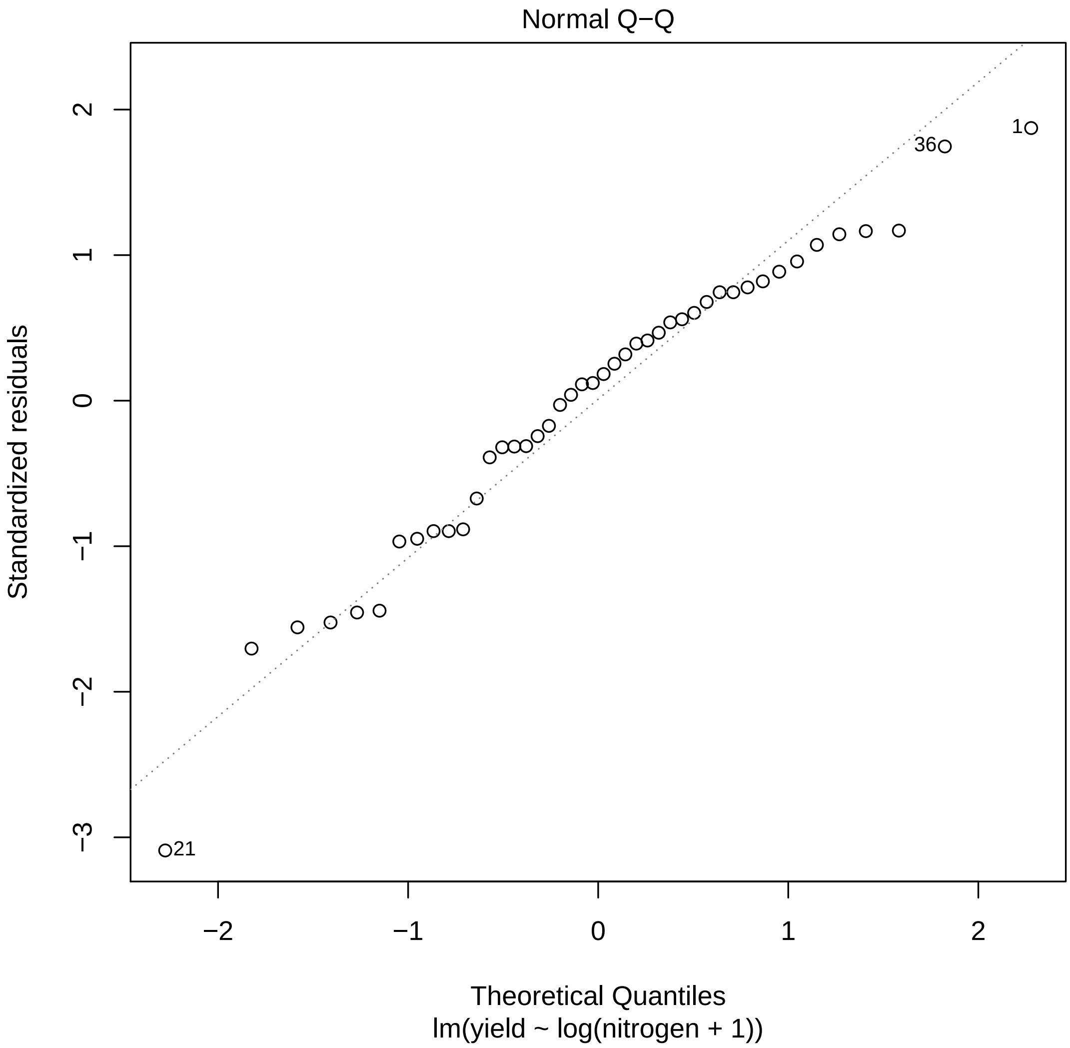

Normal Q−Q
Std. deviance resid.
Theoretical Quantiles
glm(yield ~ log(nitrogen + 1))
2
1
0
−1
−2
−3
−4
−2
−1
0
1
2
1
25
21

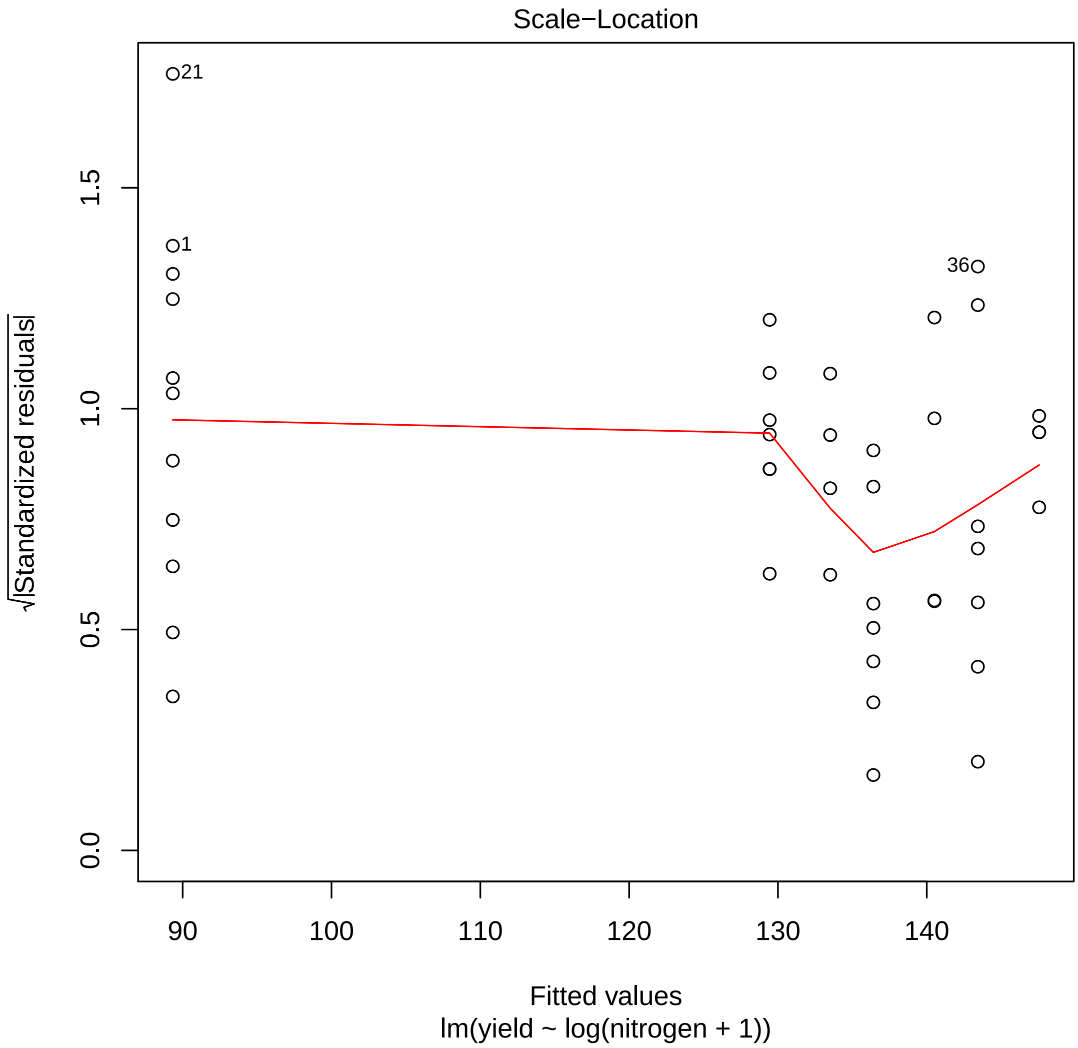

Scale−Location
21
25
1
2.0
1.5
1.0
0.5
0.0
√|Std. deviance resid.|
90
100
110
120
130
140
150
Predicted values
glm(yield ~ log(nitrogen + 1))

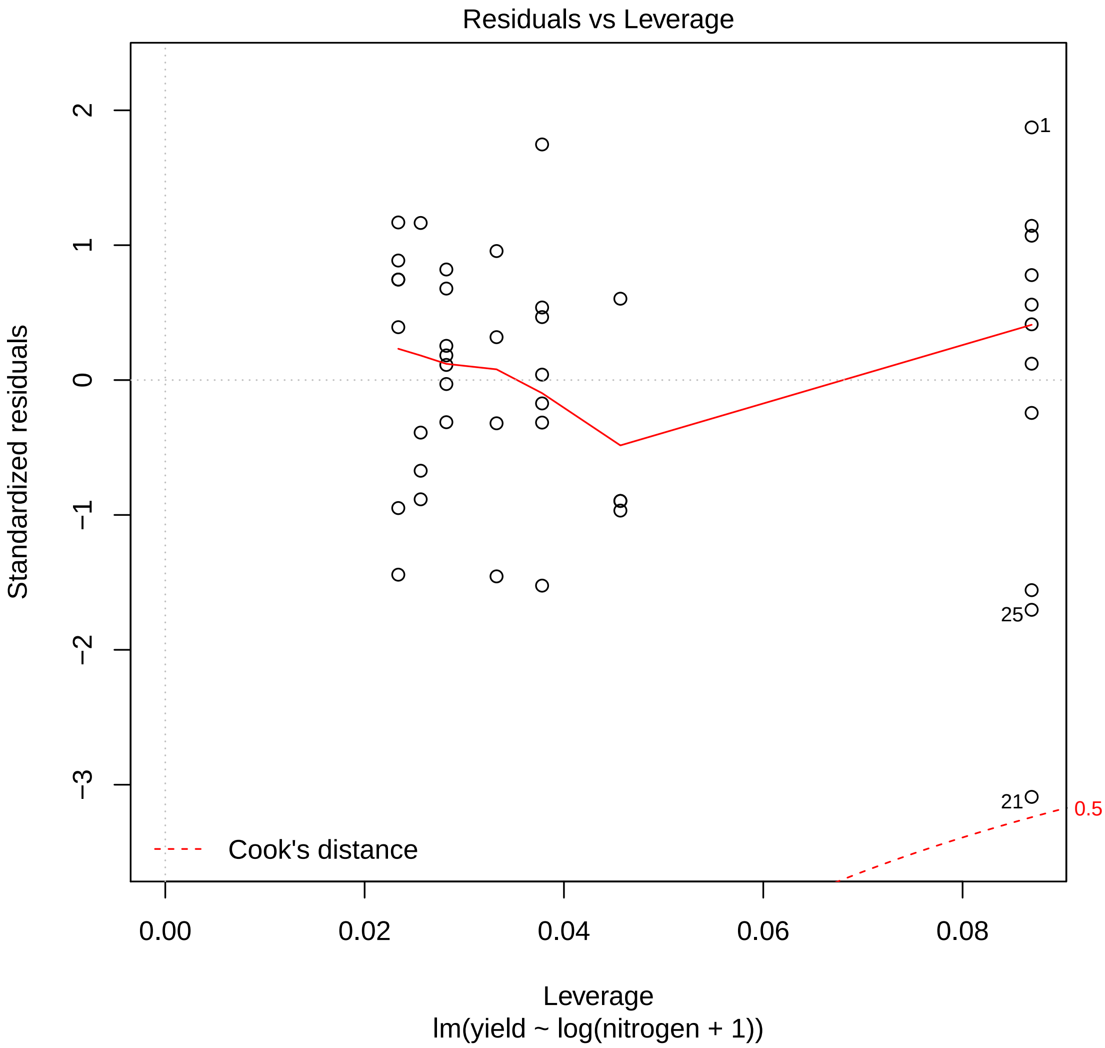



```
gmod4 <- lm(yield ~ log(nitrogen+1), data=cornnit)
plot(gmod4)
```


Residuals vs Fitted
Residuals
20
0
−20
−40
1
36
21
90
100
110
120
130
140
Fitted values
lm(yield ~ log(nitrogen + 1))


Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(yield ~ log(nitrogen + 1))
2
1
0
−1
−2
−3
−2
−1
0
1
2
36
1
21


Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
90
100
110
120
130
140
21
1
36
Fitted values
lm(yield ~ log(nitrogen + 1))



The plots are similar and the lack of change in the variance with the mean is clear from both the deviance and standardized residuals, confirming the normal model is appropriate. The residuals do seem to be approximately normally distributed.

2. In the `multinom` function from the `nnet` package, the response should be a factor with K levels or a matrix with K columns, which will be interpreted as counts for each of K classes. The first case is a short hand for responses of the form multinomial(1, $\mathbf{p}$).

   The `hsb` data from the `faraway` package was collected as a subset of the "High School and Beyond" study, conducted by the National Education Longitudinal Studies program of the U.K. National Center for Education Statistics. The variables are gender; race; socioeconomic status; school type; chosen high school program type; scores on reading, writing, math, science, and social studies. We want to determine which factors are related to the choice of the type of program—academic, vocational, or general—that the students pursue in high school. The response is multinomial with three levels.

   (a) Fit a trinomial response model with the other relevant variables as predictors (untransformed).
   Solution:

```
library(faraway)
data(hsb)
library(nnet)
```

```r
mmod <- multinom(prog ~ gender + race + ses + schtyp + read + write + math +
                   science + socst, hsb, trace = FALSE)
summary(mmod)
```

```
## Call:
## multinom(formula = prog ~ gender + race + ses + schtyp + read +
##     write + math + science + socst, data = hsb, trace = FALSE)
##
## Coefficients:
##          (Intercept)  gendermale raceasian racehispanic racewhite
## general     3.631901 -0.09264717  1.352739   -0.6322019 0.2965156
## vocation    7.481381 -0.32104341 -0.700070   -0.1993556 0.3358881
##               seslow sesmiddle schtyppublic        read       write
## general  1.09864111 0.7029621    0.5845405 -0.04418353 -0.03627381
## vocation 0.04747323 1.1815808    2.0553336 -0.03481202 -0.03166001
##                math    science       socst
## general  -0.1092888 0.10193746 -0.01976995
## vocation -0.1139877 0.05229938 -0.08040129
##
## Std. Errors:
##          (Intercept) gendermale raceasian racehispanic racewhite    seslow
## general     1.823452  0.4548778  1.058754    0.8935504 0.7354829 0.6066763
## vocation    2.104698  0.5021132  1.470176    0.8393676 0.7480573 0.7045772
##          sesmiddle schtyppublic       read      write       math
## general  0.5045938    0.5642925 0.03103707 0.03381324 0.03522441
## vocation 0.5700833    0.8348229 0.03422409 0.03585729 0.03885131
##             science      socst
## general  0.03274038 0.02712589
## vocation 0.03424763 0.02938212
##
## Residual Deviance: 305.8705
## AIC: 357.8705
```

(b) Use either backward elimination with χ^2 tests (using the `anova` command), or the AIC (using `step`), to produce a parsimonious model. Give an interpretation of the resulting model.
Solution: I just used the AIC, as provided by `step`.

```r
mmod2 <- step(mmod, scope=~., direction="backward", trace = FALSE)
```

```
## trying - gender
## trying - race
## trying - ses
## trying - schtyp
## trying - read
## trying - write
## trying - math
## trying - science
## trying - socst
## trying - gender
## trying - ses
## trying - schtyp
## trying - read
## trying - write
## trying - math
## trying - science
## trying - socst
## trying - ses
## trying - schtyp
```

```
## trying - read
## trying - write
## trying - math
## trying - science
## trying - socst
## trying - ses
## trying - schtyp
## trying - read
## trying - math
## trying - science
## trying - socst
## trying - ses
## trying - schtyp
## trying - math
## trying - science
## trying - socst
```

```
summary(mmod2)
```

```
## Call:
## multinom(formula = prog ~ ses + schtyp + math + science + socst,
##     data = hsb, trace = FALSE)
##
## Coefficients:
##          (Intercept)      seslow sesmiddle schtyppublic       math
## general     2.587029  0.87607389 0.6978995    0.6468812 -0.1212242
## vocation    6.687272 -0.01569301 1.2065000    1.9955504 -0.1369641
##              science       socst
## general  0.08209791 -0.04441228
## vocation 0.03941237 -0.09363417
##
## Std. Errors:
##          (Intercept)    seslow sesmiddle schtyppublic       math
## general     1.686492 0.5758781 0.4930330     0.545598 0.03213345
## vocation    1.945363 0.6690861 0.5571202     0.812881 0.03591701
##              science      socst
## general  0.02787694 0.02344856
## vocation 0.02864929 0.02586717
##
## Residual Deviance: 315.5511
## AIC: 343.5511
```

Compared to students from a high socioeconomic class, students from a low socioeconomic class are more likely to choose a general high school program, while students from a middle socioeconomic class are more likely to choose a general program but even more likely to choose a vocational program. It is interesting that students from a low socioeconimic class do not show more of an interest in vocational programs.

Students from public schools are are more likely to choose a general program and much more likely to choose a vocational program, than students from private schools.

High scores in maths and social sciences indicate a higher chance of choosing an academic program, while (curiously) high scores in science indicate a lower chance of choosing an academic program.

If you wish to use a chisquared test instead of the AIC, then you will have to separately fit all the candidate models, and then compare them using `anova`. For example:

```
mmodXgender <- multinom(prog ~ race + ses + schtyp + read + write + math +
                   science + socst, hsb, trace = FALSE)
anova(mmod, mmodXgender)
```

```
## Likelihood ratio tests of Multinomial Models
##
## Response: prog
##                                                                   Model
## 1          race + ses + schtyp + read + write + math + science + socst
## 2 gender + race + ses + schtyp + read + write + math + science + socst
##   Resid. df Resid. Dev   Test    Df LR stat.   Pr(Chi)
## 1       376   306.2857
## 2       374   305.8705 1 vs 2     2 0.415142 0.8125556
```

Clearly considering all possible variables to drop will take some time.

(c) For the student with id 99, compute the predicted probabilities of the three possible choices.
Solution:

```
hsb[hsb$id==99,]

##     id gender  race  ses schtyp    prog read write math science socst
## 102 99 female white high public general   47    59   56      66    61

predict(mmod2, newdata = hsb[hsb$id==99,], type="probs")

##   academic    general   vocation
## 0.64426309 0.27665609 0.07908082
```

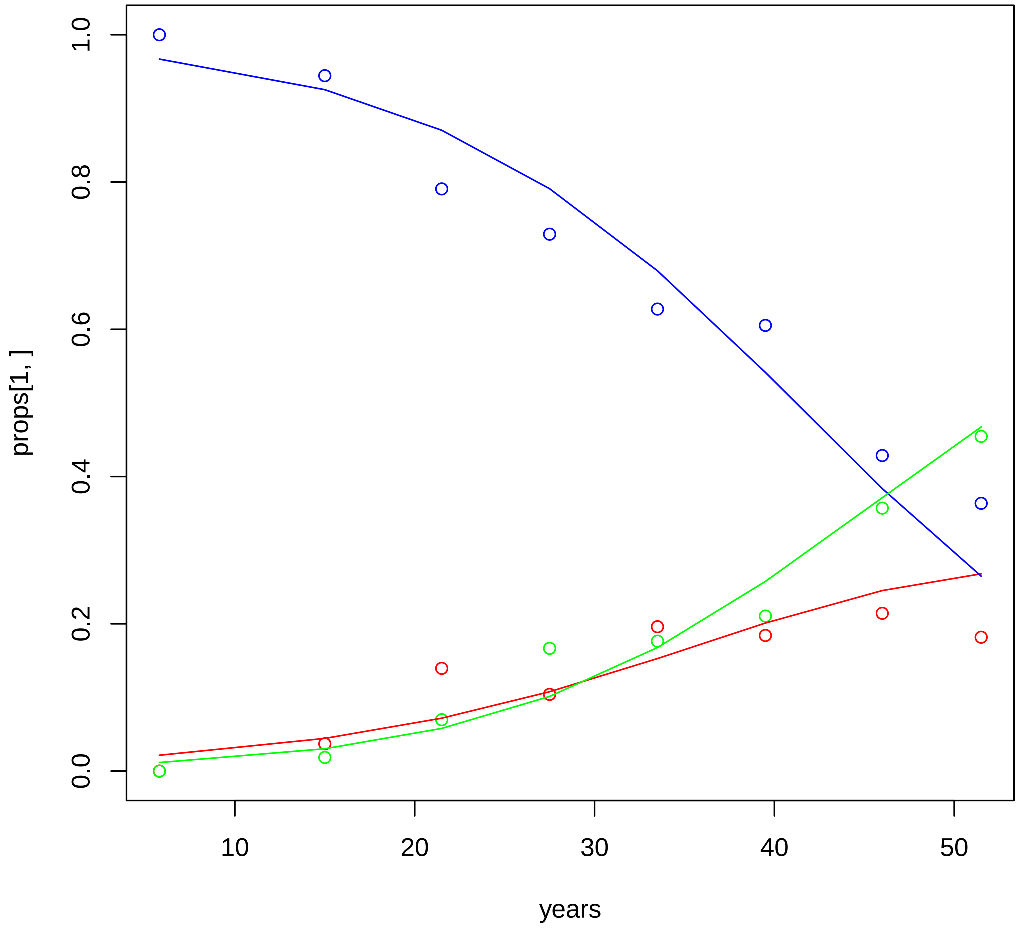
3. The `pneumo` data from the `faraway` package gives the number of coal miners classified by radiological examination into one of three categories of pneumonoconiosis and by the number of years spent working at the coal face divided into eight categories.

(a) Treating the pneumonoconiosis status as response variable as nominal, build a model for predicting the frequency of the three outcomes in terms of length of service and use it to predict the outcome for a miner with 25 years of service.
Solution: First we have a look at the data. Then the data needs to be reformatted before we can use the `multinom` function to fit a model. The fit looks quite good.

```
data(pneumo)
counts <- xtabs(Freq ~ status + year, pneumo)
(props <- prop.table(counts, 2))

##         year
## status          5.8         15       21.5       27.5       33.5       39.5
##   mild   0.00000000 0.03703704 0.13953488 0.10416667 0.19607843 0.18421053
##   normal 1.00000000 0.94444444 0.79069767 0.72916667 0.62745098 0.60526316
##   severe 0.00000000 0.01851852 0.06976744 0.16666667 0.17647059 0.21052632
##         year
## status           46       51.5
##   mild   0.21428571 0.18181818
##   normal 0.42857143 0.36363636
##   severe 0.35714286 0.45454545

years <- c(5.8, 15, 21.5, 27.5, 33.5, 39.5, 46, 51.5)
par(mfrow=c(1,1))
plot(years, props[1,], col="red", ylim=c(0,1))
points(years, props[2,], col="blue")
points(years, props[3,], col="green")

mmod <- multinom(t(counts) ~ years, trace=FALSE)
summary(mmod)

## Call:
## multinom(formula = t(counts) ~ years, trace = FALSE)
```

```
##
## Coefficients:
##          (Intercept)       years
## normal   4.2916723 -0.08356506
## severe  -0.7681706  0.02572027
##
## Std. Errors:
##          (Intercept)      years
## normal   0.5214110 0.01528044
## severe   0.7377192 0.01976662
##
## Residual Deviance: 417.4496
## AIC: 425.4496
```

```
fitted <- predict(mmod, newdata=list(year=years), type="probs")
lines(years, fitted[,1], col="red")
lines(years, fitted[,2], col="blue")
lines(years, fitted[,3], col="green")
```



For a miner with 25 year down pit we have the following fitted probabilities

```
predict(mmod, newdata=list(years=25), type="probs")

##        mild      normal      severe
```

```
## 0.09148821 0.82778696 0.08072483
```

In the model above we had eight multinomial observations, with the numer of trials equal to 98, 54, 43, 48, 51, 38, 28, 11. Each of these multinomials can be regarded as the sum of a number of independent multinomials each based on a single trial (just as a binomial is a sum of independent Bernoulli random variables). If we treat the data this way and fit a multinomial logistic regression, we get the same model, but what happens to the deviance degrees of freedom?

```
pneumo2 <- data.frame(status = rep(pneumo$status, pneumo$Freq),
                      year = rep(pneumo$year, pneumo$Freq))
mmod2 <- multinom(status ~ year, data = pneumo2, trace = FALSE)
summary(mmod2)
```

```
## Call:
## multinom(formula = status ~ year, data = pneumo2, trace = FALSE)
##
## Coefficients:
##        (Intercept)        year
## normal   4.2916723 -0.08356506
## severe  -0.7681706  0.02572027
##
## Std. Errors:
##        (Intercept)       year
## normal   0.5214110 0.01528044
## severe   0.7377192 0.01976662
##
## Residual Deviance: 417.4496
## AIC: 425.4496
```

(b) Repeat the analysis with the pneumonoconiosis status being treated as ordinal.

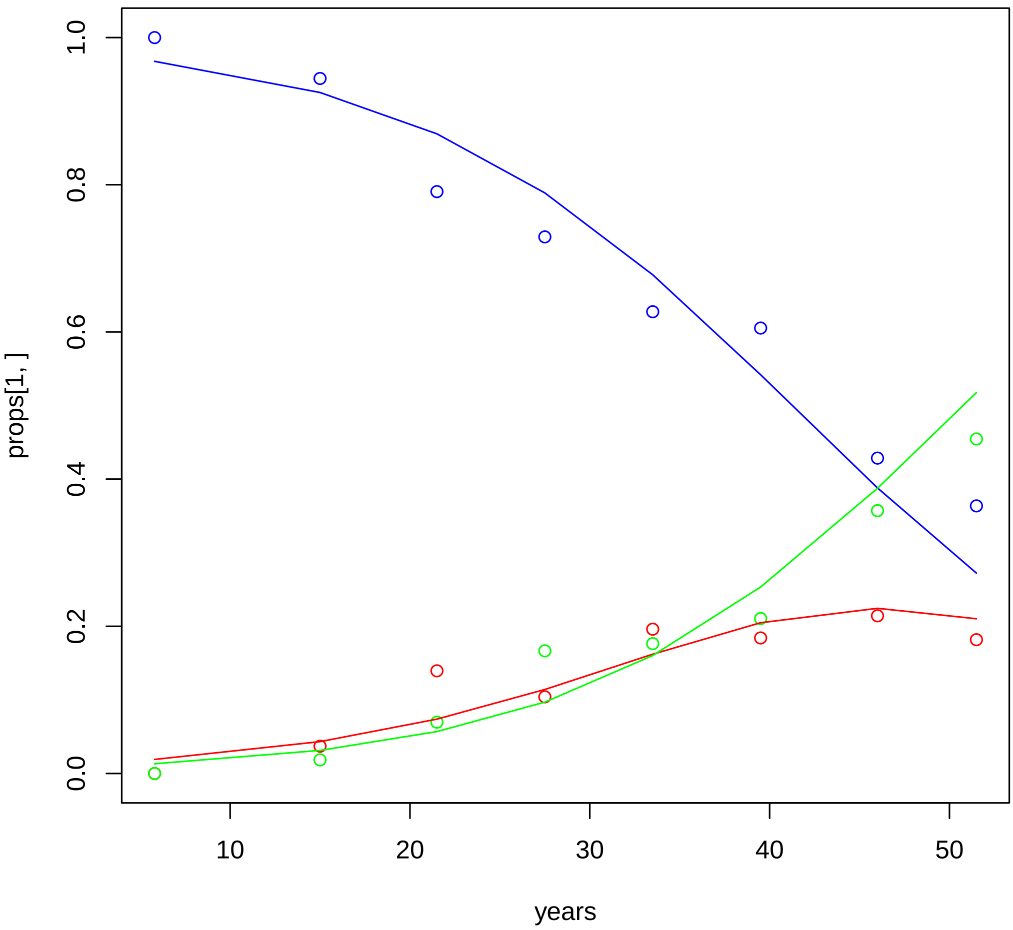
Solution:

First we convert `status` into an ordered factor (take care to get the order correct), then use the `polr` function. The fit looks good, and the AIC for this model is slightly smaller than that for the multinomial logistic regression model, so we prefer it.

```
pneumo2$status <- ordered(pneumo2$status, levels=c("normal", "mild", "severe"))
library(MASS)
omod <- polr(status ~ year, pneumo2)
summary(omod)
```

```
##
## Re-fitting to get Hessian

## Call:
## polr(formula = status ~ year, data = pneumo2)
##
## Coefficients:
##       Value Std. Error t value
## year 0.0959    0.01194   8.034
##
## Intercepts:
##             Value   Std. Error t value
## normal|mild  3.9558  0.4097     9.6558
## mild|severe  4.8690  0.4411    11.0383
##
## Residual Deviance: 416.9188
## AIC: 422.9188
```

```r
plot(years, props[1,], col="red", ylim=c(0,1))
points(years, props[2,], col="blue")
points(years, props[3,], col="green")
fitted <- predict(omod, newdata=list(year=years), type="probs")
lines(years, fitted[,1], col="blue")
lines(years, fitted[,2], col="red")
lines(years, fitted[,3], col="green")
```



For a miner with 25 years exposure we have the following fitted probabilities

```r
predict(omod, newdata=list(year=25), type="probs")

##     normal       mild     severe
## 0.82610096 0.09601474 0.07788430
```

2 Workshop

4. Suppose $Y_i, i = 1, \cdots, n$ are from a generalised linear model so they are independent from an exponential family:

$$f(y; \theta, \phi) = \exp\left[\frac{y\theta - b(\theta)}{a(\phi)} + c(y, \phi)\right]$$

with the parameter ϕ constant and supposed known but θ_i varies. Recall that

$$\mu = \mathbb{E}Y = b'(\theta)$$
$$var(\mu) = \operatorname{Var} Y = b''(\theta)a(\phi)$$
$$var = b'' \circ (b')^{-1} a(\phi)$$

and that there is a link function, g, so that $g(\mu_i) = \mathbf{x}_i^T \boldsymbol{\beta}$ where $\boldsymbol{\beta}$ are the parameters of interest, $\mu_i = \mathbb{E}Y_i$ and $\mathbf{x}_i$ is a vector of explanatory variables (this is the ith row of the predictor matrix X). In answering the questions below, you will establish that the Newton-Raphson method with Fisher scoring is the same as the iteratively weighted least squares algorithm introduced in lectures.

(a) Write down the log likelihood as a function of $\boldsymbol{\beta}$ and show that its derivative, $U(\beta_j)$, with respect to β_j may be written as:

$$\sum_{i=1}^{n} \frac{y_i - \mu_i}{var(\mu_i)} \frac{x_{ij}}{g'(\mu_i)}.$$

Solution: The log likelihood is

$$\sum_{i=1}^{n} \log f(y_i; \theta_i, \phi) = \sum_{i=1}^{n} \left[\frac{y_i\theta_i - b(\theta_i)}{a(\phi)} + c(y_i, \phi) \right]$$

so the derivative with respect to β_j is

$$\frac{\partial l(\boldsymbol{\beta}; \mathbf{y})}{\partial \beta_j} = \sum_{i=1}^{n} \frac{y_i - b'(\theta_i)}{a(\phi)} \frac{\partial \theta_i}{\partial \beta_j}. \tag{1}$$

Writing $\eta_i = \mathbf{x}_i^T \boldsymbol{\beta}$, $g(\mu_i) = \eta_i$ and $\mu_i = b'(\theta_i)$ so $\theta_i = b'^{-}1(\mu_i) = b'^{-}1(g^{-1}(\eta_i))$. Since $x = f^{-1}(y) \implies (f^{-1})'(y) = \frac{1}{f'(x)}$ applying the chain rule for differentiation twice gives

$$\frac{\partial \theta_i}{\partial \beta_j} = \frac{1}{b''(\mu_i)} \frac{1}{g'(\mu_i)} x_{ij}.$$

Using the preamble formulae for mean and variance in equation 1 gives the required derivative of the log likelihood.

(b) Hence show that

$$Cov(U(\beta_j)U(\beta_k)) = \sum_{i=1}^{n} \frac{x_{ij}x_{ik}}{var(\mu_i)(g'(\mu_i))^2}.$$

Solution: The covariance is the expected value of the product $U(\beta_j)U(\beta_k)$ since both random variables have zero mean from Workshop 9 Question 8.

The terms in the multiplication of two sums are the sum of all the products. Both $U(\beta_j)$ and $U(\beta_k)$ are sums with n terms each. The terms where the indices in the sums are different all have expectation 0 since they are the expected value of a product of independent random variables each with zero mean. Hence

$$Cov(U(\beta_j)U(\beta_k)) = \sum_{i=1}^{n} E\left(\frac{(y_i - \mu_i)^2}{var^2(\mu_i)} \frac{x_{ij}x_{ik}}{(g'(\mu_i))^2} \right)$$

giving the required equation since $v(\mu_i) = E((y_i - \mu_i)^2)$.

(c) Find the Fisher information and show that it is $X^T W(\boldsymbol{\beta}) X$ where $W(\boldsymbol{\beta})$ is a diagonal matrix whose ith diagonal entry is

$$\frac{1}{var(\mu_i)(g'(\mu_i))^2}.$$

Solution: The Fisher information matrix is defined to the matrix whose entries are $Cov(U(\beta_j)U(\beta_k)), j, k = 1, \cdots, n$.

If D is a diagonal matrix with diagonal entries $d_i, i = 1, \cdots, n$, then the jth row of X^T is the row vector with entries $x_{ij}d_i, i = 1, \cdots, n$. Taking the dot product of this vector with the kth column of X gives $\sum_{i=1}^{n} x_{ij}x_{ik}d_i$. This expression is the (j, k) entry of $X^T DX$.

Taking D to be the diagonal matrix $W(\boldsymbol{\beta})$ and using part (b) gives the required expression for the Fisher information matrix.

(d) Hence show that the Newton-Raphson iteration step with expected information replacing the Hessian can be expressed as

$$\boldsymbol{\beta}(m+1) = \boldsymbol{\beta}(m) + (X^T W(\boldsymbol{\beta}(m))X)^{-1} U(\boldsymbol{\beta}(m)).$$

Solution: The Newton-Raphson iteration step from p. 7 and 8 of Module 8, but expressed with the current notation is:

$$\boldsymbol{\beta}(m+1) = \boldsymbol{\beta}(m) - H(m)^{-1} U(\boldsymbol{\beta}(m))$$

where $H(m)$ is the matrix of second derivatives of the log likelihood evaluated at the current estimate $\boldsymbol{\beta}(m)$.

By Workshop 9 Question 8, the expected value of $-H(m)$ is the Fisher information.

Hence replacing $-H(m)$ by its expectation produces the algorithm in this part.

(e) Hence show that the Newton-Raphson method with Fisher scoring is the same as the iteratively weighted least squares algorithm in lectures (note that there is some confusion in notation with the iterative step in lectures being labelled n whereas here it is m because n refers to the number of observations.)

Solution: The vector $U(\boldsymbol{\beta})$ can be written as $X^T W(\boldsymbol{\beta})\tilde{y}$ where $\tilde{y}_i = (y_i - \mu_i)g'(\mu_i), i = 1, \cdots, n$.

Hence, the Newton-Raphson step with expected information replacing observed information, following part (d), is

$$\begin{aligned}
\boldsymbol{\beta}(m+1) &= \boldsymbol{\beta}(m) + (X^T W(\boldsymbol{\beta}(m))X)^{-1} U(\boldsymbol{\beta}(m)) \\
&= \boldsymbol{\beta}(m) + (X^T W(\boldsymbol{\beta}(m))X)^{-1} X^T W(\boldsymbol{\beta}(m))\tilde{y} \\
&= (X^T W(\boldsymbol{\beta}(m))X)^{-1} X^T W(\boldsymbol{\beta}(m))X\boldsymbol{\beta}(m) + (X^T W(\boldsymbol{\beta}(m))X)^{-1} X^T W(\boldsymbol{\beta})\tilde{y} \\
&= (X^T W(\boldsymbol{\beta}(m))X)^{-1} X^T W(\boldsymbol{\beta}(m))\mathbf{z}(\boldsymbol{\beta}(m))
\end{aligned}$$

where as in lectures $z_i(\boldsymbol{\beta}) = X\boldsymbol{\beta}_i + \tilde{y}_i = g(\mu_i) + (y_i - \mu_i)g'(\mu_i)$. This is the weighted least squares estimate using $\mathbf{z}(\boldsymbol{\beta}(m))$ as the data vector, predictor matrix X and weights $W(\boldsymbol{\beta}(m))$ as required.

5. Suppose that students answer questions on a test and that a specific student has an aptitude T. A particular question might have difficulty d_i and the student will get the answer correct only if $T > d_i$. Consider d_i fixed and $T \sim N(\mu, \sigma^2)$, then the probability that a randomly selected student will get the answer wrong is $p_i = \mathbb{P}(T < d_i)$.

   Show how you might model this situation using a probit regression model.

   Solution: We have

$$\begin{aligned}
p_i &= \mathbb{P}(T < d_i) \\
&= \mathbb{P}\left(\frac{T-\mu}{\sigma} < \frac{d_i - \mu}{\sigma}\right) \\
&= \Phi\left(\frac{1}{\sigma}d_i - \frac{\mu}{\sigma}\right)
\end{aligned}$$

   which is in the form of a probit regression model with predictor variable d, $\beta_0 = -\mu/\sigma$ and $\beta_1 = 1/\sigma$.

6. **Proportional odds in ordinal regression.** Suppose that Y_i takes values in the ordered set $\{1, \ldots, J\}$. Using a logit link, our model for $\gamma_{ij} = \mathbb{P}(Y_i \leq j)$ is

$$\gamma_{ij} = \text{logit}^{-1}(\theta_j - \mathbf{x}_i^T \boldsymbol{\beta}).$$

   Thinking of γ_{ij} as a function of $\mathbf{x}_i$, we can rewrite it as $\gamma_j(\mathbf{x}_i) = \mathbb{P}(Y \leq j|\mathbf{x}_i)$.

   Recall the odds for an event A are given by $\mathbb{P}(A)/(1 - \mathbb{P}(A))$. By relative odds we mean the ratio of two odds. Show that the relative odds for $\{Y \leq j|\mathbf{x}_A\}$ and $\{Y \leq j|\mathbf{x}_B\}$ do not depend on j.

Solution: The odds ratio is

$$
\begin{aligned}
\frac{\frac{\mathbb{P}(Y\leq j|\mathbf{x}_A)}{1-\mathbb{P}(Y\leq j|\mathbf{x}_A)}}{\frac{\mathbb{P}(Y\leq j|\mathbf{x}_B)}{1-\mathbb{P}(Y\leq j|\mathbf{x}_B)}} &= \frac{\exp(\mathrm{logit}(\mathbb{P}(Y\leq j|\mathbf{x}_A)}{\exp(\mathrm{logit}(\mathbb{P}(Y\leq j|\mathbf{x}_B)} \\
&= \frac{\exp(\theta_j - \mathbf{x}_A^T\boldsymbol{\beta})}{\exp(\theta_j - \mathbf{x}_B^T\boldsymbol{\beta})} \\
&= \exp(-(\mathbf{x}_A - \mathbf{x}_B)^T\beta)
\end{aligned}
$$

which does not depend on j, as required.

Note that the difference between the log odds is just $-(\mathbf{x}_A - \mathbf{x}_B)^T\beta$.