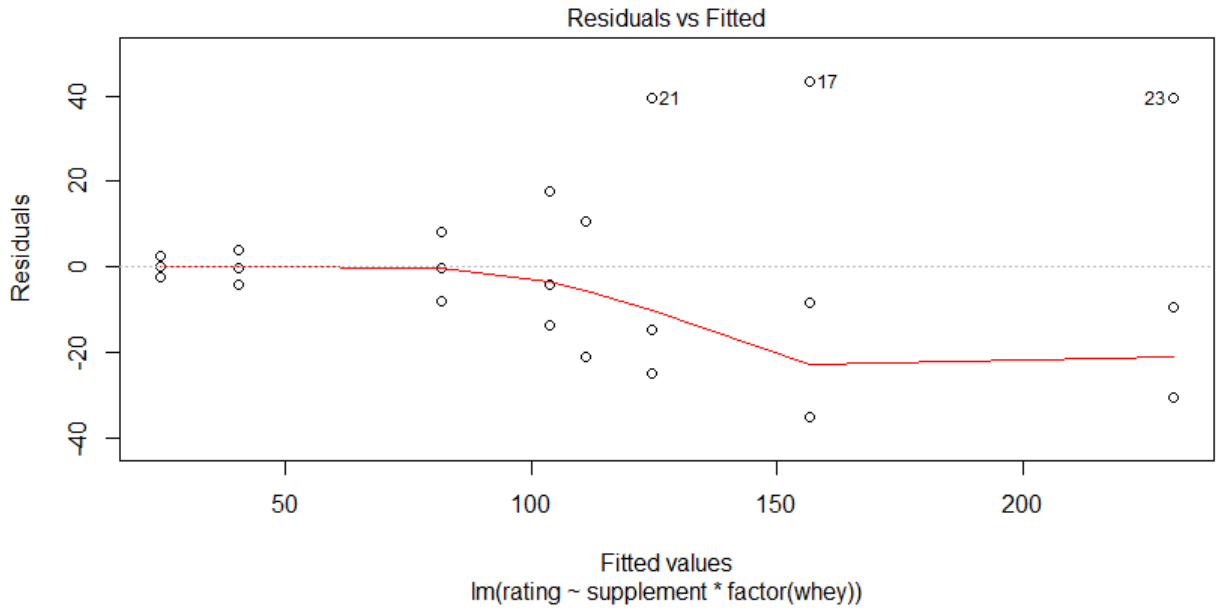
STAT3015 GENERALISED LINEAR MODELLING

R Output for the Practice Final Examination

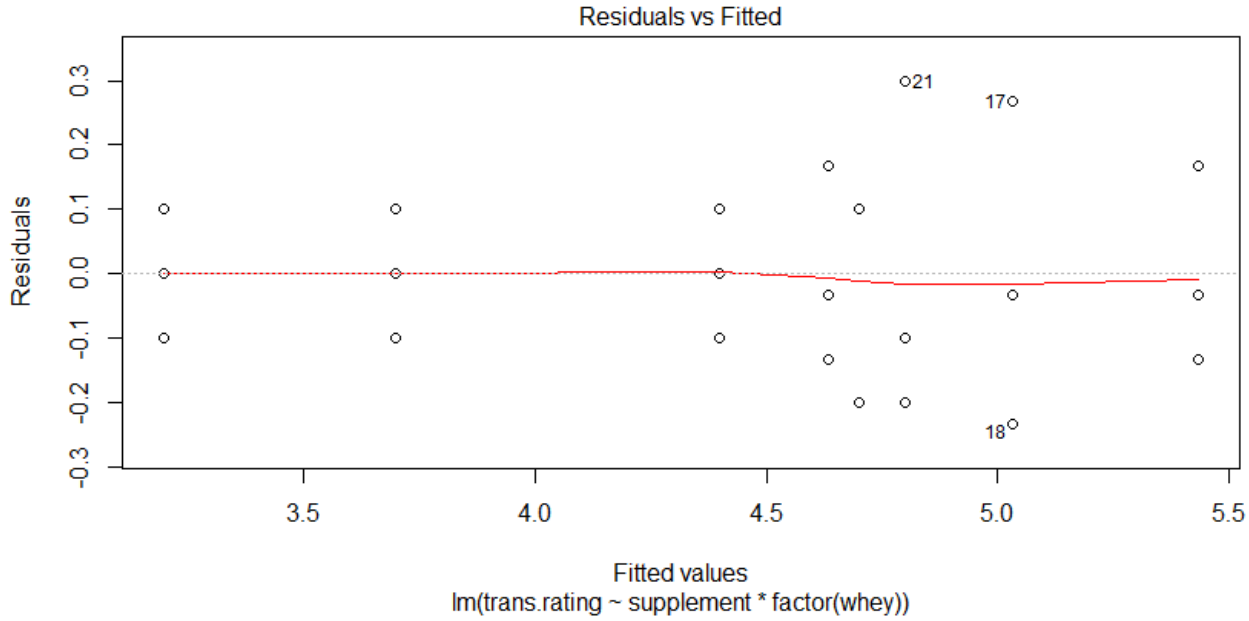
Question 1

```
> pancakes <- read.table("pancakes.txt", header=T)
> attach(pancakes)
>
> rating <- exp(trans.rating)
> data.frame(rating, trans.rating, supplement, whey)
      rating trans.rating supplement whey
1   81.45087          4.4         no    0
2   90.01713          4.5         no    0
3   73.69979          4.3         no    0
4   27.11264          3.3        yes    0
5   24.53253          3.2        yes    0
6   22.19795          3.1        yes    0
7   99.48432          4.6         no   10
8   90.01713          4.5         no   10
9  121.51042          4.8         no   10
10  44.70118          3.8        yes   10
11  40.44730          3.7        yes   10
12  36.59823          3.6        yes   10
13  90.01713          4.5         no   20
14 121.51042          4.8         no   20
15 121.51042          4.8         no   20
16 148.41316          5.0        yes   20
17 200.33681          5.3        yes   20
18 121.51042          4.8        yes   20
19  99.48432          4.6         no   30
20 109.94717          4.7         no   30
21 164.02191          5.1         no   30
22 221.40642          5.4        yes   30
23 270.42641          5.6        yes   30
24 200.33681          5.3        yes   30
>
> pancake.lm1 <- lm(rating ~ supplement*factor(whey))
> plot(pancake.lm1, which=1)
>
```

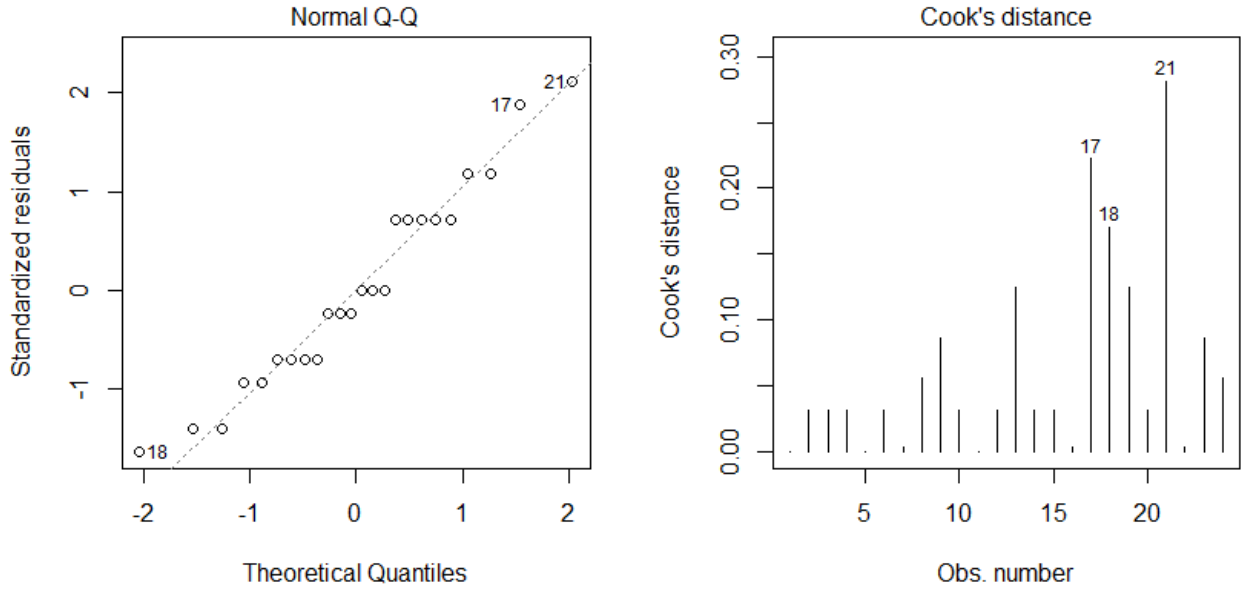


Question 1 continued

```
> pancake.lm <- lm(trans.rating ~ supplement*factor(whey))
>
> plot(pancake.lm, which=1)
```




```
>
> par(mfrow=c(1,2))
> plot(pancake.lm, which=c(2,4))
```



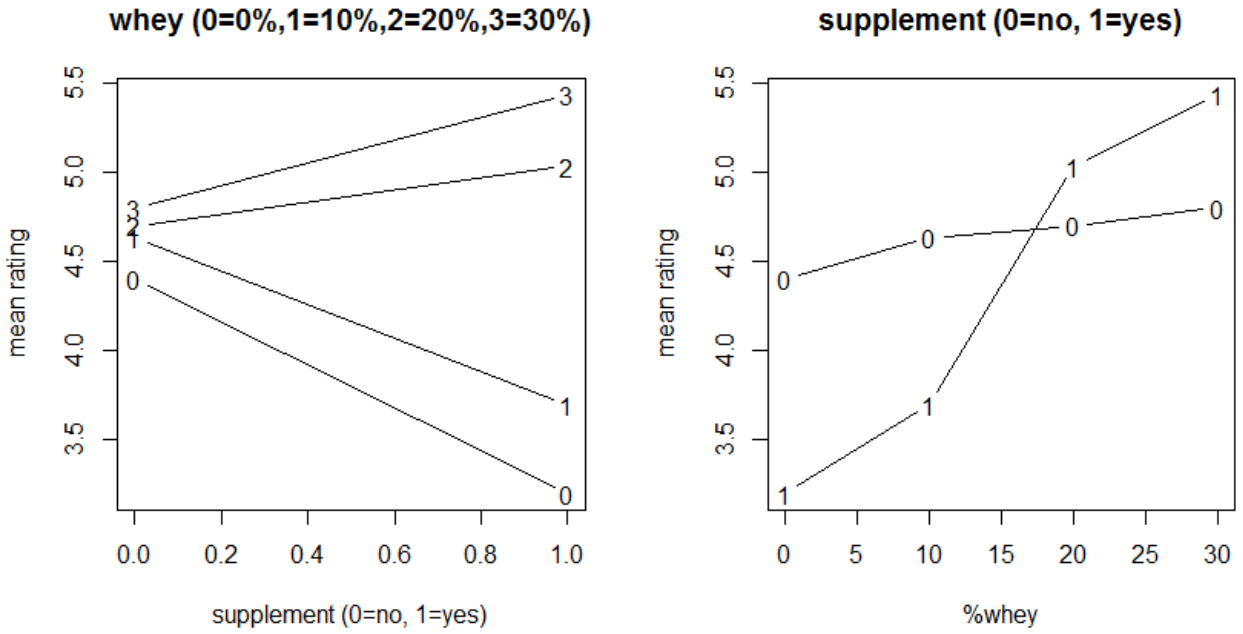

```
>
```

Question 1 continued

```
> anova(pancake.lm)
Analysis of Variance Table

Response: trans.rating
                       Df Sum Sq Mean Sq F value    Pr(>F)
supplement              1 0.5104 0.51042  17.014 0.0007942 ***
factor(whey)            3 6.6912 2.23042  74.347 1.304e-09 ***
supplement:factor(whey) 3 3.7246 1.24153  41.384 9.130e-08 ***
Residuals              16 0.4800 0.03000
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
>
> summary(pancake.lm)

Call:
lm(formula = trans.rating ~ supplement * factor(whey))

Residuals:
     Min       1Q   Median       3Q      Max
-0.23333 -0.10000 -0.01667  0.10000  0.30000

Coefficients:
                            Estimate Std. Error t value Pr(>|t|)
(Intercept)                   4.4000     0.1000  44.000  < 2e-16 ***
supplementyes                -1.2000     0.1414  -8.485 2.56e-07 ***
factor(whey)10                0.2333     0.1414   1.650   0.1185
factor(whey)20                0.3000     0.1414   2.121   0.0499 *
factor(whey)30                0.4000     0.1414   2.828   0.0121 *
supplementyes:factor(whey)10  0.2667     0.2000   1.333   0.2011
supplementyes:factor(whey)20  1.5333     0.2000   7.667 9.59e-07 ***
supplementyes:factor(whey)30  1.8333     0.2000   9.167 9.09e-08 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1732 on 16 degrees of freedom
Multiple R-squared:  0.9579,      Adjusted R-squared:  0.9395
F-statistic: 52.03 on 7 and 16 DF,  p-value: 7.94e-10

>
```

Question 1 continued

```
> means <- tapply(trans.rating, list(supplement, whey), mean)
> means
      0       10       20       30
no  4.4 4.633333 4.700000 4.800000
yes 3.2 3.700000 5.033333 5.433333
>
> matplot(c(0,1),means,type="b",pch="0123", lty=c(1,1,1,1), col=c(1,1,1,1),
xlab="supplement (0=no, 1=yes)", ylab="mean rating")
> title("whey (0=0%,1=10%,2=20%,3=30%)")
> matplot(c(0,10,20,30),t(means),type="b",pch="01", lty=c(1,1), col=c(1,1),
xlab="%whey",ylab="mean rating")
> title("supplement (0=no, 1=yes)")
```




```
> par(mfrow=c(1,1))
>
```

STAT3015 GENERALISED LINEAR MODELLING

R Output for the Practice Final Examination

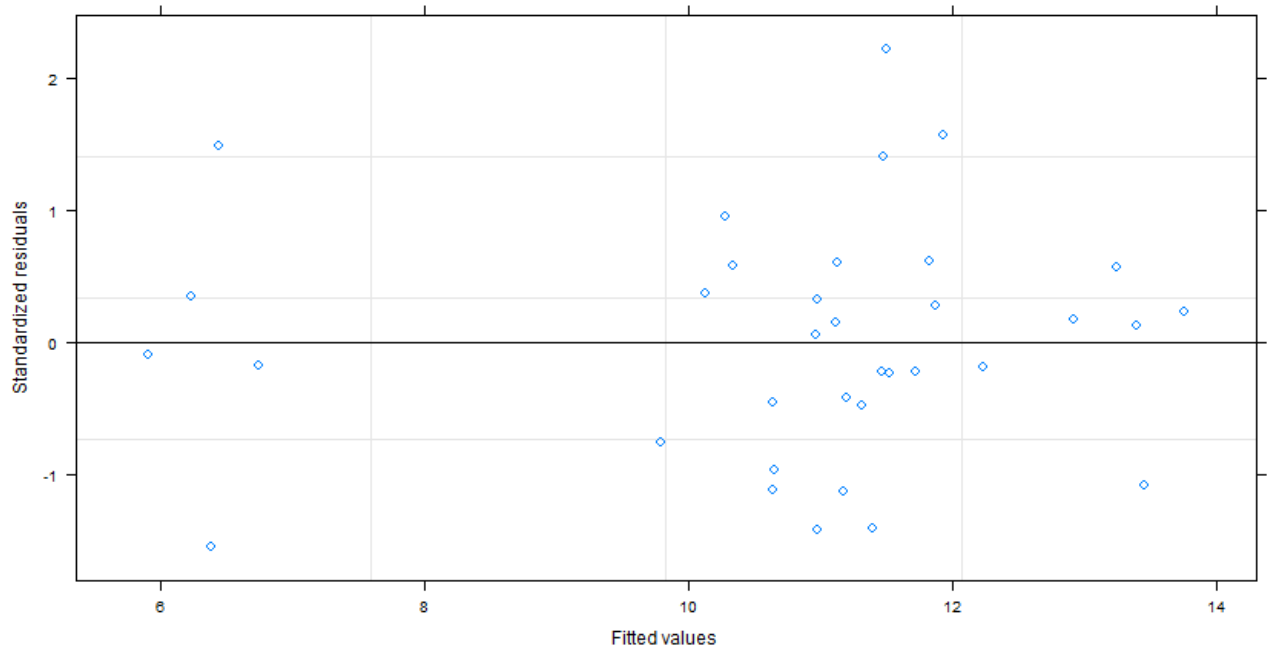
Question 1A

```
> rubber.plants <- read.table("rubber.txt", header=T)
> rubber.plants
   Block Col   P1    P2 Variety
1      1   1 4.06  3.75       2
2      1   2 6.65  6.71       4
3      1   3 6.85  4.94       7
4      1   4 1.46  6.39       1
5      1   5 2.96  2.71       6
6      1   6 2.53  6.93       3
7      1   7 2.06  6.12       5
8      2   1 4.07  7.73       1
9      2   2 5.92  5.00       7
10     2   3 1.85  6.44       3
11     2   4 4.06  6.65       4
12     2   5 4.36  5.85       6
13     2   6 9.27  6.64       2
14     2   7 5.00  5.12       5
15     3   1 3.88  6.22       7
16     3   2 2.59  4.79       5
17     3   3 7.77  6.91       4
18     3   4 2.03  5.08       6
19     3   5 6.42  4.72       2
20     3   6 5.20  5.90       3
21     3   7 6.29  4.77       1
22     4   1 4.43  7.31       2
23     4   2 6.89  0.89       1
24     4   3 6.49  8.55       3
25     4   4 5.41  0.87       6
26     4   5 6.71  6.67       7
27     4   6 6.46 10.66       5
28     4   7 6.12  8.21       4
29     5   1 5.82  5.08       7
30     5   2 6.64  5.92       2
31     5   3 0.48  1.97       6
32     5   4 7.30  4.19       3
33     5   5 8.11  5.59       4
34     5   6 7.35  5.33       1
35     5   7 7.66  5.00       5
> attach(rubber.plants)
> Yield <- P1 + P2
>
> options(contrasts=c(factor="contr.sum", ordered="contr.poly"))
> library(nlme)
Warning message:
package ‘nlme’ was built under R version 3.2.4
>
```

R Output for the Practice Final Examination

Question 1A continued

```
> rubber.lme <- lme(Yield ~ factor(Variety), random = ~ 1 | factor(Block))
>
> plot(rubber.lme)
```




```
>
```

Question 1A continued

```
> anova(rubber.lme)
                numDF denDF  F-value p-value
(Intercept)         1    24 457.2630  <.0001
factor(Variety)     6    24   3.6612  0.0101
>
> summary(rubber.lme)
Linear mixed-effects model fit by REML
 Data: NULL
       AIC      BIC    logLik
  166.0625 178.0524 -74.03127

Random effects:
 Formula: ~1 | factor(Block)
        (Intercept) Residual
StdDev:   0.5886959 2.538402

Fixed effects: Yield ~ factor(Variety)
                     Value Std.Error DF   t-value p-value
(Intercept)      10.764571 0.5034006 24 21.383710  0.0000
factor(Variety)1 -0.530571 1.0509982 24 -0.504826  0.6183
factor(Variety)2  1.067429 1.0509982 24  1.015633  0.3199
factor(Variety)3  0.311429 1.0509982 24  0.296317  0.7695
factor(Variety)4  2.591429 1.0509982 24  2.465683  0.0212
factor(Variety)5  0.327429 1.0509982 24  0.311541  0.7581
factor(Variety)6 -4.420571 1.0509982 24 -4.206069  0.0003
 Correlation:
                 (Intr) fc(V)1 fc(V)2 fc(V)3 fc(V)4 fc(V)5
factor(Variety)1  0.000
factor(Variety)2  0.000 -0.167
factor(Variety)3  0.000 -0.167 -0.167
factor(Variety)4  0.000 -0.167 -0.167 -0.167
factor(Variety)5  0.000 -0.167 -0.167 -0.167 -0.167
factor(Variety)6  0.000 -0.167 -0.167 -0.167 -0.167 -0.167

Standardized Within-Group Residuals:
        Min          Q1         Med          Q3         Max
-1.55078373 -0.47006319  0.05227838  0.46631841  2.21589640

Number of Observations: 35
Number of Groups: 5
>
```

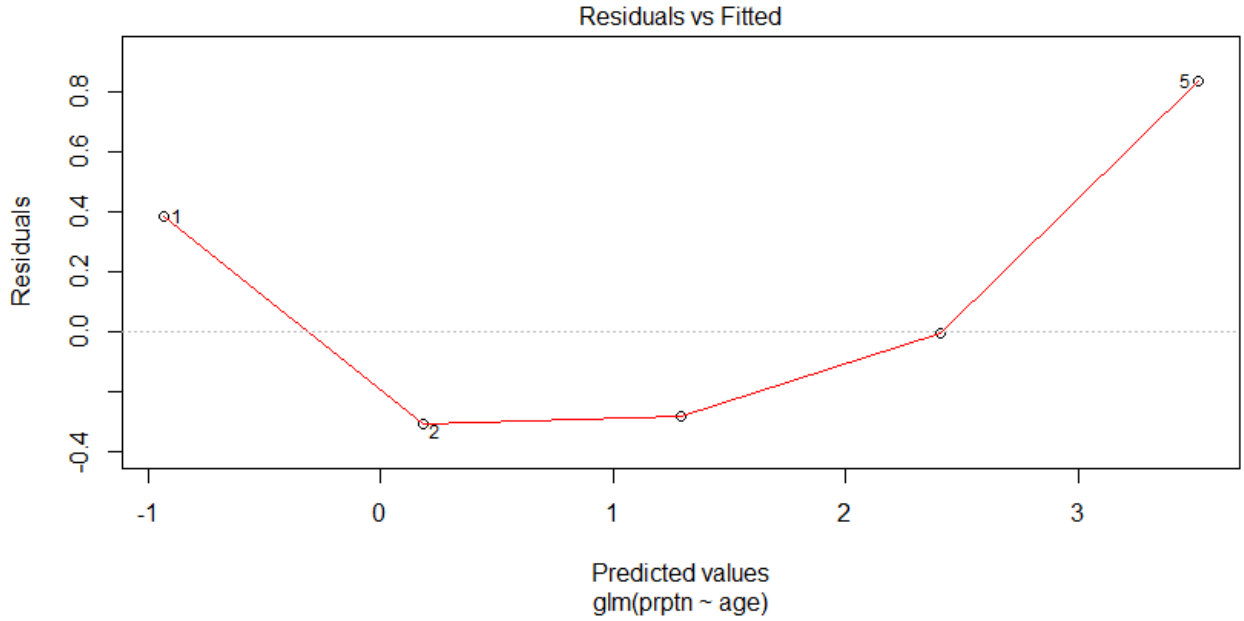
R Output for the Practice Final Examination

Question 2

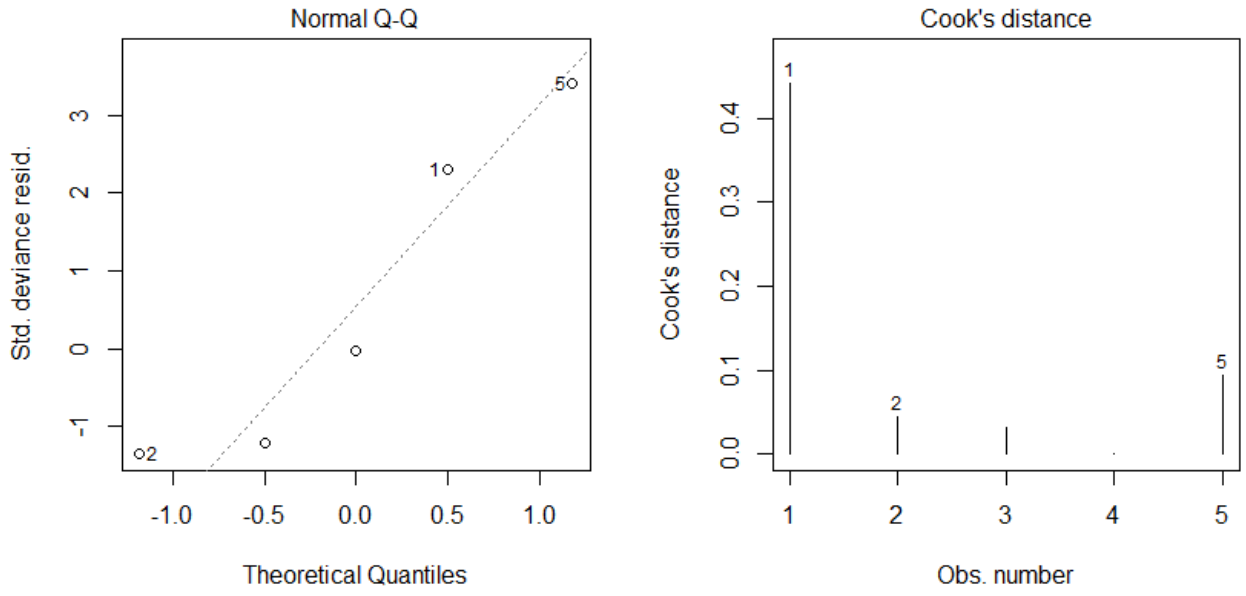
```
> age <- c(20, 30, 40, 50, 60)
> ssize <- rep(12, 5)
> favour <- c(4, 6, 9, 11, 12)
> prptn <- favour/ssize
> pension.glm <- glm(prptn ~ age, family=binomial, weights=ssize)
>
> summary(pension.glm)

Call:
glm(formula = prptn ~ age, family = binomial, weights = ssize)

Deviance Residuals:
       1         2         3         4         5
 0.38580  -0.31097  -0.28548  -0.00638   0.83794

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -3.15770    1.06971  -2.952 0.003158 **
age          0.11125    0.03093   3.596 0.000323 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 21.0117  on 4  degrees of freedom
Residual deviance:  1.0292  on 3  degrees of freedom
AIC: 15.498

Number of Fisher Scoring iterations: 4

>
> anova(pension.glm, test="Chisq")
Analysis of Deviance Table

Model: binomial, link: logit

Response: prptn

Terms added sequentially (first to last)


     Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
NULL                     4    21.0117
age   1   19.983         3     1.0292 7.815e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
>
```

Question 2 continued

```
> plot(pension.glm, which=1)
```




```
>
> par(mfrow=c(1,2))
> plot(pension.glm, which=c(2,4))
```




```
> par(mfrow=c(1,1))
>
```

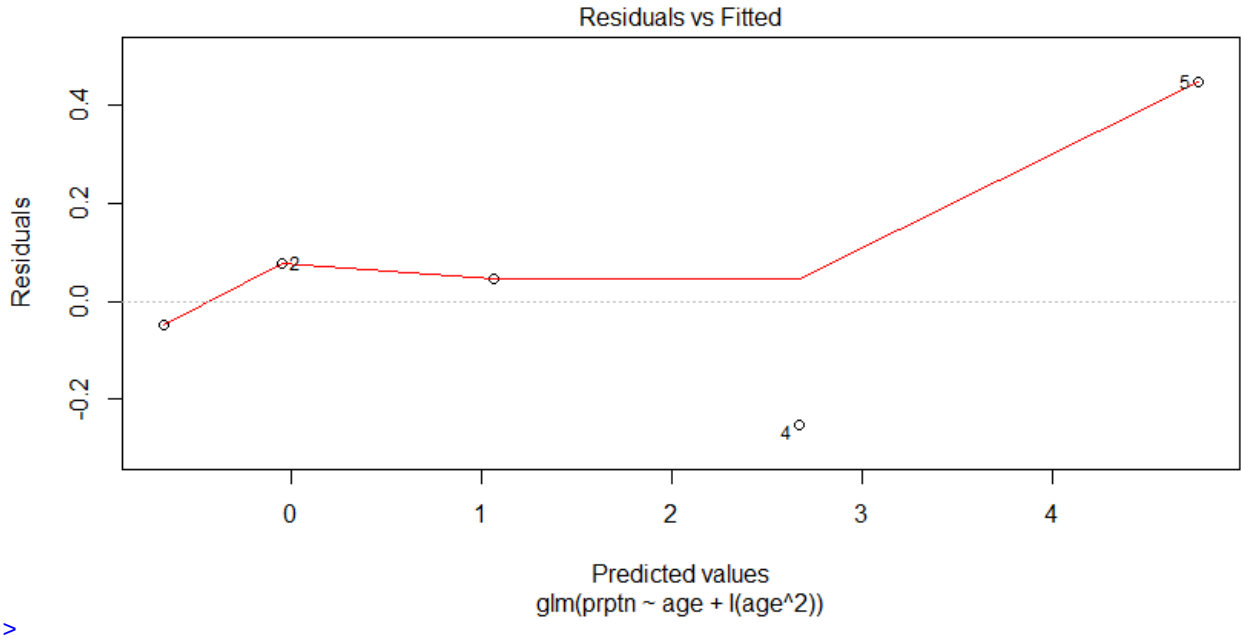
R Output for the Practice Final Examination

Question 2 continued

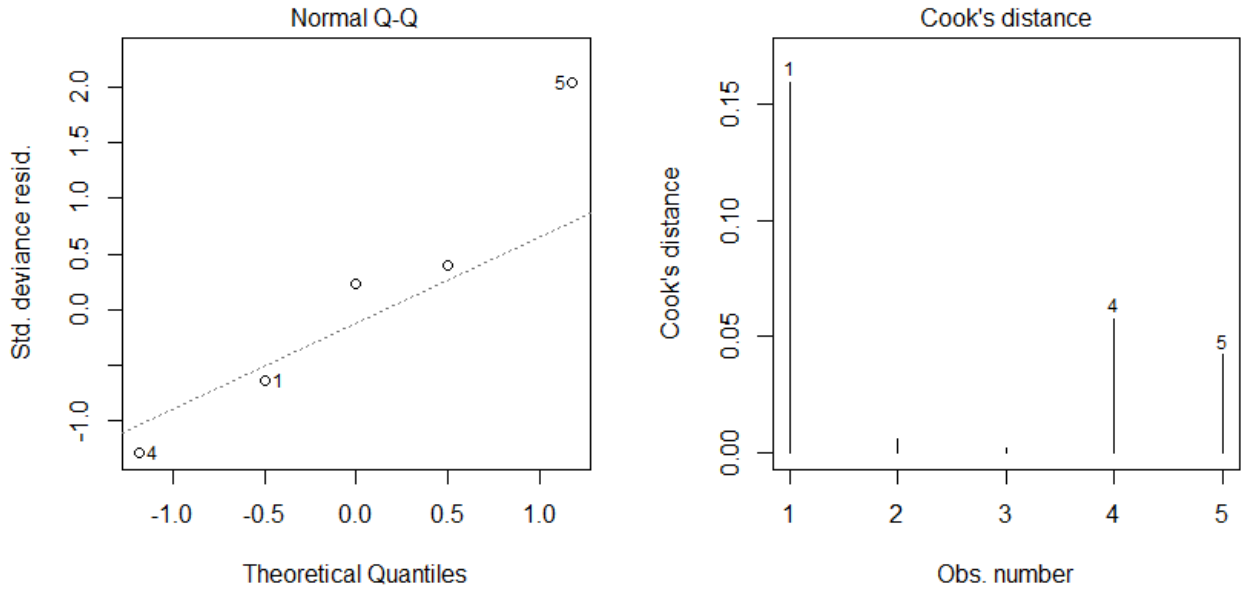
```
> pension.glm1 <- glm(prptn ~ age + I(age^2), family=binomial, weights=ssize)
>
> summary(pension.glm1)

Call:
glm(formula = prptn ~ age + I(age^2), family = binomial, weights = ssize)

Deviance Residuals:
       1         2         3         4         5
-0.04752   0.07658   0.04589  -0.25346   0.45018

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -0.426534   3.412301  -0.125    0.901
age         -0.061122   0.211230  -0.289    0.772
I(age^2)     0.002462   0.003071   0.802    0.423

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 21.01174  on 4  degrees of freedom
Residual deviance:  0.27713  on 2  degrees of freedom
AIC: 16.746

Number of Fisher Scoring iterations: 5

>
> anova(pension.glm1, test="Chisq")
Analysis of Deviance Table

Model: binomial, link: logit

Response: prptn

Terms added sequentially (first to last)


         Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
NULL                         4    21.0117
age       1  19.9825         3     1.0292 7.815e-06 ***
I(age^2)  1   0.7521         2     0.2771    0.3858
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
>
```

Question 2 continued

```
> plot(pension.glm1, which=1)
```




```
>
> par(mfrow=c(1,2))
> plot(pension.glm1, which=c(2,4))
```



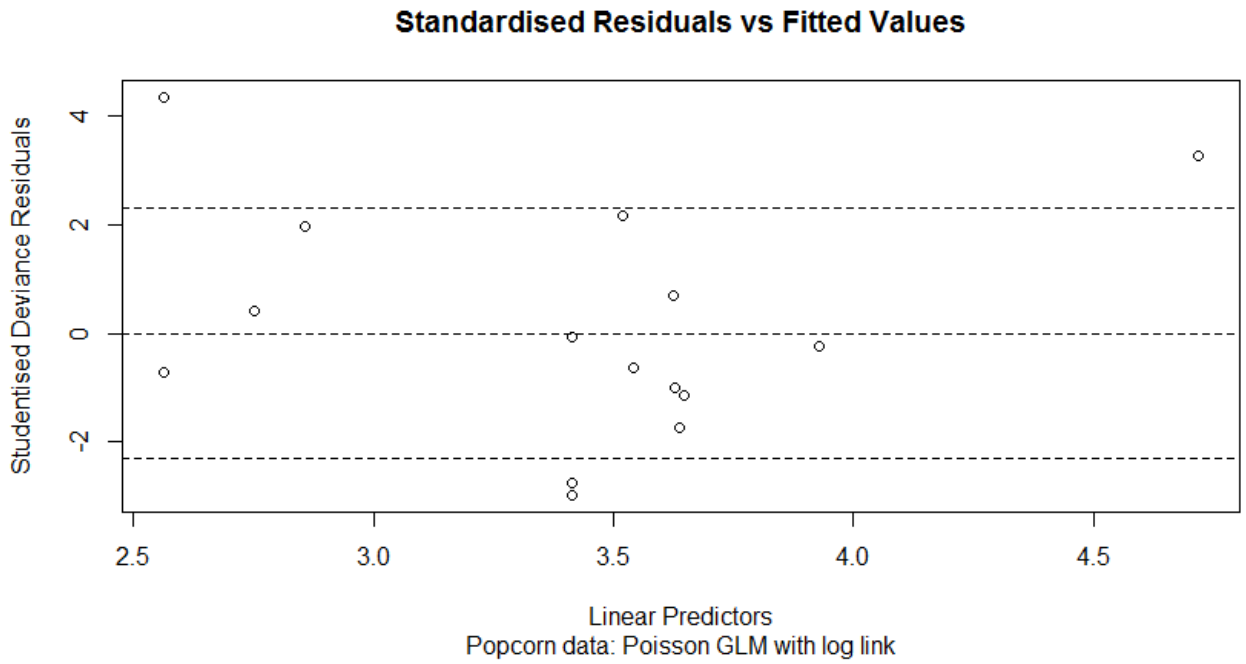

```
> par(mfrow=c(1,1))
>
```

Question 3

```
> popcorn <- read.table("popcorn.txt", header=T)
> popcorn
   temp oil time count
1     7   4   90    24
2     5   3  105    28
3     7   3  105    40
4     7   2   90    42
5     6   4  105    11
6     6   3   90    16
7     5   3   75   126
8     6   2  105    34
9     5   4   90    32
10    6   2   75    32
11    5   2   90    34
12    7   3   75    17
13    6   3   90    30
14    6   3   90    17
15    6   4   75    50
> attach(popcorn)
>
> popcorn.glm <- glm(count ~ temp*oil*time, family=poisson)
>
> std.residuals <- function(model, type="deviance"){
+   # Function to standardise residuals from a GLM model object
+   # Produces standardised deviance residuals, unless type="pearson" requested
+   std.error <- sqrt(summary(model)$dispersion * (1 - influence(model)$hat))
+   std.res <- residuals(model)/std.error
+   if (type=="pearson") std.res <- residuals(model, "pearson")/std.error
+   std.res
+ }
>
> popcorn.glm$linear.predictors
       1        2        3        4        5        6        7        8        9       10
11       12       13       14
2.856383 2.564289 3.625843 3.520263 2.563493 3.415060 4.719304 3.626638 3.636623 3.540594
3.646970 2.750804 3.415060 3.415060
      15
3.929514
> std.residuals(popcorn.glm)
          1           2           3           4           5           6           7
8           9          10
 1.98289019  4.36158572  0.70843123  2.16888626 -0.70669537 -2.99071051  3.26162695 -0.99
202193 -1.73189946 -0.64445870
         11          12          13          14          15
-1.13165035  0.41326491 -0.07910504 -2.76083580 -0.23370824
>
```

Question 3 continued

```
> plot(popcorn.glm$linear.predictors, std.residuals(popcorn.glm), xlab="Linear Predictors",
ylab="Studentised Deviance Residuals")
> abline(h=c(qt(0.025,popcorn.glm$df.residual),0,qt(0.975,popcorn.glm$df.residual)), lty=2)
> title("Standardised Residuals vs Fitted Values", sub="Popcorn data: Poisson GLM with log
link")
```




```
>
> anova(popcorn.glm, test="Chisq")
Analysis of Deviance Table

Model: poisson, link: log

Response: count

Terms added sequentially (first to last)


             Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
NULL                         14    213.454
temp          1   33.295        13    180.159 7.921e-09 ***
oil           1    2.204        12    177.955   0.13767
time          1   44.571        11    133.385 2.453e-11 ***
temp:oil      1    3.068        10    130.317   0.07987 .
temp:time     1   77.162         9     53.155 < 2.2e-16 ***
oil:time      1   16.440         8     36.715 5.020e-05 ***
temp:oil:time 0    0.000         8     36.715
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
>
```

Question 3 continued

```
> summary(popcorn.glm)

Call:
glm(formula = count ~ temp * oil * time, family = poisson)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.8772  -0.6562  -0.1240   0.8412   3.6036

Coefficients: (1 not defined because of singularities)
              Estimate Std. Error z value Pr(>|z|)
(Intercept)  24.996417   3.652979   6.843 7.77e-12 ***
temp         -4.281669   0.567942  -7.539 4.74e-14 ***
oil           2.989841   0.765450   3.906 9.38e-05 ***
time         -0.251735   0.037325  -6.744 1.54e-11 ***
temp:oil     -0.163383   0.090324  -1.809   0.0705 .
temp:time     0.050501   0.005694   8.868  < 2e-16 ***
oil:time     -0.024201   0.005959  -4.061 4.88e-05 ***
temp:oil:time        NA         NA      NA       NA
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 213.454  on 14  degrees of freedom
Residual deviance:  36.715  on  8  degrees of freedom
AIC: 129.29

Number of Fisher Scoring iterations: 4

>
> popcorn.glm1 <- glm(count ~ temp + oil + time + temp:time + oil:time, family=poisson)
>
> anova(popcorn.glm1, test="Chisq")
Analysis of Deviance Table

Model: poisson, link: log

Response: count

Terms added sequentially (first to last)


          Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
NULL                         14    213.454
temp       1   33.295        13    180.159 7.921e-09 ***
oil        1    2.204        12    177.955    0.1377
time       1   44.571        11    133.385 2.453e-11 ***
temp:time  1   77.438        10     55.947 < 2.2e-16 ***
oil:time   1   15.962         9     39.985 6.463e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
>
```

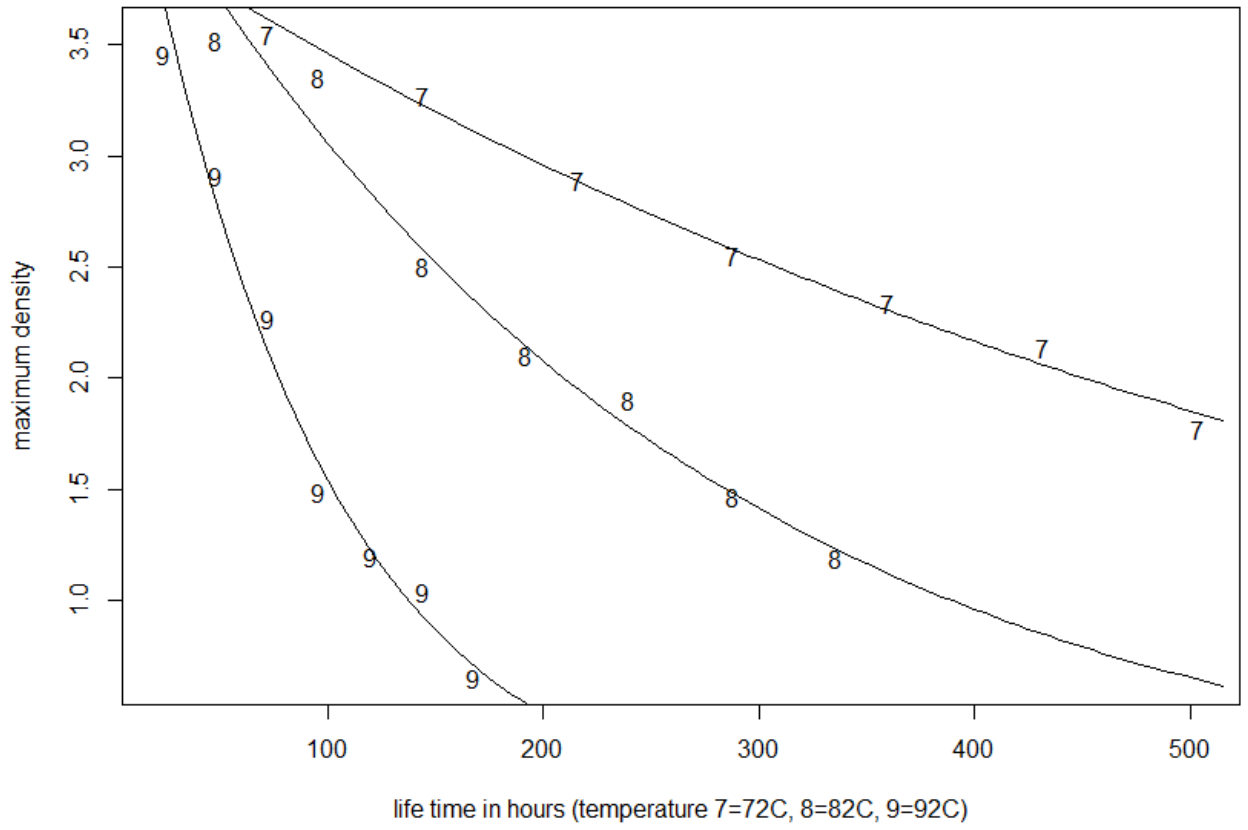
R Output for the Practice Final Examination

Question 4

```
> photo <- read.table("photo.txt", header=T)
> attach(photo)
The following object is masked from popcorn:

    temp

> temp82 <- ifelse(temp==82,1,0)
> temp92 <- ifelse(temp==92,1,0)
> temps <- cbind(temp82, temp92)
>
> photo.glm <- glm(density ~ lifetime * temps, family = Gamma(link=log))
>
> anova(photo.glm)
Analysis of Deviance Table

Model: Gamma, link: log

Response: density

Terms added sequentially (first to last)

               Df Deviance Resid. Df Resid. Dev
NULL                              20     3.7808
lifetime        1  0.45327        19     3.3275
temps           2  1.81813        17     1.5094
lifetime:temps  2  1.45322        15     0.0561
>
> summary(photo.glm)

Call:
glm(formula = density ~ lifetime * temps, family = Gamma(link = log))

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-0.08160  -0.03264  -0.01145   0.03803   0.11451

Coefficients:
                      Estimate Std. Error t value Pr(>|t|)
(Intercept)          1.3966619  0.0520877  26.814 4.35e-14 ***
lifetime            -0.0015594  0.0001618  -9.640 8.08e-08 ***
tempstemp82          0.1059092  0.0736631   1.438   0.1710
tempstemp92          0.1796877  0.0736631   2.439   0.0276 *
lifetime:tempstemp82 -0.0022972  0.0002916  -7.877 1.04e-06 ***
lifetime:tempstemp92 -0.0098955  0.0005115 -19.344 5.11e-12 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for Gamma family taken to be 0.00379838)

    Null deviance: 3.780765  on 20  degrees of freedom
Residual deviance: 0.056144  on 15  degrees of freedom
AIC: -20.254

Number of Fisher Scoring iterations: 4
>
```

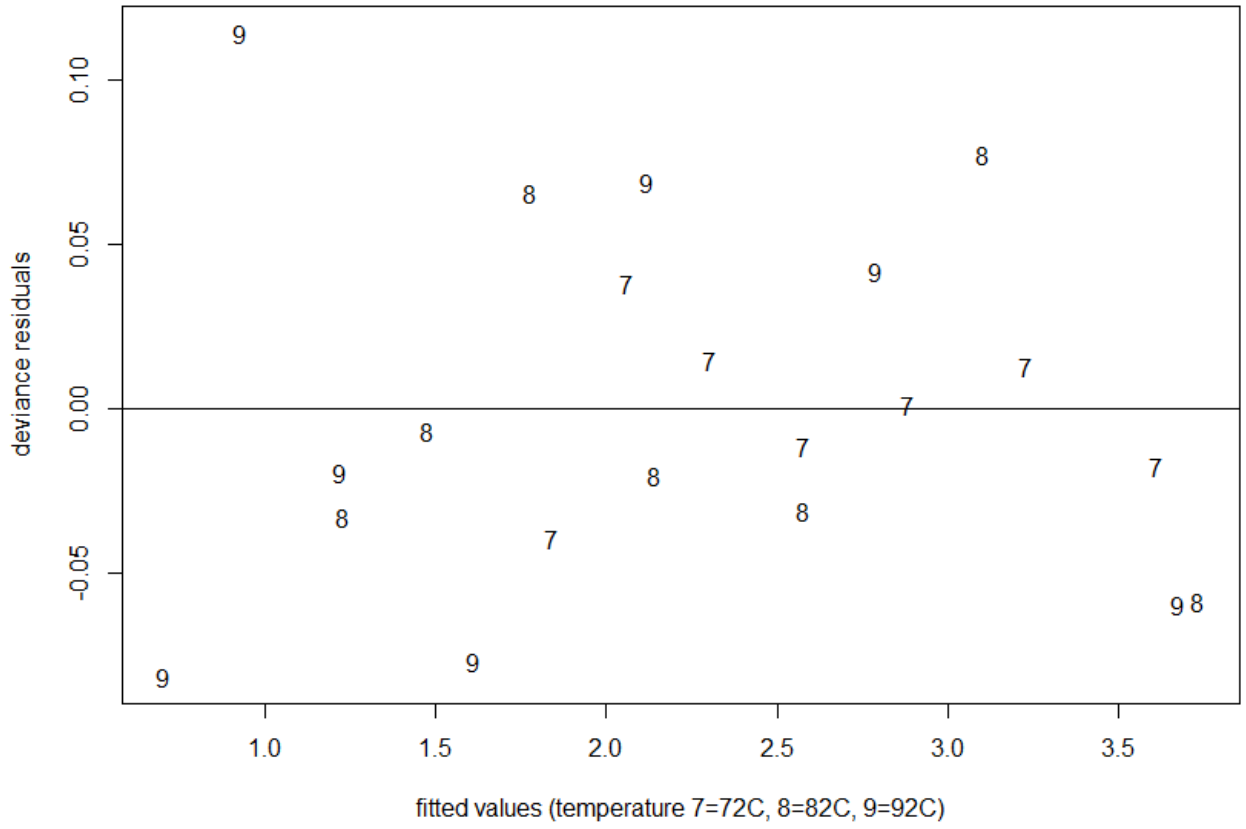
Question 4 continued

```
> plot(lifetime, density, type="n", xlab="life time in hours (temperature 7=72C, 8=82C,
9=92C)", ylab="maximum density")
> points(lifetime[temp==72], density[temp==72], pch="7")
> points(lifetime[temp==82], density[temp==82], pch="8")
> points(lifetime[temp==92], density[temp==92], pch="9")
> newltime <- 5*0:103
> fitted72 <- as.vector(cbind(1, newltime, 0, 0, 0, 0) %*% coef(photo.glm))
> lines(newltime, exp(fitted72))
> fitted82 <- as.vector(cbind(1, newltime, 1, 0, newltime, 0) %*% coef(photo.glm))
> lines(newltime, exp(fitted82))
> fitted92 <- as.vector(cbind(1, newltime, 0, 1, 0, newltime) %*% coef(photo.glm))
> lines(newltime, exp(fitted92))
>
```



Question 4 continued

```
> plot(fitted(photo.glm), residuals(photo.glm, "deviance"), type="n", xlab="fitted values
(temperature 7=72C, 8=82C, 9=92C)", ylab="deviance residuals")
> abline(0,0)
> points(fitted(photo.glm)[temp==72], residuals(photo.glm, "deviance")[temp==72], pch="7")
> points(fitted(photo.glm)[temp==82], residuals(photo.glm, "deviance")[temp==82], pch="8")
> points(fitted(photo.glm)[temp==92], residuals(photo.glm, "deviance")[temp==92], pch="9")
```




```
>
```

R Output for the Practice Final Examination

Question 5

```
> # Enter the data:
> carprefs <- array(0, c(2,2,2))
> carprefs[,,1] <- matrix(c(68,168,12,32),ncol=2)
> carprefs[,,2] <- matrix(c(16,84,24,164),ncol=2)
> dimnames(carprefs) <-
list(c("imported","local"),c("city","country"),c("males","females"))
> carprefs
, , males

         city country
imported   68      12
local     168      32

, , females

         city country
imported   16      24
local      84     164

>
> # Calculate the expected values for the two sexes separately:
> obs.males <- carprefs[,,"males"]
> males.row.tot <- apply(obs.males, 1, sum)
> males.col.tot <- apply(obs.males, 2, sum)
> males.total <- sum(obs.males)
> exp.males <- males.row.tot %*% t(males.col.tot) / males.total
> dimnames(exp.males)[[1]] <- c("imported", "local")
> exp.males
             city  country
imported  67.42857 12.57143
local    168.57143 31.42857
>
> # Pearson.chisq <- sum((obs.males - exp.males)^2/exp.males)
> # Pearson.chisq
> # 1-pchisq(Pearson.chisq, (nrow(obs.males)-1)*(ncol(obs.males)-1))
>
> obs.females <- carprefs[,,"females"]
> females.row.tot <- apply(obs.females, 1, sum)
> females.col.tot <- apply(obs.females, 2, sum)
> females.total <- sum(obs.females)
> exp.females <- females.row.tot %*% t(females.col.tot) / females.total
> dimnames(exp.females)[[1]] <- c("imported", "local")
> exp.females
            city   country
imported 13.88889  26.11111
local    86.11111 161.88889
>
> # Pearson.chisq <- sum((obs.females - exp.females)^2/exp.females)
> # Pearson.chisq
> # 1-pchisq(Pearson.chisq, (nrow(obs.females)-1)*(ncol(obs.females)-1))
>
```

Question 5 continued

```
> # Collapse the data table so that sex is ignored:
> obs.nosex <- apply(carprefs, c(1,2), sum)
> obs.nosex
         city country
imported   84      36
local     252     196
>
> # nosex.row.tot <- apply(obs.nosex, 1, sum)
> # nosex.col.tot <- apply(obs.nosex, 2, sum)
> # nosex.total <- sum(obs.nosex)
> # exp.nosex <- nosex.row.tot %*% t(nosex.col.tot) / nosex.total
> # dimnames(exp.nosex)[[1]] <- c("imported", "local")
> # exp.nosex
>
> # LR.chisq <- 2*sum(obs.nosex*log(obs.nosex/exp.nosex))
> # LR.chisq
> # 1-pchisq(LR.chisq, (nrow(obs.nosex)-1)*(ncol(obs.nosex)-1))
>
> # Collapse the data table so that residence is ignored:
> obs.nores <- apply(carprefs, c(1,3), sum)
> obs.nores
         males females
imported    80      40
local      200     248
>
> nores.row.tot <- apply(obs.nores, 1, sum)
> nores.col.tot <- apply(obs.nores, 2, sum)
> nores.total <- sum(obs.nores)
> exp.nores <- nores.row.tot %*% t(nores.col.tot) / nores.total
> dimnames(exp.nores)[[1]] <- c("imported", "local")
> exp.nores
             males   females
imported  59.15493  60.84507
local    220.84507 227.15493
>
> Pearson.chisq <- sum((obs.nores - exp.nores)^2/exp.nores)
> Pearson.chisq
[1] 18.36716
> 1-pchisq(Pearson.chisq, (nrow(obs.nores)-1)*(ncol(obs.nores)-1))
[1] 1.82171e-05
>
> # Collapse the data table so that preference is ignored:
> obs.nopref <- apply(carprefs, c(2,3), sum)
> obs.nopref
        males females
city      236     100
country    44     188
>
> nopref.row.tot <- apply(obs.nopref, 1, sum)
> nopref.col.tot <- apply(obs.nopref, 2, sum)
> nopref.total <- sum(obs.nopref)
> exp.nopref <- nores.row.tot %*% t(nopref.col.tot) / nopref.total
> dimnames(exp.nopref)[[1]] <- c("city", "country")
> exp.nopref
            males   females
city     59.15493  60.84507
country 220.84507 227.15493
>
> LR.chisq <- 2*sum(obs.nopref*log(obs.nopref/exp.nopref))
> LR.chisq
[1] 539.3575
> 1-pchisq(LR.chisq, (nrow(obs.nopref)-1)*(ncol(obs.nopref)-1))
[1] 0
>
```

R Output for the Practice Final Examination

Statistical Tables

```
> # Selected quantiles of Student's t distribution:
> DF <- c(1:100, 150, 200, 250, 300, 400, 500, 750, 1000, 10000, 1000000)
> tquantiles0.025 <- round(qt(0.025,DF),4)
> tquantiles0.05 <- round(qt(0.05,DF),4)
> tquantiles0.95 <- round(qt(0.95,DF),4)
> tquantiles0.975 <- round(qt(0.975,DF),4)
> tquantiles <- rbind(tquantiles0.025, tquantiles0.05, tquantiles0.95, tquantiles0.975)
> dimnames(tquantiles) <- list(c("t 0.025", "t 0.05 ", "t 0.95 ", "t 0.975"), DF)
> tquantiles
              1       2       3       4       5       6       7       8       9      10
t 0.025 -12.7062 -4.3027 -3.1824 -2.7764 -2.5706 -2.4469 -2.3646 -2.3060 -2.2622 -2.2281
t 0.05   -6.3138 -2.9200 -2.3534 -2.1318 -2.0150 -1.9432 -1.8946 -1.8595 -1.8331 -1.8125
t 0.95    6.3138  2.9200  2.3534  2.1318  2.0150  1.9432  1.8946  1.8595  1.8331  1.8125
t 0.975  12.7062  4.3027  3.1824  2.7764  2.5706  2.4469  2.3646  2.3060  2.2622  2.2281
             11      12      13      14      15      16      17      18      19      20
t 0.025  -2.2010 -2.1788 -2.1604 -2.1448 -2.1314 -2.1199 -2.1098 -2.1009 -2.0930 -2.0860
t 0.05   -1.7959 -1.7823 -1.7709 -1.7613 -1.7531 -1.7459 -1.7396 -1.7341 -1.7291 -1.7247
t 0.95    1.7959  1.7823  1.7709  1.7613  1.7531  1.7459  1.7396  1.7341  1.7291  1.7247
t 0.975   2.2010  2.1788  2.1604  2.1448  2.1314  2.1199  2.1098  2.1009  2.0930  2.0860
             21      22      23      24      25      26      27      28      29      30
t 0.025  -2.0796 -2.0739 -2.0687 -2.0639 -2.0595 -2.0555 -2.0518 -2.0484 -2.0452 -2.0423
t 0.05   -1.7207 -1.7171 -1.7139 -1.7109 -1.7081 -1.7056 -1.7033 -1.7011 -1.6991 -1.6973
t 0.95    1.7207  1.7171  1.7139  1.7109  1.7081  1.7056  1.7033  1.7011  1.6991  1.6973
t 0.975   2.0796  2.0739  2.0687  2.0639  2.0595  2.0555  2.0518  2.0484  2.0452  2.0423
             31      32      33      34      35      36      37      38      39      40
t 0.025  -2.0395 -2.0369 -2.0345 -2.0322 -2.0301 -2.0281 -2.0262 -2.0244 -2.0227 -2.0211
t 0.05   -1.6955 -1.6939 -1.6924 -1.6909 -1.6896 -1.6883 -1.6871 -1.6860 -1.6849 -1.6839
t 0.95    1.6955  1.6939  1.6924  1.6909  1.6896  1.6883  1.6871  1.6860  1.6849  1.6839
t 0.975   2.0395  2.0369  2.0345  2.0322  2.0301  2.0281  2.0262  2.0244  2.0227  2.0211
             41      42      43      44      45      46      47      48      49      50
t 0.025  -2.0195 -2.0181 -2.0167 -2.0154 -2.0141 -2.0129 -2.0117 -2.0106 -2.0096 -2.0086
t 0.05   -1.6829 -1.6820 -1.6811 -1.6802 -1.6794 -1.6787 -1.6779 -1.6772 -1.6766 -1.6759
t 0.95    1.6829  1.6820  1.6811  1.6802  1.6794  1.6787  1.6779  1.6772  1.6766  1.6759
t 0.975   2.0195  2.0181  2.0167  2.0154  2.0141  2.0129  2.0117  2.0106  2.0096  2.0086
             51      52      53      54      55      56      57      58      59      60
t 0.025  -2.0076 -2.0066 -2.0057 -2.0049 -2.0040 -2.0032 -2.0025 -2.0017 -2.0010 -2.0003
t 0.05   -1.6753 -1.6747 -1.6741 -1.6736 -1.6730 -1.6725 -1.6720 -1.6716 -1.6711 -1.6706
t 0.95    1.6753  1.6747  1.6741  1.6736  1.6730  1.6725  1.6720  1.6716  1.6711  1.6706
t 0.975   2.0076  2.0066  2.0057  2.0049  2.0040  2.0032  2.0025  2.0017  2.0010  2.0003
             61      62      63      64      65      66      67      68      69      70
t 0.025  -1.9996 -1.9990 -1.9983 -1.9977 -1.9971 -1.9966 -1.9960 -1.9955 -1.9949 -1.9944
t 0.05   -1.6702 -1.6698 -1.6694 -1.6690 -1.6686 -1.6683 -1.6679 -1.6676 -1.6672 -1.6669
t 0.95    1.6702  1.6698  1.6694  1.6690  1.6686  1.6683  1.6679  1.6676  1.6672  1.6669
t 0.975   1.9996  1.9990  1.9983  1.9977  1.9971  1.9966  1.9960  1.9955  1.9949  1.9944
             71      72      73      74      75      76      77      78      79      80
t 0.025  -1.9939 -1.9935 -1.9930 -1.9925 -1.9921 -1.9917 -1.9913 -1.9908 -1.9905 -1.9901
t 0.05   -1.6666 -1.6663 -1.6660 -1.6657 -1.6654 -1.6652 -1.6649 -1.6646 -1.6644 -1.6641
t 0.95    1.6666  1.6663  1.6660  1.6657  1.6654  1.6652  1.6649  1.6646  1.6644  1.6641
t 0.975   1.9939  1.9935  1.9930  1.9925  1.9921  1.9917  1.9913  1.9908  1.9905  1.9901
             81      82      83      84      85      86      87      88      89      90
t 0.025  -1.9897 -1.9893 -1.9890 -1.9886 -1.9883 -1.9879 -1.9876 -1.9873 -1.9870 -1.9867
t 0.05   -1.6639 -1.6636 -1.6634 -1.6632 -1.6630 -1.6628 -1.6626 -1.6624 -1.6622 -1.6620
t 0.95    1.6639  1.6636  1.6634  1.6632  1.6630  1.6628  1.6626  1.6624  1.6622  1.6620
t 0.975   1.9897  1.9893  1.9890  1.9886  1.9883  1.9879  1.9876  1.9873  1.9870  1.9867
             91      92      93      94      95      96      97      98      99     100
t 0.025  -1.9864 -1.9861 -1.9858 -1.9855 -1.9853 -1.9850 -1.9847 -1.9845 -1.9842 -1.9840
t 0.05   -1.6618 -1.6616 -1.6614 -1.6612 -1.6611 -1.6609 -1.6607 -1.6606 -1.6604 -1.6602
t 0.95    1.6618  1.6616  1.6614  1.6612  1.6611  1.6609  1.6607  1.6606  1.6604  1.6602
t 0.975   1.9864  1.9861  1.9858  1.9855  1.9853  1.9850  1.9847  1.9845  1.9842  1.9840
            150     200     250     300     400     500     750    1000   10000   1e+06
t 0.025  -1.9759 -1.9719 -1.9695 -1.9679 -1.9659 -1.9647 -1.9631 -1.9623 -1.9602 -1.9600
t 0.05   -1.6551 -1.6525 -1.6510 -1.6499 -1.6487 -1.6479 -1.6469 -1.6464 -1.6450 -1.6449
t 0.95    1.6551  1.6525  1.6510  1.6499  1.6487  1.6479  1.6469  1.6464  1.6450  1.6449
t 0.975   1.9759  1.9719  1.9695  1.9679  1.9659  1.9647  1.9631  1.9623  1.9602  1.9600
>
```

Statistical Tables continued

```
> #Selected quantiles of the Chi-square distribution:
> DF <- c(1:50, 60, 70, 80, 90, 100, 150, 200, 250:270, 300, 350, 400, 450, 500,
550, 600, 650, 700, 750, 800, 850, 900, 950, 1000)
> chisq0.025 <- round(qchisq(0.025,DF),4)
> chisq0.05 <- round(qchisq(0.05,DF),4)
> chisq0.95 <- round(qchisq(0.95,DF),4)
> chisq0.975 <- round(qchisq(0.975,DF),4)
> chisq_quantiles <- rbind(chisq0.025, chisq0.05, chisq0.95, chisq0.975)
> dimnames(chisq_quantiles) <- list(c("chisq 0.025","chisq 0.05 ","chisq 0.95 ",
"chisq 0.975"), DF)
> chisq_quantiles
                  1       2       3       4       5       6       7       8       9      10
chisq 0.025  0.0010  0.0506  0.2158  0.4844  0.8312  1.2373  1.6899  2.1797  2.7004  3.2470
chisq 0.05   0.0039  0.1026  0.3518  0.7107  1.1455  1.6354  2.1673  2.7326  3.3251  3.9403
chisq 0.95   3.8415  5.9915  7.8147  9.4877 11.0705 12.5916 14.0671 15.5073 16.9190 18.3070
chisq 0.975  5.0239  7.3778  9.3484 11.1433 12.8325 14.4494 16.0128 17.5345 19.0228 20.4832
                 11      12      13      14      15      16      17      18      19      20
chisq 0.025  3.8157  4.4038  5.0088  5.6287  6.2621  6.9077  7.5642  8.2307  8.9065  9.5908
chisq 0.05   4.5748  5.2260  5.8919  6.5706  7.2609  7.9616  8.6718  9.3905 10.1170 10.8508
chisq 0.95  19.6751 21.0261 22.3620 23.6848 24.9958 26.2962 27.5871 28.8693 30.1435 31.4104
chisq 0.975 21.9200 23.3367 24.7356 26.1189 27.4884 28.8454 30.1910 31.5264 32.8523 34.1696
                 21      22      23      24      25      26      27      28      29      30
chisq 0.025 10.2829 10.9823 11.6886 12.4012 13.1197 13.8439 14.5734 15.3079 16.0471 16.7908
chisq 0.05  11.5913 12.3380 13.0905 13.8484 14.6114 15.3792 16.1514 16.9279 17.7084 18.4927
chisq 0.95  32.6706 33.9244 35.1725 36.4150 37.6525 38.8851 40.1133 41.3371 42.5570 43.7730
chisq 0.975 35.4789 36.7807 38.0756 39.3641 40.6465 41.9232 43.1945 44.4608 45.7223 46.9792
                 31      32      33      34      35      36      37      38      39      40
chisq 0.025 17.5387 18.2908 19.0467 19.8063 20.5694 21.3359 22.1056 22.8785 23.6543 24.4330
chisq 0.05  19.2806 20.0719 20.8665 21.6643 22.4650 23.2686 24.0749 24.8839 25.6954 26.5093
chisq 0.95  44.9853 46.1943 47.3999 48.6024 49.8018 50.9985 52.1923 53.3835 54.5722 55.7585
chisq 0.975 48.2319 49.4804 50.7251 51.9660 53.2033 54.4373 55.6680 56.8955 58.1201 59.3417
                 41      42      43      44      45      46      47      48      49      50
chisq 0.025 25.2145 25.9987 26.7854 27.5746 28.3662 29.1601 29.9562 30.7545 31.5549 32.3574
chisq 0.05  27.3256 28.1440 28.9647 29.7875 30.6123 31.4390 32.2676 33.0981 33.9303 34.7643
chisq 0.95  56.9424 58.1240 59.3035 60.4809 61.6562 62.8296 64.0011 65.1708 66.3386 67.5048
chisq 0.975 60.5606 61.7768 62.9904 64.2015 65.4102 66.6165 67.8206 69.0226 70.2224 71.4202
                 60      70      80      90     100     150     200     250     251
chisq 0.025 40.4817 48.7576  57.1532  65.6466  74.2219 117.9845 162.7280 208.0978 209.0102
chisq 0.05  43.1880 51.7393  60.3915  69.1260  77.9295 122.6918 168.2786 214.3916 215.3180
chisq 0.95  79.0819 90.5312 101.8795 113.1453 124.3421 179.5806 233.9943 287.8815 288.9551
chisq 0.975 83.2977 95.0232 106.6286 118.1359 129.5612 185.8004 241.0579 295.6886 296.7763
                252     253     254     255     256     257     258     259     260
chisq 0.025 209.9227 210.8355 211.7484 212.6614 213.5747 214.4881 215.4017 216.3154 217.2293
chisq 0.05  216.2446 217.1713 218.0982 219.0253 219.9524 220.8797 221.8072 222.7348 223.6625
chisq 0.95  290.0285 291.1017 292.1749 293.2478 294.3207 295.3934 296.4659 297.5383 298.6106
chisq 0.975 297.8637 298.9510 300.0381 301.1250 302.2118 303.2984 304.3848 305.4711 306.5572
                261     262     263     264     265     266     267     268     269
chisq 0.025 218.1434 219.0576 219.9720 220.8866 221.8013 222.7162 223.6313 224.5465 225.4619
chisq 0.05  224.5904 225.5184 226.4466 227.3748 228.3033 229.2318 230.1605 231.0893 232.0183
chisq 0.95  299.6827 300.7547 301.8266 302.8983 303.9699 305.0413 306.1126 307.1838 308.2548
chisq 0.975 307.6431 308.7289 309.8145 310.8999 311.9852 313.0703 314.1553 315.2401 316.3247
                270     300     350     400     450     500     550     600     650
chisq 0.025 226.3774 253.9123 300.0637 346.4818 393.1177 439.9360 486.9099 534.0186 581.2454
chisq 0.05  232.9474 260.8781 307.6476 354.6410 401.8173 449.1468 496.6068 544.1801 591.8526
chisq 0.95  309.3258 341.3951 394.6258 447.6325 500.4562 553.1268 605.6668 658.0936 710.4211
chisq 0.975 317.4092 349.8745 403.7233 457.3055 510.6697 563.8515 616.8777 669.7692 722.5423
                700     750     800     850     900             950            1000
chisq 0.025 628.5772 676.0026 723.5126 771.0993 818.7560          866.4769          914.2572
chisq 0.05  639.6130 687.4522 735.3623 783.3369 831.3702          879.4574          927.5944
chisq 0.95  762.6607 814.8215 866.9114 918.9369 970.9036         1022.8164         1074.6794
chisq 0.975 775.2107 827.7853 880.2753 932.6887 985.0320         1037.3111         1089.5309
```

(End of R Output)