MSc Applied Statistics Programmes: Statistical Analysis (Spring Term)

Generalized Linear Models:Worked Examples/Case Studies

The following 7 worked examples/case studies all involve generalized linear models and should be worked through over the next $2-4$ weeks. It would be sensible to try to reproduce the results yourself using R/S-Plus. Usually the relevant code is provided, but sometimes this is left for you to supply. The appendix gives general information about R code for generalized linear models.

You should pay particular attention to the *choice* and *interpretation in context* of the fitted models. If you have difficulties please make an appointment for a one-to-one tutorial with the lecturer. Example 7 should be left until log-linear models for three-way contingency tables have been covered in lectures.

Examples 1 and 2 involve count data and use poisson regression. Examples 3 and 4 involve lifetime data (in some sense): example 3 uses the exponential distribution (the gamma distribution with scale parameter 1), while example 4 uses the general gamma distribution, but investigates the possibility that an exponential distribution is sufficient. Example 5 looks at binary data modelled by the Bernoulli distribution (i.e. the binomial distribution with one trial). You should note that for Bernoulli data the deviance can be shown to be degenerate and so cannot be used to carry out a goodness-of-fit test. It is still possible, however, to use *changes* in deviance to test the effect of adding or removing terms from the linear predictor. Example 6 compares binomial and poisson regression (two ways of approaching the same problem). Example 7 is a case study using a 3-way contingency table of counts.

Example 1: (Count Data)

Evans A W (January 2007). Consider the following data on occurrences of fatal train collisions, derailments and overruns in the UK from 1967 to 2006. The accidents are categorized into 5-year time periods, and into whether they would have been prevented if Automatic Train Protection had been installed.

	ATP-preventable		
Period	Yes	No	**Total**
1967-1971	9	16	25
1972-1976	7	7	14
1977-1981	5	5	10
1982-1986	3	8	11
1987-1991	4	6	10
1992-1996	2	4	6
1997-2001	2	1	3
2002-2006	0	1	1
Total	32	48	80

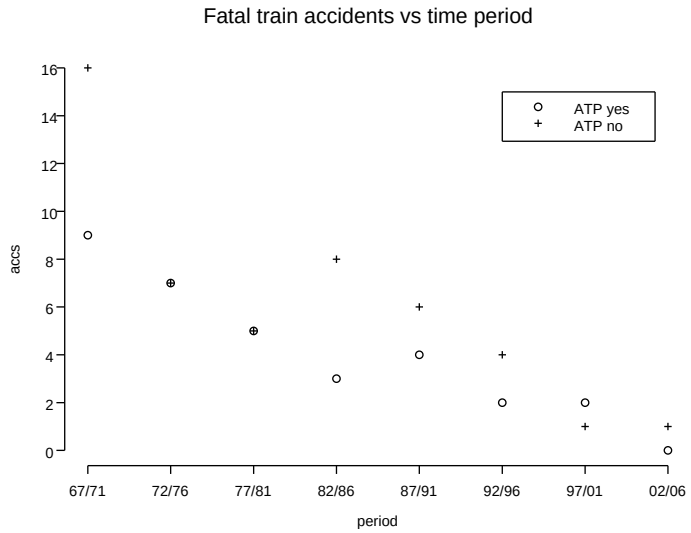
The following R code reads in these data and plots them.

```
period <- rep(seq(1969,2004,5),2)
period.adj <- period - 1969
ATPprevent <- factor(c(rep("yes",8),rep("no",8)),levels=c("yes","no"))
accs <- c(9,7,5,3,4,2,2,0,16,7,5,8,6,4,1,1)
railaccs <- data.frame(period,period.adj,ATPprevent,accs)
levels(ATPprevent)
rm(period,period.adj,ATPprevent,accs)

attach(railaccs)
plot(period, accs, type="n", axes=F,
main="Fatal train accidents vs time period")
axis(1,at=seq(1969,2004,5),labels=c("67/71","72/76","77/81","82/86",
"87/91","92/96","97/01","02/06"))
axis(2,at=seq(0,16,2))
points(period[ATPprevent=="yes"], accs[ATPprevent=="yes"], pch=1)
points(period[ATPprevent=="no"], accs[ATPprevent=="no"], pch=3)
legend(1994,15,c("ATP yes","ATP no"), pch=c(1,3))
```

yielding the output

```
> period <- rep(seq(1969, 2004, 5), 2)
> period.adj <- period - 1969
> ATPprevent <- factor(c(rep("yes", 8), rep("no", 8)), levels = c("yes","no"))
> accs <- c(9, 7, 5, 3, 4, 2, 2, 0, 16, 7, 5, 8, 6, 4, 1, 1)
> railaccs <- data.frame(period, period.adj, ATPprevent, accs)
> levels(ATPprevent)
[1] "yes" "no"
> rm(period, period.adj, ATPprevent, accs)

> attach(railaccs)
> plot(period, accs, type = "n", axes = F, main = "Fatal train accidents vs time period")
> axis(1, at = seq(1969, 2004, 5), labels = c("67/71", "72/76", "77/81",
    "82/86", "87/91", "92/96", "97/01", "02/06"))
```

```
> axis(2, at = seq(0, 16, 2))
> points(period[ATPprevent == "yes"], accs[ATPprevent == "yes"], pch = 1)
> points(period[ATPprevent == "no"], accs[ATPprevent == "no"], pch = 3)
> legend(1994, 15, c("ATP yes", "ATP no"), pch = c(1, 3))
```



The following code

```
railaccs.glm1 <- glm(accs ~ period.adj + ATPprevent + ATPprevent:period.adj,
                             data = railaccs, family = poisson(link=log))
options(digits=4)
summary(railaccs.glm1)
dummy.coef(railaccs.glm1)
anova(railaccs.glm1)
```

fits a generalized linear model for these data, in which the data are assumed to be Poisson, and a logarithmic link function (the canonical link) is used. On the scale of the linear predictor the full model is fitted, in which the single available explanatory variable is time (`period.adj`), but a separate intercept and slope are allowed for ATP-preventable accidents and non-ATP-preventable accidents. The resulting output follows:

```
> railaccs.glm1 <- glm(accs ~ period.adj + ATPprevent + ATPprevent:period.adj,
+                              data = railaccs, family = poisson(link=log))
> options(digits=4)
> summary(railaccs.glm1)

Call:
glm(formula = accs ~ period.adj + ATPprevent + ATPprevent:period.adj,
    family = poisson(link = log), data = railaccs)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.4392  -0.6328   0.0177   0.5501   0.9802

Coefficients:
                      Estimate Std. Error z value Pr(>|z|)
(Intercept)            2.24148    0.25154    8.91  < 2e-16 ***
period.adj            -0.06304    0.01789   -3.52  0.00043 ***
ATPpreventno           0.38409    0.32550    1.18  0.23800
```

```
period.adj:ATPpreventno  0.00212    0.02302    0.09  0.92678
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 46.5268  on 15  degrees of freedom
Residual deviance:  8.5813  on 12  degrees of freedom
AIC: 66.18

Number of Fisher Scoring iterations: 4

> dummy.coef(railaccs.glm1)
Full coefficients are

(Intercept):                2.241
period.adj:                -0.06304
ATPprevent:                     yes       no
                            0.0000   0.3841
period.adj:ATPprevent:          yes       no
                          0.000000 0.002115
> anova(railaccs.glm1)
Analysis of Deviance Table

Model: poisson, link: log

Response: accs

Terms added sequentially (first to last)


                     Df Deviance Resid. Df Resid. Dev
NULL                                    15       46.5
period.adj            1     34.7        14       11.8
ATPprevent            1      3.2        13        8.6
period.adj:ATPprevent 1      0.0        12        8.6
```

Here we have a naturally hierarchical sequence of models, and the analysis of deviance table allows us to investigate the appropriate model. It is clear that, apart from the null model, all three models give a good fit to the data. (Testing 11.8 as χ^2_{14}, 8.6 as χ^2_{13}, and 8.6 as χ^2_{12}.) Clearly the interaction term is non-significant (testing the change in deviance, 0.01, as χ^2_1). So we now try the model with two intercepts and a single slope - on the scale of the linear predictor.

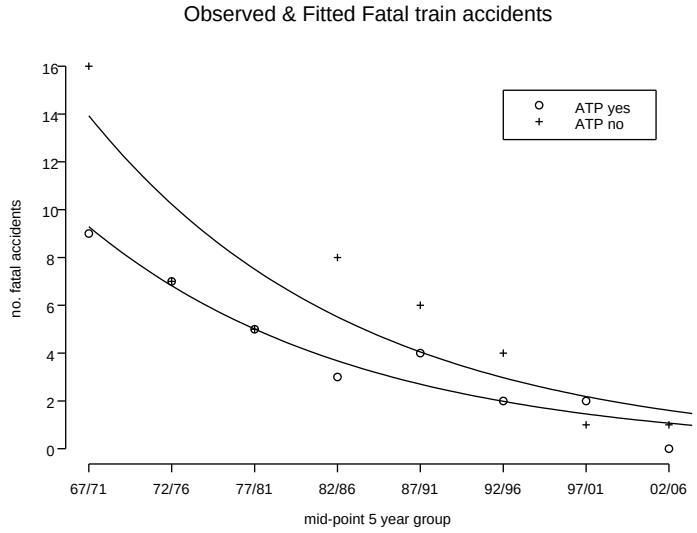
```
> railaccs.glm2 <- glm(accs ~ period.adj + ATPprevent, data = railaccs,
+     family = poisson(link = "log"))
> summary(railaccs.glm2)

Call:
glm(formula = accs ~ period.adj + ATPprevent, family = poisson(link = "log"),
    data = railaccs)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.4622  -0.6088   0.0044   0.5472   0.9904

Coefficients:
```

```
             Estimate Std. Error z value Pr(>|z|)
(Intercept)    2.2287     0.2103   10.60  < 2e-16 ***
period.adj    -0.0618     0.0113   -5.49  4.1e-08 ***
ATPpreventno   0.4055     0.2282    1.78    0.076 .
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 46.5268  on 15  degrees of freedom
Residual deviance:  8.5898  on 13  degrees of freedom
AIC: 64.19

Number of Fisher Scoring iterations: 4

>
> dummy.coef(railaccs.glm2)
Full coefficients are

(Intercept):       2.229
period.adj:     -0.06177
ATPprevent:          yes     no
                  0.0000 0.4055
>
> anova(railaccs.glm2)
Analysis of Deviance Table

Model: poisson, link: log

Response: accs

Terms added sequentially (first to last)


           Df Deviance Resid. Df Resid. Dev
NULL                          15       46.5
period.adj  1     34.7        14       11.8
ATPprevent  1      3.2        13        8.6
```

The results suggest that, *if* there is a significant difference between the level of ATP-preventable accidents and non-ATP-preventable accidents, then non-ATP-preventable accidents are $\exp 0.4055 = 1.5$ times more likely than ATP-preventable accidents. That is, we would predict that there are 50% more non-ATP-preventable accidents than there are ATP-preventable ones. Nevertheless the decline over time is strongly significant for both types of fatal accident, and we would predict that from one 5-year period to the next, numbers of accidents would be multiplied by $\exp(-5 \times 0.06177) = 0.734$, a reduction of 26.6%. The following code plots the observed and fitted values using this model.

```
> Yr5Grp <- rep(seq(0, 40, length = 100), 2)
> Acctype <- factor(c(rep("yes", 100), rep("no", 100)),
                        levels = c("yes", "no"))
> Fittedaccsper5yrs <- predict(railaccs.glm2, type = "response",
     newdata = data.frame(period.adj = Yr5Grp, ATPprevent = Acctype))
> plot(period.adj, accs, type = "n", axes = F,
        xlab = "mid-point 5 year group", ylab = "no. fatal accidents",
        main = "Observed & Fitted Fatal train accidents")
```

```
> axis(1, at = seq(0, 35, 5), labels = c("67/71", "72/76", "77/81", "82/86",
    "87/91", "92/96", "97/01", "02/06"))
> axis(2, at = seq(0, 16, 2))
> points(period.adj[ATPprevent == "yes"], accs[ATPprevent == "yes"], pch = 1)
> points(period.adj[ATPprevent == "no"], accs[ATPprevent == "no"], pch = 3)
> lines(Yr5Grp[1:100], Fittedaccsper5yrs[1:100])
> lines(Yr5Grp[101:200], Fittedaccsper5yrs[101:200])
> legend(25, 15, c("ATP yes", "ATP no"), pch = c(1, 3))
```



But is there really a significant difference between the occurrence of non-ATP-preventable accidents and ATP-preventable ones? Testing the effect of the different intercepts we obtain

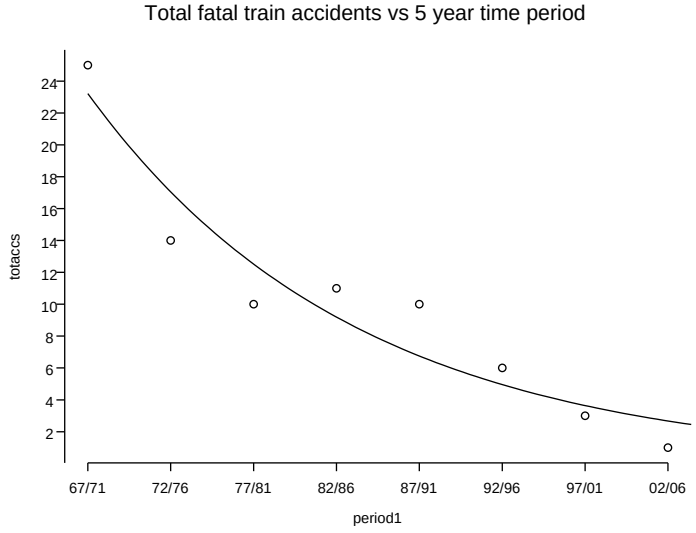
```
> pchisq(3.22, 1, lower.tail=F)
[1] 0.07274
```

This is NS at the 5% level, but significant at the 10% level. A matter for the judgement of the decision maker. If it is proposed that Automatic Train Protection equipment be installed (a large investment) then it may be that we wish to retain a model that distinguishes between accidents that would be prevented by ATP, and those that would not. Moreover there is no *a priori* reason to suggest that these two types of accident are equally likely, which is the null hypothesis here.

However, if we decide to forget the difference between non-ATP-preventable and ATP-preventable accidents (assuming they are equally likely) then we can amalgamate the data and fit the following overall model.

```
> totaccs <- accs[ATPprevent == "yes"] + accs[ATPprevent == "no"]
> period1 <- seq(0, 35, 5)
> railaccs.glm3 <- glm(totaccs ~ period1, family = poisson(link = log))
> summary(railaccs.glm3)

Call:
glm(formula = totaccs ~ period1, family = poisson(link = log))

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.1744  -0.7442   0.0096   0.4845   1.1664
```

```
Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)   3.1450     0.1596   19.70  < 2e-16 ***
period1      -0.0618     0.0113   -5.49  4.1e-08 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 39.3737  on 7  degrees of freedom
Residual deviance:  4.6583  on 6  degrees of freedom
AIC: 39.42

Number of Fisher Scoring iterations: 4

> pchisq(4.658, 6, lower.tail=F)
[1] 0.5884

> Yr5Grpshort <- seq(0, 40, length = 100)
> Fittedtotaccsper5yrs <- predict(railaccs.glm3, type = "response", newdata
     = data.frame(period1 = Yr5Grpshort))
> plot(period1, totaccs, type = "n", axes = F, main =
    "Total fatal train accidents vs 5 year time period")
> axis(1, at = seq(0, 35, 5), labels = c("67/71", "72/76", "77/81", "82/86",
    "87/91", "92/96", "97/01", "02/06"))
> axis(2, at = seq(0, 26, 2))
> points(period1, totaccs)
> lines(Yr5Grpshort, Fittedtotaccsper5yrs)
```

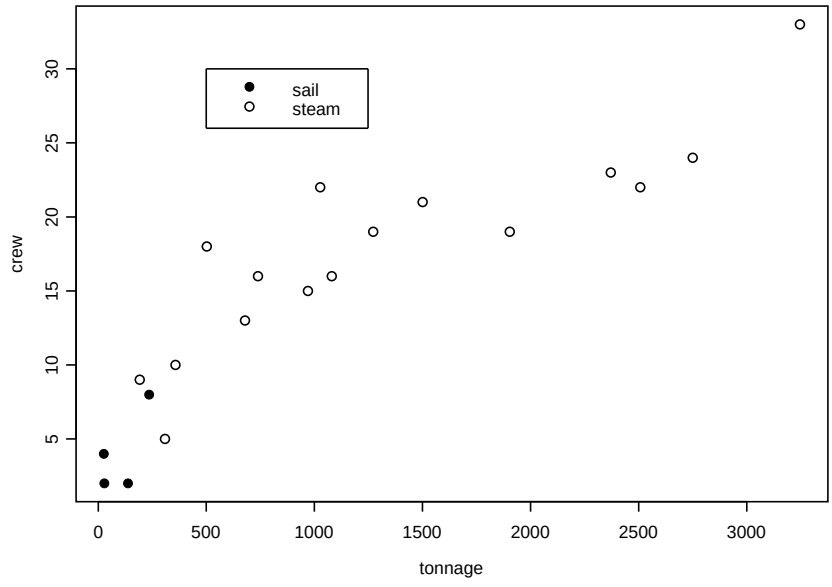


Example 2: (Count Data)

Floud (1973). The following dataset, Floud (1973), gives the tonnage, size of crew and type of power (sail or steam) for 20 British merchant ships in 1907. Interest centres on how crew size depends on tonnage, and whether the relationship is different for different types of power.

tonnage	crew	power	tonnage	crew	power
236	8	sail	357	10	steam
739	16	steam	1080	16	steam
970	15	steam	1027	22	steam
2371	23	steam	28	2	sail
309	5	steam	2507	22	steam
679	13	steam	138	2	sail
26	4	sail	502	18	steam
1272	19	steam	1501	21	steam
3246	33	steam	2750	24	steam
1904	19	steam	192	9	steam

The data are shown in the scatterplot below.



There seems to be a fairly strong, if non-linear relationship between crew size and tonnage. There are just four sailing ships as opposed to sixteen steam ships, and it is very noticeable that the sailing ships are all small (they comprise four of the smallest five ships in terms of either crew size or tonnage).

Since the response variable comprises counts (of crew members), it is natural to use Poisson regression to analyse these data. This is carried out in R as follows:

```
> ton <-
c(236,739,970,2371,309,679,26,1272,3246,1904,357,1080,1027,28,2507,138,502,1501,
      2750,192)
> crew <- c(8,16,15,23,5,13,4,19,33,19,10,16,22,2,22,2,18,21,24,9)
> power <-
c("sail",rep("steam",5),"sail",rep("steam",6),"sail","steam","sail",
        rep("steam",4))
> levels(factor(power))
[1] "sail"  "steam"

> ship <- data.frame(ton,crew,power)

> ship.glm <- glm(crew ~ ton + factor(power), family = poisson(link = log))
> summary(ship.glm)

Call:
glm(formula = crew ~ ton + factor(power), family = poisson(link = log))

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.3270  -0.6121  -0.0899   0.5398   1.6626

Coefficients:
                   Estimate Std. Error z value Pr(>|z|)
(Intercept)        1.35e+00   2.50e-01    5.40  6.6e-08 ***
ton                3.26e-04   6.03e-05    5.41  6.5e-08 ***
factor(power)steam 1.05e+00   2.73e-01    3.83  0.00013 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 99.990  on 19  degrees of freedom
Residual deviance: 18.039  on 17  degrees of freedom
AIC: 110.8

Number of Fisher Scoring iterations: 4
```

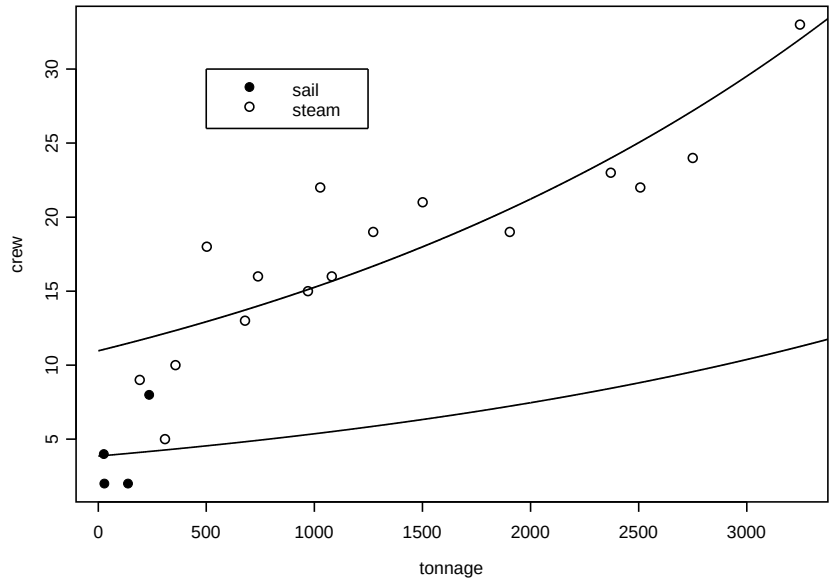
This gives a fitted response for Y, the numbers of crew, for the sail powered ships of

$$\log(E[Y]) = 1.35 + 0.00033\text{tonnage} \text{ or } E[Y] = \exp(1.35 + 0.00033\text{tonnage})$$

and for the steam powered ships of,

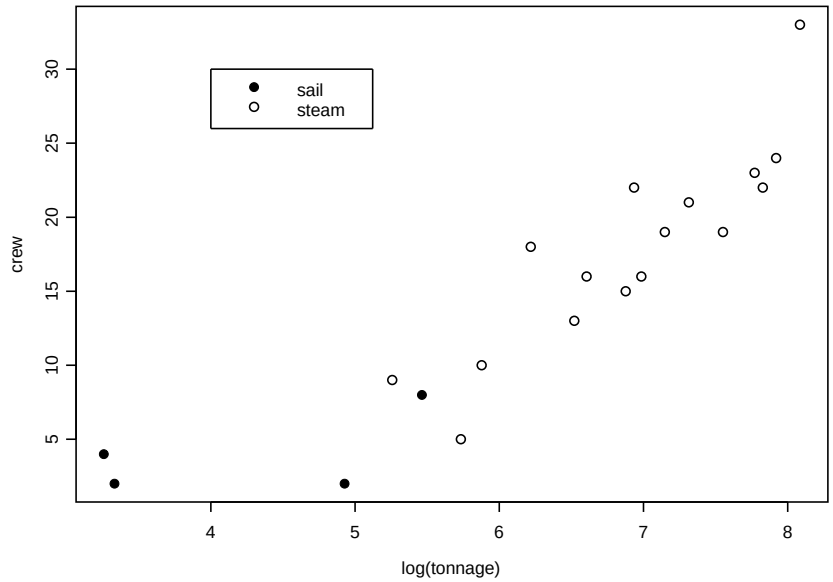
$$\log(E[Y]) = 1.35 + 1.05 + 0.00033\text{tonnage} \text{ or } E[Y] = \exp(2.395 + 0.00033\text{tonnage})$$

The fitted response curves are shown on the following plot.



There are certain disquieting features in this plot. The most obvious is the enormous degree of extrapolation going on in producing the whole of the lower curve from just four observations on sailing ships which are grouped together towards the left-hand end. Also, for the steam ships, while the fitted response curve increases its slope as you move to the right, you could well argue that the steam ship data (with the possible exception of the very largest ship) follows a curve that is the other way up. A better model is called for.

A *log* transformation would pull the smaller tonnages apart and bring the larger tonnages closer together, and this would work in the right direction. A scatterplot of crew size versus log(tonnage) is shown below.



The revised model is fitted in R as follows:

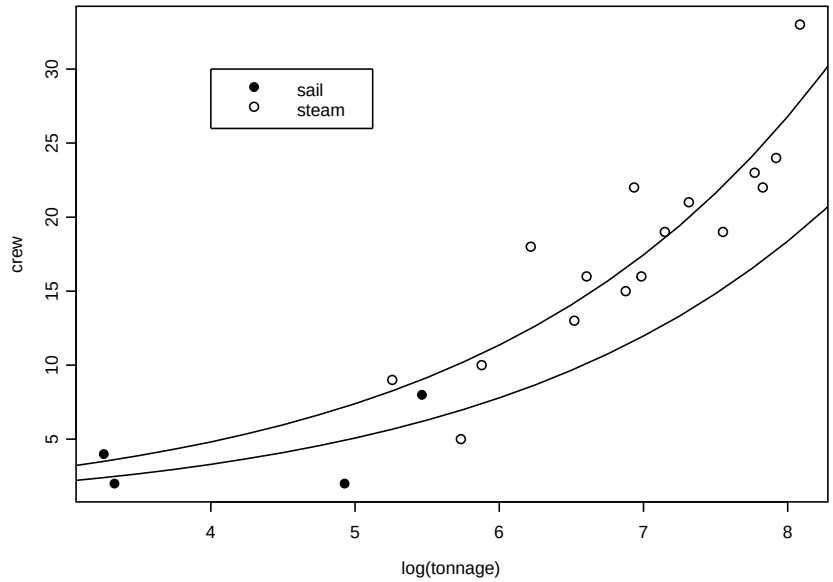
```
> logton <- log(ton)

> ship.glm <- glm(crew ~ logton + factor(power), family = poisson(link = log))
> summary(ship.glm)

Call:
glm(formula = crew ~ logton + factor(power), family = poisson(link = log))

Deviance Residuals:
   Min      1Q  Median      3Q     Max
-1.790  -0.379  -0.253   0.418   1.465

Coefficients:
                   Estimate Std. Error z value Pr(>|z|)
(Intercept)          -0.521      0.436   -1.19     0.23
logton                0.429      0.077    5.57  2.5e-08 ***
factor(power)steam    0.377      0.323    1.17     0.24
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 99.990  on 19  degrees of freedom
Residual deviance: 13.004  on 17  degrees of freedom
AIC: 105.8

Number of Fisher Scoring iterations: 4

> anova(ship.glm)
Analysis of Deviance Table

Model: poisson, link: log

Response: crew

Terms added sequentially (first to last)


              Df Deviance Resid. Df Resid. Dev
NULL                              19      100.0
logton         1     85.6        18       14.4
factor(power)  1      1.4        17       13.0
```

The fitted response curves are shown below. For the sailing ships, we have

$$E[Y] = \exp(-0.521 + 0.429\log(\text{tonnage}))$$

and for the steam powered ships

$$E[Y] = \exp(-0.144 + 0.429\log(\text{tonnage})).$$



It is noticeable that the two response curves are rather similar and quite close together. Perhaps there is a case for unifying steam and sailing ships and providing just a single response curve.

To compare the models with and without the factor power, we refer to the analysis of deviance table in the above printout.
The change in deviance from adding this model term is 14.42325-13.00395=1.4193, which should be compared with the $\chi^2(1)$ distribution.

```
> p.value.approx <- pchisq(1.4193, 1, lower.tail=F)
> p.value.approx
[1] 0.2335
```

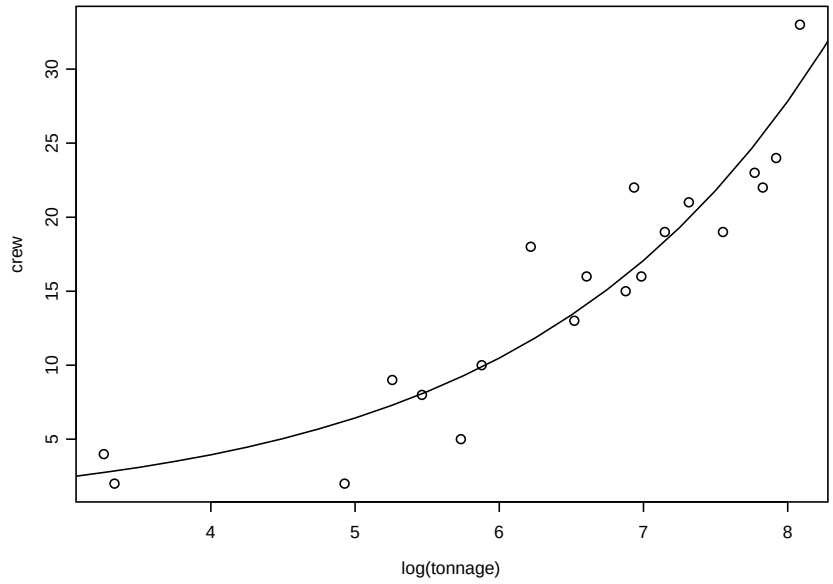
Hence, there is no significant difference between these models, so that the simpler model, without the power term is preferred. That is, there do not seem to be different relationships for steam and sailing ships (at least not given the very small amount of data on sailing ships that we have).

Fitting this simpler model in R is carried out as follows:

```
> ship.glm <- glm(crew ~ logton, family = poisson(link = log))
> summary(ship.glm)

Call:
glm(formula = crew ~ logton, family = poisson(link = log))

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.9677  -0.5252  -0.0745   0.5360   1.7267

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -0.5780     0.4232   -1.37     0.17
```

```
logton        0.4875     0.0596    8.18  2.8e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 99.990  on 19  degrees of freedom
Residual deviance: 14.423  on 18  degrees of freedom
AIC: 105.2

Number of Fisher Scoring iterations: 4
```

(Note the residual deviance is 14.423 on 18 d.f., which is clearly not significant. Hence, this model would appear to provide an adequate fit to the data). The fitted response curve for the overall data is shown below.



The fitted response is given by

$$E[Y] = \exp(-0.578 + 0.488\log(\text{tonnage})) = \exp(-0.578)(\text{tonnage})^{0.488}$$
$$= 0.561(\text{tonnage})^{0.488} \approx 0.5(\text{tonnage})^{0.5}$$

That is the mean crew size (for ships in 1907) is, approximately, $\frac{1}{2}\sqrt{\text{tonnage}}$.

Example 3: (Lifetime Data)

Feigl and Zelen (1965). The data in the following table give the time to death, Y, in terms of weeks from diagnosis, and $\log_{10}$(initial white blood cell count), x, for 17 patients suffering from leukemia.

x	Y
3.36	65
2.88	156
3.63	100
3.41	134
3.78	16
4.02	108
4.00	121
4.23	4
3.73	39
3.85	143
3.97	56
4.51	26
4.54	22
5.00	1
5.00	1
4.72	5
5.00	65

We wish to relate Y to x. These data were analysed by Cox and Snell (1981). Survival time was modelled as an exponential distribution $Exp(\lambda)$ (with constant hazard function $h(x) = f(x)/(1-F(x)) = \lambda$), where $f(x)$ is the exponential p.d.f. and $S(x) = 1-F(x)$ is the survivor function, the probability that $X \geq x$. The parameter λ was permitted to vary with the value x, $\log_{10}$(white blood cell count at diagnosis). Mean survival time, $\mathbb{E}[Y] = \mu$, was modelled as an exponential function of x, giving the generalized linear model with logarithmic link function:

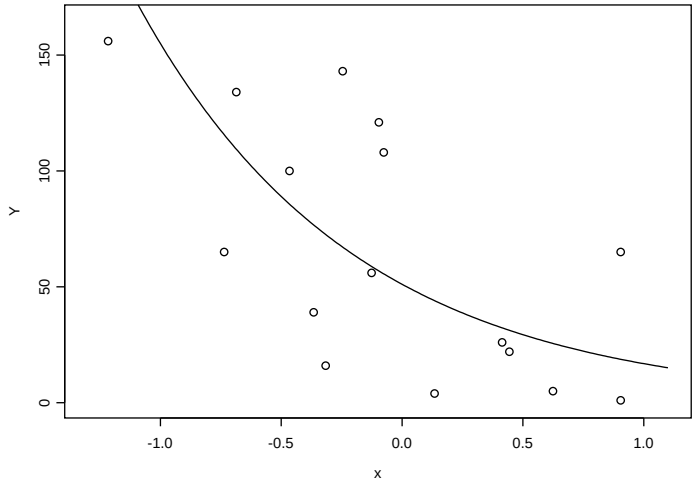
$$\log(\mu) \;=\; \beta_0 + \beta_1 x.$$

Note that this is *not* the canonical link function for the exponential distribution.

Fitting the model in R gives

```
> Y <- c(65, 156, 100, 134, 16, 108, 121, 4, 39, 143, 56, 26, 22, 1, 1, 5, 65)
> x <- c(3.36, 2.88, 3.63, 3.41, 3.78, 4.02, 4.0, 4.23, 3.73, 3.85, 3.97,
         4.51, 4.54, 5.0, 5.0, 4.72, 5.0)
> x <- x - mean(x)

> survive.glm <- glm(Y ~ x, family = Gamma(link = log))
# Gamma distribution, log link
> summary(survive.glm, dispersion = 1)
# dispersion parameter set to 1 for Exponential distribution

Call:
glm(formula = Y ~ x, family = Gamma(link = log))
```

```
Deviance Residuals:
   Min      1Q  Median      3Q     Max
-1.992  -1.210  -0.224   0.210   1.565

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)    3.934      0.243   16.22   <2e-16 ***
x             -1.109      0.400   -2.78   0.0055 **
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

(Dispersion parameter for Gamma family taken to be 1)

    Null deviance: 26.282  on 16  degrees of freedom
Residual deviance: 19.457  on 15  degrees of freedom
AIC: 174

Number of Fisher Scoring iterations: 8

> pchisq(19.46, 15, lower.tail=F)
[1] 0.1936

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

Model: Gamma, link: log

Response: Y

Terms added sequentially (first to last)


     Df Deviance Resid. Df Resid. Dev Pr(>Chi)
NULL                    16       26.3
x     1     6.83        15       19.5    0.007 **
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

> newx <- seq(-1.3, 1.1, 0.01)
> FittedY <- predict(survive.glm, type = "response", data.frame(x = newx))
> plot(x, Y, xlim = c(-1.3, 1.1), ylim = c(0, 165))
> lines(newx, FittedY)
```



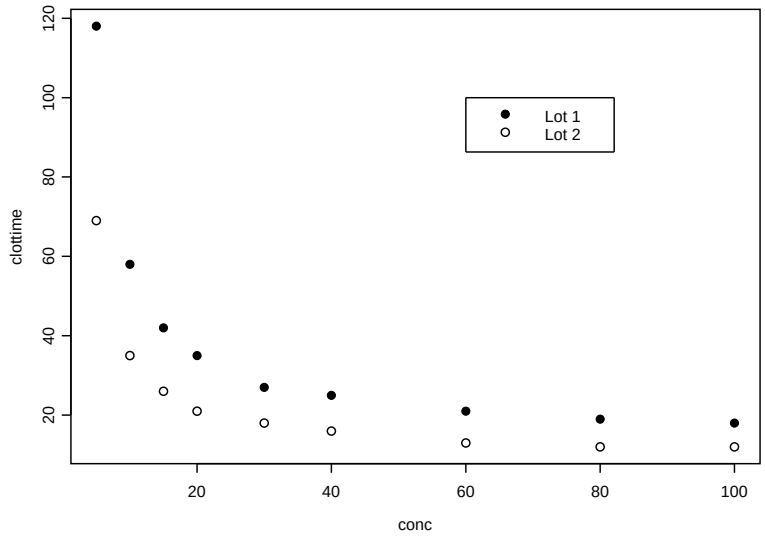
Clearly the goodness-of-fit test would not reject this model, and the effect of white blood cell count at diagnosis on average further survival time is strongly significant. Try re-running the model using the canonical link, the reciprocal. Which model do you prefer and why? [Note that residual plots might help here. See the final lecture on generalized linear models.]

Example 4: (Lifetime Data)

Hurn *et al* (1945) reported an experiment which investigated the relationship between normal blood plasma concentration in blood and clotting time. Samples of normal blood plasma were diluted to nine different percentage concentrations, using plasma free of the clotting factor prothrombin. Clotting was then induced in each of the samples using two different lots of thromboplastin, and the clotting time in seconds was recorded.

The data were entered into S-Plus as three vectors: a variable `conc` giving the concentration of normal plasma (as a percentage), a factor `lot` at two levels (1 and 2) denoting the lot of thromboplastin, and a variable `clottime` giving the clotting time in seconds. Treating `clottime` as the response variable, a plot of `clottime` versus `conc` is shown below for the different lots.

Clotting time v's concentration



An appropriate model for these data was considered to be a gamma regression with a reciprocal link function (the canonical link). To obtain a good fit it was necessary to transform `conc` by taking logs (to give `lconc`). Within this framework, three different models (A, B and C) were fitted, and the resulting output is as shown below.

```
> lconc <- log(conc)
```

Model A

```
> modA.glm <- glm(clottime ~ lconc, family = Gamma)
> summary(modA.glm)
Deviance Residuals:
       Min         1Q      Median        3Q       Max
 -0.321785 -0.2331564 -0.01690691 0.1970125 0.3129572
```

```
Coefficients:
                  Value  Std. Error    t value
(Intercept) -0.01963451 0.004127218  -4.757323
      lconc  0.01860894 0.001826701  10.187185

(Dispersion Parameter for Gamma family taken to be 0.0621964 )
    Null Deviance: 7.708668 on 17 degrees of freedom
Residual Deviance: 1.01826 on 16 degrees of freedom
```

Model B

```
> modB.glm <- glm(clottime ~ lconc + lot, family = Gamma)
> summary(modB.glm)

Deviance Residuals:
        Min         1Q     Median         3Q       Max
 -0.2073445 -0.1221923 0.05503781 0.09440285 0.2246345

Coefficients:
                  Value  Std. Error    t value
(Intercept) -0.02144969 0.002184986  -9.816856
      lconc  0.01775637 0.001022044  17.373393
        lot  0.01086845 0.001947636   5.580329

(Dispersion Parameter for Gamma family taken to be 0.0195687 )
    Null Deviance: 7.708668 on 17 degrees of freedom
Residual Deviance: 0.3004207 on 15 degrees of freedom
```

Model C

```
> modC.glm <- glm(clottime ~ lconc + lot + lconc:lot, family = Gamma)
> summary(modC.glm)

Deviance Residuals:
         Min          1Q       Median        3Q        Max
 -0.05573777 -0.03547972 -0.008216152 0.0260727 0.08641118

Coefficients:
                   Value   Std. Error    t value
(Intercept) -0.016554382 0.0008654840 -19.127311
      lconc  0.015343115 0.0003871945  39.626377
        lot -0.007354088 0.0016779383  -4.382812
  lconc:lot  0.008256099 0.0007352771  11.228555

(Dispersion Parameter for Gamma family taken to be 0.0021297 )
    Null Deviance: 7.708668 on 17 degrees of freedom
Residual Deviance: 0.0294015 on 14 degrees of freedom

> dummy.coef(modC.glm)
$"(Intercept)":
 (Intercept)
 -0.01655438

$lconc:
      lconc
 0.01534311
```

```
$lot:
 1            2
 0 -0.007354088

$"lconc:lot":
 lconc1      lconc2
      0 0.008256099
```

(i) What model is being fitted in each of models A, B and C?

In all these models it is assumed that clotting time has a gamma distribution with mean equal to the reciprocal of a linear function of the explanatory variables. In model A, the explanatory variable is `lconc`. In model B, the explanatory variables are `lconc` and an indicator variable `lot`, for the lot of thromboplastin. In model C, an interaction term is added to the terms of model B.

[Strictly speaking, a gamma distribution has two parameters. We should add the assumption that the variance of the gamma distribution is proportional to the square of its mean, which is what assuming a constant 'dispersion parameter' implies].

(ii) By reference to the residual deviances for the three models, we can explain why model C is the most appropriate to model these data.

Note that the dispersion parameter is being estimated here ($D/(N-p)$), thus deviance ratios are relevant rather than the differences.

Model	Residual Deviance	d.f.	Deviance ratio (c/f Previous Model)	F-dist.
A	1.01826	16		
B	0.30042	15	$\frac{1.01826-0.30042}{0.30042/15}=\frac{0.71784}{0.02003}=35.84$	F(1,15)
C	0.02940	14	$\frac{0.30042-0.02940}{0.02940/14}=\frac{0.27102}{0.00210}=129.06$	F(1,14)

We begin by comparing models B and C.

The deviance ratio for the change in going from model B to model C is much greater than 1, so there is evidence that the interaction term is needed. (No need for tables here!) Model C is therefore to be preferred to model B and thus also to model A which is simpler still.

(iii) The experimenters wanted to know whether the relationship between normal plasma concentration and clotting time differed between the two lots of thromboplastin.

The significant interaction term in model C implies that the relationship is different for the two lots. According to this model, reciprocal mean clotting time increases by 0.008256 per unit of `lconc` more with lot 2 material than with lot 1.

(iv) For model C, we can now write down the fitted equations for the mean clotting time, separately for Lot 1 and Lot 2. The fitted model can be used to predict the mean clotting time in different circumstances. For example, what would be the mean clotting time for a normal plasma concentration of 50%using thromboplastin from Lot 2?

For lot 1, the fitted equation is

$$E[\texttt{clottime}] = \frac{1}{-0.016554 + 0.015343\texttt{lconc}}$$

And, the fitted equation for lot 2 is

$$\begin{aligned} E[\texttt{clottime}] &= \frac{1}{(-0.016554 - 0.007354) + (0.015343 + 0.008256)\texttt{lconc}} \\ &= \frac{1}{-0.023908 + 0.023599\texttt{lconc}} \end{aligned}$$

If the plasma concentration was 50% and lot 2 was used, we would predict a clotting time of

$$\frac{1}{-0.023908 + 0.023599\log(50)} = \frac{1}{0.0684} = 14.62 \text{ seconds}$$

[Check that this looks about right on the graph.]

(v) On the basis of the outputs given, would we expect that exponential regression would provide a suitable model for these data?

Since exponential regression is a special case of gamma with the dispersion parameter equal to 1, we look at the estimated scale (dispersion) parameter. According to the fit of model C, the preferred model, the estimated dispersion parameter is 0.00210, which is not close to 1, so that exponential regression would not be appropriate.

Example 5: Challenger Mission
Dalal et. al (1989)

Data on the 23 space shuttle flights that occurred before the Challenger mission in 1986 are given in the following table. For each of the 23 missions, data on the temperature, in ^oF, at the time of flight (Temp.), and whether at least one primary O-ring suffered thermal distress (TD) were recorded.

Temp	TD	Temp	TD	Temp	TD
66	0	57	1	70	0
70	1	63	1	81	0
69	0	70	1	76	0
68	0	78	0	79	0
67	0	67	0	75	1
72	0	53	1	76	0
73	0	67	0	58	1
70	0	75	0		

We fit a logistic regression model between TD and Temp.

```
> td <- c(0, 1, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 1)
> temp <- c(66, 70, 69, 68, 67, 72, 73, 70, 57, 63, 70, 78, 67, 53, 67, 75, 70,
  + 81, 76, 79, 75, 76, 58)
> shuttle <- data.frame(td, temp)
> shuttle.glm <- glm(td ~ temp, family = binomial, data = shuttle)
> summary(shuttle.glm)

Call:
glm(formula = td ~ temp, family = binomial, data = shuttle)

Deviance Residuals:
   Min      1Q  Median      3Q     Max
-1.061  -0.761  -0.378   0.452   2.217

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)   15.043      7.379    2.04    0.041 *
temp          -0.232      0.108   -2.14    0.032 *
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 28.267  on 22  degrees of freedom
Residual deviance: 20.315  on 21  degrees of freedom
AIC: 24.32

Number of Fisher Scoring iterations: 5

> anova(shuttle.glm)
Analysis of Deviance Table

Model: binomial, link: logit
```

```
Response: td

Terms added sequentially (first to last)


     Df Deviance Resid. Df Resid. Dev
NULL                    22       28.3
temp  1     7.95        21       20.3

> pchisq(7.95196, 1, lower.tail=F)
[1] 0.004804
```

Notice that the `weights` argument has not been specified since we are dealing with Bernoulli response data, i.e. binary data.
The residual deviance is 20.315 on 21 degrees of freedom. This is not relevant since it can be shown that with Bernoulli response data the residual deviance is degenerate, makes no comparison between observed and predicted responses, and cannot be used as the basis of a goodness-of-fit test. However we can see that adding `temp` to the null model gives a change in deviance of 7.95 for 1 degree of freedom. This is significant if tested as χ^2 with 1 degree of freedom ($p = 0.0048$), and we can reasonably reject the null hypothesis that temperature has no effect on the probability of thermal distress.

Assuming, for the sake of argument, that the model does provide a good fit, then we see that

$$\widehat{\pi} = \frac{\exp(\widehat{\beta}_1 + \widehat{\beta}_2 x)}{1 + \exp(\widehat{\beta}_1 + \widehat{\beta}_2 x)} \tag{1}$$

where $\widehat{\beta}_1 = 15.043$, $\widehat{\beta}_2 = -0.232$, x is the temperature at the time of flight, and $\widehat{\pi}$ is the (corresponding) estimated probability of thermal distress of at least one primary O-ring.

Question: What is the predicted probability of thermal distress at 31^oF (supposedly the temperature at the time of the Challenger flight)?

Setting $x = 31$ in (1) yields

$$\widehat{\pi} = \frac{\exp(15.043 - 0.232 \times 31)}{1 + \exp(15.043 - 0.232 \times 31)} = 0.999$$

i.e. it is predicted that thermal distress is almost certain at that temperature. Note however, that we are predicting a long way outside the range of the observed data, always a dangerous thing to do!

Question: At which temperature is it estimated that there is a 50% chance that thermal distress occurs?

Substituting for $\widehat{\pi} = 0.5$ into (1), we find that the corresponding x satisfies

$$\widehat{\beta}_1 + \widehat{\beta}_2 x = 0$$

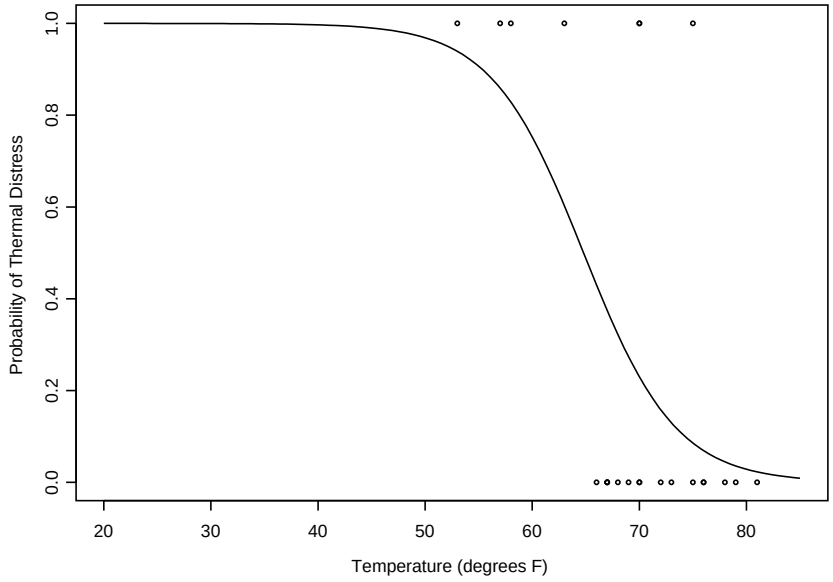
which implies that

$$x = -\frac{\widehat{\beta}_1}{\widehat{\beta}_2} = -\frac{15.043}{-0.232} = 64.794$$

i.e. about $65^o F$.

See below for a plot of the observed data and fitted model.

```
> fittemp <- seq(20, 85, length = 100)
> etashuttle <- coef(shuttle.glm)[1] + coef(shuttle.glm)[2] * fittemp
> fitshuttle <- exp(etashuttle)/(1 + exp(etashuttle))
> ylim <- c(0, 1)
> plot(fittemp, fitshuttle, xlab = "Temperature (degrees F)", ylab =
    "Probability of Thermal Distress", type = "l",
    ylim = ylim, xlim = c(20, 85),lty = 1)
> points(temp, td, cex=0.5)
```



Example 6: Binomial and Poisson Regression
Missing Persons Data, The Independent, March 8 1994

The data in the following table were printed in *The Independent* on March 8 1994 under the headline 'Thousands of people who disappear without trace'. The figures which were recorded by the Metropolitan Police relate to the numbers of persons reported missing during the year ending March 1993 (N), and the number still missing at the end of that year (S).

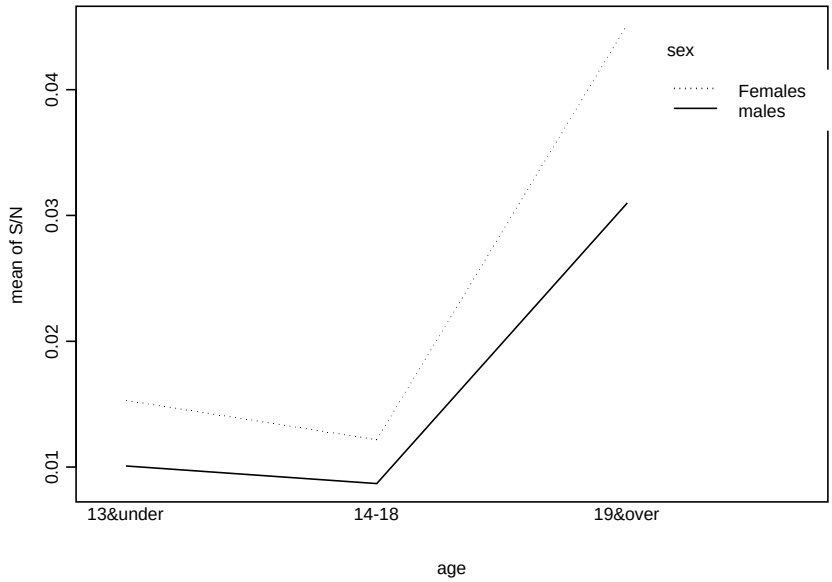
		Males		Females	
		Still Missing (S)	Reported Missing (N)	Still Missing (S)	Reported Missing (N)
Age	≤ 13	33	3271	38	2486
	14-18	63	7257	108	8877
	≥ 19	157	5065	159	3520

We begin by entering the data in R.

```
> S <- c(33, 63, 157, 38, 108, 159)
> N <- c(3271, 7257, 5065, 2486, 8877, 3520)

> sex <- c(rep("males", 3), rep("Females", 3))
> sex <- factor(sex)

> age <- rep(c("13&under", "14-18", "19&over"), 2)
> age <- factor(age)

> missing <- data.frame(S, N, sex, age)
> missing
    S    N     sex      age
1  33 3271   males 13&under
2  63 7257   males    14-18
3 157 5065   males  19&over
4  38 2486 Females 13&under
5 108 8877 Females    14-18
6 159 3520 Females  19&over

> tapply(S/N, list(age, sex), sum)
         Females    males
13&under 0.01529 0.010089
14-18    0.01217 0.008681
19&over  0.04517 0.030997
```

The proportions show that those most likely to stay missing are females in the '19 and over' category.

```
> interaction.plot(age, sex, S/N)
```



We fit the model $S_{ij} \sim \text{Bin}(N_{ij}, \pi_{ij})$ for $1 \leq i \leq 2$, $1 \leq j \leq 3$, so that i, j correspond to the factors *sex* and *age* respectively.

That is, we fit

$$\log\left(\frac{\pi_{ij}}{1-\pi_{ij}}\right) = \mu + \alpha_i + \beta_j$$

for $i = 1, 2$ and $j = 1, 2, 3$ with (as usual) $\alpha_1 = \beta_1 = 0$.

```
> mod.binomial <- glm(S/N ~ age + sex, family = binomial, weights = N)
> summary(mod.binomial)

Call:
glm(formula = S/N ~ age + sex, family = binomial, weights = N)

Deviance Residuals:
      1       2       3       4       5       6
-0.1227  0.1965 -0.0675  0.1162 -0.1484  0.0678

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -4.1845     0.1264  -33.11  < 2e-16 ***
age14-18     -0.1980     0.1424   -1.39     0.16
age19&over    1.1279     0.1325    8.51  < 2e-16 ***
sexmales     -0.3803     0.0869   -4.38  1.2e-05 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 221.81706  on 5  degrees of freedom
```

```
Residual deviance:   0.09832  on 2  degrees of freedom
AIC: 45.1

Number of Fisher Scoring iterations: 3
```

The model fits the data well. There is (therefore) no interaction between *sex* and *age*. The model coefficients and fitted values are shown below.

```
> dummy.coef(mod.binomial)
Full coefficients are

(Intercept):      -4.184
age:            13&under   14-18 19&over
                   0.000  -0.198   1.128
sex:             Females   males
                  0.0000 -0.3803

> fitted(mod.binomial)
       1        2        3        4        5        6
0.010305 0.008469 0.031162 0.015001 0.012340 0.044934

> N * fitted(mod.binomial)
     1      2      3      4      5      6
 33.71  61.46 157.83  37.29 109.54 158.17
```

Recall that the Binomial with large N and small p (probability of success) is approximated by a Poisson distribution with mean Np. We can repeat the above analysis using Poisson regression, with an appropriate 'offset' variable.

That is, we now have $S_{ij} \sim \text{Poisson}(N_{ij}\mu_{ij})$, so that $\log(E[S_{ij}]) = \log(\mu_{ij}) + log(N_{ij})$. We fit

$$\log(\mu_{ij}) = \mu + \log(N_{ij}) + \alpha_i + \beta_j$$

That is, the $\log(N_{ij})$ term enters the `glm( )` as an 'offset'. (A term with fixed coefficient 1).

```
> n <- log(N)
> mod.poisson <- glm(S ~ sex + age + offset(n), family = poisson)

> summary(mod.poisson)

Call:
glm(formula = S ~ sex + age + offset(n), family = poisson)

Deviance Residuals:
      1        2        3        4        5        6
-0.1382   0.1646  -0.0396   0.1307  -0.1244   0.0395

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -4.2021     0.1255  -33.48  < 2e-16 ***
sexmales     -0.3703     0.0857   -4.32  1.6e-05 ***
age14-18     -0.1950     0.1415   -1.38     0.17
age19&over    1.1017     0.1313    8.39  < 2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

```
(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 217.10061  on 5  degrees of freedom
Residual deviance:   0.08189  on 2  degrees of freedom
AIC: 45.21

Number of Fisher Scoring iterations: 3

> dummy.coef(mod.poisson)
Full coefficients are

(Intercept):      -4.202
sex:             Females   males
                  0.0000 -0.3703
age:            13&under   14-18 19&over
                   0.000  -0.195   1.102

> fitted(mod.poisson)
    1     2     3     4     5     6
 33.8  61.7 157.5  37.2 109.3 158.5
```

The Binomial and Poisson models give very similar results, in terms of deviances and parameter estimates.

Example 7: (A Three-way Contingency Table)

In May 1967, a survey of student opinion on the Vietnam War was taken at the University of North Carolina in Chapel Hill (USA). Students were asked to fill in 'ballot papers', stating which of the following four policies they supported.

Policy A: The US should defeat the power of North Vietnam by widespread bombing of its industries, ports and harbours and by land invasion.
Policy B: The US should follow its present policy in Vietnam.
Policy C: The US should de-escalate its military activity, stop bombing North Vietnam, and intensify its efforts to begin negotiation.
Policy D: The US should withdraw its military forces from Vietnam immediately.

The results were published in the student newspaper, and are also given in the table below. They are classified by gender and by year of study (undergraduate years 1-4 or graduate status).

		Response				**Total**
Gender	Year	A	B	C	D	
male	1	175	116	131	17	439
	2	160	126	135	21	442
	3	132	120	154	29	435
	4	145	95	185	44	469
	grad	118	176	345	141	780
female	1	13	19	40	5	77
	2	5	9	33	3	69
	3	22	29	110	6	167
	4	12	21	58	10	101
	grad	19	27	128	13	187

In all, 26% of the male students enrolled at the university at the time, and 17% of the female students, responded to the survey. Though these are actually reasonably high response rates for a survey of this type, it is still not appropriate to treat the students who responded as a representative sample of all students at the university. However, we can investigate how the response varies with gender and year of study, for those students who did respond by fitting log-linear models.

Note these data involve a clearly defined response and some explanatory factors. In some datasets, this means that the resulting data can be analysed using logistic regression instead of by fitting log-linear models. This is not the case here because the response variable takes more than two values. (In fact Multinomial as well as Binomial regression is possible but is beyond the scope of this course.)

Model Fitting and Interpretation

Fitting a log-linear model to these data, including the three main effects and all the two-factor interactions between gender, year and response, leads to a residual deviance of 19.19 on 12 d.f. (p=0.0839), so there is little evidence for the inclusion of the three-way interaction.

This suggests the partial association model (*) may be appropriate, but we should check to see if a more parsimonious model can be found which still provides an appropriate fit to the data.

At the other extreme, a model with only the three main effects gives a residual deviance of 423.83 on 31 d.f. This model (of complete independence) is clearly not an appropriate fit to the data.

[Alternatively, we test the change in deviance of $423.83 - 19.19 = 404.64$ on $31 - 12 = 19$ d.f., $p = 0.0000$, so that there is evidence that at least one of the omitted interactions should not be omitted. That is, the model with main effects only does not fit].

Next, we attempt to drop each of the two-way interactions from the model (while leaving the remaining ones in the model), leading to the following table.

term dropped	residual deviance	d.f.	SP for residual deviance	change in deviance compared to model (*)	d.f.	SP for change in deviance
gender:year	70.64	16	0.0000	51.54	4	0.0000
gender:response	153.9	15	0.0000	134.7	3	0.0000
year:response	216.3	24	0.0000	197.1	12	0.0000

Hence, there is strong evidence against dropping any of these interactions from the model, leading us to accept the partial association model.

In such circumstances, where there are partial associations between all the variables, it can sometimes be easier to describe what is going on by analysing the data at different levels of one of the factors. In this case, it may be appropriate to consider separate analyses of the relationship between YEAR and RESPONSE by GENDER i.e. for males and females separately.

For males only, fitting a model with just main effects, for YEAR and RESPONSE, gives a residual deviance of 203.0493 on 12 d.f., which is clearly significant ($p = 0.0000$), so that the interaction term cannot be omitted. Thus the appropriate model for the males includes the main effects of YEAR and RESPONSE, together with their interaction.

For the females only, fitting a model with just main effects, gives a residual deviance of 13.26291 on 12 d.f., $p = 0.3502$. The interaction term can safely be omitted, and as usual in contingency table analysis, it makes no good sense to omit the main effects.

Thus, for females only, the appropriate model includes just the main effects of YEAR and RESPONSE.

We can use these investigations to report how the patterns of response to this survey varied with gender and undergraduate year/graduate status.

For male students, because it was necessary to include an interaction term in the model, we must consider the different years separately. One useful way is to calculate the proportion of responses in each category (in the table in the question) by the corresponding row total, to give the following:

	Response			
Year	A	B	C	D
1	0.40	0.26	0.30	0.04
2	0.36	0.29	0.31	0.05
3	0.30	0.28	0.35	0.07
4	0.31	0.20	0.39	0.09
grad	0.15	0.23	0.44	0.18

• The clear pattern is that, for the four undergraduate years, the patterns of response are similar, with the majority of responses in categories A and C, a substantial number (about a quarter) in B, and only a small proportion in D.

i.e there is some tendency for the pattern of responses to move from 'hawkish' (A and B) to 'dovish' (C and D) with increasing undergraduate years.

• However, the pattern for graduate students is considerably different, with a much greater proportion of the responses going on the more 'dovish' policies C and D.

For female students, because there is no significant interaction between YEAR and RESPONSE, we can consider all the years together. The proportions of the total response in each category are as follows:

	Response			
	A	B	C	D
female	0.12	0.18	0.63	0.06

• Thus, overall, a clear majority of the female students chose response category C, and the pattern was similar over all years (including graduate students).

Generally the pattern for women students was much more 'dovish' than for male undergraduates, though not dissimilar from that of the male graduate students.

For completeness, the table of RESPONSE by YEAR for female students is calculated as:

	Response			
Year	A	B	C	D
1	0.17	0.25	0.52	0.06
2	0.07	0.13	0.48	0.04
3	0.13	0.17	0.66	0.04
4	0.12	0.21	0.57	0.10
grad	0.10	0.14	0.68	0.07

The appropriate R commands, with edited output, to complete the analysis in Example 1 follow:

```
> gender <- c(rep("male", 20), rep("female", 20))
> year <- rep(c(rep(1, 4), rep(2, 4), rep(3, 4), rep(4, 4), rep("grad", 4)), 2)
> response <- rep(c("A", "B", "C", "D"), 10)
> count <- c(175, 116, 131, 17, 160, 126, 135, 21, 132, 120, 154, 29, 145, 95,
             185, 44, 118, 176, 345, 141, 13, 19, 40, 5, 5, 9, 33, 3, 22, 29, 110,
             6, 12, 21, 58, 10, 19, 27, 128, 13)
> gender <- factor(gender)
> year <- factor(year)
> response <- factor(response)
> vietnam.df <- data.frame(gender, year, response, count)
> vietnam.df[1:10,  ]
   gender year response count
1    male    1        A   175
2    male    1        B   116
3    male    1        C   131
4    male    1        D    17
5    male    2        A   160
6    male    2        B   126
7    male    2        C   135
8    male    2        D    21
9    male    3        A   132
10   male    3        B   120

> mod1.glm <- glm(count ~ gender * year * response - gender:year:response,
                  family = poisson)

> dev <- mod1.glm$deviance
> df <- mod1.glm$df.residual

> dev
[1] 19.19
> df
[1] 12

> pchisq(dev, df, lower.tail=F)
[1] 0.08395

> mod2.glm <- glm(count ~ gender + year + response, family = poisson)
> dev <- mod2.glm$deviance
> df <- mod2.glm$df.residual
> dev
[1] 423.8
> df
[1] 31
```

```
> pchisq(dev, df, lower.tail=F)
[1] 1.591e-70

> pchisq(404.64, 19, lower.tail=F)
[1]  4.751e-74

> mod3.glm <- update(mod1.glm,  ~ . - gender:year)
> mod3.glm$deviance
[1] 70.64
> mod3.glm$df.residual
[1] 16

> mod4.glm <- update(mod1.glm,  ~ . - gender:response)
> mod4.glm$deviance
[1] 153.9
> mod4.glm$df.residual
[1] 15

> mod5.glm <- update(mod1.glm,  ~ . - year:response)
> mod5.glm$deviance
[1] 216.3
> mod5.glm$df.residual
[1] 24

> vietnam.males <- vietnam.df[1:20,  ]
> vietnam.males[1:5,  ]
  gender year response count
1   male    1        A   175
2   male    1        B   116
3   male    1        C   131
4   male    1        D    17
5   male    2        A   160

> vietnam.females <- vietnam.df[21:40,  ]
> vietnam.females[1:5,  ]
   gender year response count
21 female    1        A    13
22 female    1        B    19
23 female    1        C    40
24 female    1        D     5
25 female    2        A     5

> males.glm <- glm(count ~ response + year, family = poisson, data =
    vietnam.males)
> summary(males.glm)

Call:
glm(formula = count ~ response + year, family = poisson, data = vietnam.males)

Deviance Residuals:
   Min      1Q  Median      3Q     Max
 -7.67   -2.25    0.21    1.29    6.57

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  4.82783    0.05708   84.58  < 2e-16 ***
```

```
responseB   -0.14257    0.05431   -2.63   0.0087 **
responseC    0.26342    0.04922    5.35  8.7e-08 ***
responseD   -1.06362    0.07306  -14.56  < 2e-16 ***
year2        0.00681    0.06738    0.10   0.9195
year3       -0.00915    0.06765   -0.14   0.8924
year4        0.06610    0.06641    1.00   0.3195
yeargrad     0.57479    0.05967    9.63  < 2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 809.57  on 19  degrees of freedom
Residual deviance: 203.05  on 12  degrees of freedom
AIC: 348.7

Number of Fisher Scoring iterations: 4

> pchisq(203.05, 12, lower.tail=F)
[1] 7.65e-37

> females.glm <- glm(count ~ response + year, family = poisson, data =
    vietnam.females)
> summary(females.glm)

Call:
glm(formula = count ~ response + year, family = poisson, data = vietnam.females)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.5445  -0.5365   0.0202   0.4571   1.3047

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)    2.240      0.159   14.07  < 2e-16 ***
responseB      0.391      0.154    2.55   0.0109 *
responseC      1.648      0.130   12.72  < 2e-16 ***
responseD     -0.652      0.203   -3.21   0.0013 **
year2         -0.432      0.182   -2.38   0.0174 *
year3          0.774      0.138    5.62  1.9e-08 ***
year4          0.271      0.151    1.79   0.0729 .
yeargrad       0.887      0.135    6.55  5.6e-11 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 549.407  on 19  degrees of freedom
Residual deviance:  13.263  on 12  degrees of freedom
AIC: 124

Number of Fisher Scoring iterations: 4

> pchisq(13.263, 12, lower.tail=F)
[1] 0.3502
```

1 Appendix: R for GLMs

1.1 Introduction

The aim of this appendix and associated worked examples and exercises is to extend your abilities in statistical modelling in R to include the fitting of generalized linear models, and to give you experience of fitting such models.

You may start by reviewing the analyses presented so far in the lecture notes before tackling the exercises provided.

1.2 Some Useful Commands

In R, the necessary command for fitting a generalized linear model, is `glm`

```
glm(formula,family=gaussian(link=identity))
```

where '`formula`' defines the model to be fitted, and '`family`' defines the error distribution for a suitable choice from the *exponential family*, eg Gaussian (Normal), Binomial, Poisson and Gamma (which includes the exponential distribution - see Exercises 1). The '`link`' command specifies the *link function*. If `link` is omitted, R defaults to the appropriate *canonical* link for the given distribution, e.g. *logit* for Binomial, *log* for Poisson, etc. (Recall that for a logistic (Binomial) regression, we also need to specify `weights=totals` to give the number of trials upon which each observation was based).

For a response, `resp`, and explanatory variables, `expl1` and `expl2`, say, the two model formulae

`resp`$\sim$`expl1+expl2+expl1:expl2`, and
`resp`$\sim$`expl1*expl2`

are equivalent, defining a model with two explanatory variables and their interaction. A *constant* (or intercept) is automatically included, but can be omitted can by adding a `-1` term to the RHS of the model formula.

Once fitted, the `glm` object, `data.glm` say, contains information about the deviance and the parameter coefficients, etc. You can ask R to display this information, using

```
summary(data.glm)
```

The estimated parameters, $\beta_1, \ldots, \beta_p$, from the *linear predictor*, and the model *deviance* can be extracted using

`coef(data.glm)`, or `data.glm$coef`, and
`deviance(data.glm)`, or `data.glm$deviance`

In this way, `data.glm$coef[1]` stores the value of the intercept, and `data.glm$coef[2]`, `data.glm$coef[3]`, ..., give the parameter estimates of $\beta_2, \beta_3, \ldots$, in the order the explana-

tory variables are defined in the model formula.

The command

```
anova(data.glm)
```

gives the model deviances for the sequential fitting of the explanatory variables in the order they are given, and can be used to test the significance of subsequent terms in the model, in addition to the previously fitted terms. Where the *dispersion* parameter ϕ is known (fixed) so that a χ^2 goodness-of-fit test is appropriate, use

```
pchisq(deviance,df,lower.tail=F)
```

to obtain the necessary significance probability. Also,

```
anova(data.glm,test="Chisq")
```

will request the significance probabilities associated with adding each term to be displayed directly from R. For two (nested) fitted models

```
anova(mod1.glm,mod2.glm, test="Chisq")
```

will compare the model deviances directly.

Fitted values from a given model can be obtained directly via the command

```
fitted(data.glm)
```

(although it is useful to be able to construct these 'by hand').

1.3 Other functions (`update` and `predict`)

Two further R functions you may also like to investigate are `update` and `predict`.

When investigating suitable models, the function `update` allows a useful 'shorthand' means of altering the previously fitted model. e.g.

```
update(data.glm, ~.  -1)
```

removes the intercept from the model parameters to be fitted leaving all else unaltered.

```
update(data.glm, ~.  + expl3)
```

adds a third explanatory variable. Hence,

```
anova(data.glm,update(data.glm, ~.+expl3), test="Chisq")
```

tests whether the additional explanatory variable improves the fit of the current model, etc.

```
predict(data.glm, data.frame=c(expl1,expl2),type="response")
```

can be used to give predicted responses for particular levels of the explanatory variables. Further information about these commands can be found using the Help system.

1.4 A note about factors

The default parameterization for factor effects in R is the usual 'corner-point' parameterization, which compares every treatment level with the FIRST treatment level.

Recall that ('character' vectors) explanatory variables that are to be considered as factors should be declared to R using the function `factor( )`. However note also that, by default, the factor levels (or treatments) will be ordered alphabetically. In general you will want to set your own baseline level, and so will want that level to be the first in any explanatory factor. This can be done as follows.

Consider a factor indicating the height of the wife in a married couple. Height is to be categorized as 'S' (small), 'M' (medium) or 'T' (tall). We look at two different ways of setting up the factor. The first will give the baseline level as 'M' or medium. The second will give the baseline level as 'S' or small, which may be preferable.

```
> w <- rep(c("T", "M", "S"), 3)

> wife <- factor(w)
> wife
[1] T M S T M S T M S
Levels: M S T

> levels(wife)
[1] "M" "S" "T"

> wife <- factor(w, levels = c("S", "M", "T"))
> wife
[1] T M S T M S T M S
> levels(wife)
[1] "S" "M" "T"
```