Non-Life Insurance — Assignment 2

AJ*

Autumn 2021

Question 1

We run the following code:

```
> str(sex)
 Factor w/ 2 levels "1","2": 1 1 1 1 1 1 1 1 1 1 ...

> str(rep(1:2, each=27, len=54))
 int [1:54] 1 1 1 1 1 1 1 1 1 1 ...

> 2*sex
 [1] NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA ...
[29] NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA NA
Warning message:
In Ops.factor(2, sex) : ‘*’ not meaningful for factors

> 2*rep(1:2, each=27, len=54)
 [1] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 4 4 ...
[43] 4 4 4 4 4 4 4 4 4 4 4 4
```

`sex` is correctly identified as a factor variable, whereas `(rep(1:2, each=27, len=54)` is an integer vector with entries 1 or 2. Therefore, multiplying a factor variable by 2 produces an error, but is perfectly valid in the second case.

*Student number: ∞

Question 2

We have:

```
> set.seed(50); subset <- sort(sample(1:54,10))
> data.frame(sex, region, type, job, n, expo)[subset,]

   sex region type job  n expo
3    1      1    1   3 10  210
8    1      1    3   2 12  175
11   1      2    1   2  5  133
16   1      2    3   1 13  112
18   1      2    3   3 24  203
31   2      1    2   1 18  203
46   2      3    1   1 16  175
48   2      3    1   3 14  203
51   2      3    2   3 20  112
52   2      3    3   1  9   77
```

We are asked to verify the values of the covariates for the first two cells by hand.

Consider the first 10 elements of the vectors we used to construct the levels of the covariates:

```
> sex
[1] 1 1 1 1 1 1 1 1 1 1
> region
[1] 1 1 1 1 1 1 1 1 1 2
> type
[1] 1 1 1 2 2 2 3 3 3 1
> job
[1] 1 2 3 1 2 3 1 2 3 1
```

Comparing the 3^{rd} and 8^{th} entries of the vectors with the table, the output is what we would expect.

Question 3

We modify the model with `type` before `region`:

```
# second model
anova(glm(n/expo ~ type*region, family = poisson, weights = expo))
```

which has output:

```
            Df Deviance Resid. Df Resid. Dev
NULL                           53    104.732
type         2   36.367        51     68.365
region       2   23.424        49     44.940
type:region  4    2.529        45     42.412
```

Comparing the change in deviance from the four models

`1 + region` $\rightarrow$ `1 + region + type`,

and

`1 + type` $\rightarrow$ `1 + type + region`,

we see that the order in which the terms are added *does* impact the change in deviance due to the inclusion of the same covariate, however, once both covariates are in the model the residual deviance is the same.

Question 4

We use:

```
g.wei <- glm(n/expo ~ region*type, family = poisson, weights = expo)
g.off <- glm(n ~ region*type + offset(log(expo)), family = poisson)
g.wei; g.off
```

We notice that the output is *identical* with the exception of the **AIC** term, which can only be computed under the second model (`g.off`), owing to the fact that `n/expo` is often not an integer, and the Poisson likelihood can therefore not be used.

Question 5

Multiplying the indicator variables for `region2` and `type3` will only be non-zero when both variables are present, which is exactly what the interaction term `region2:type3` is trying to capture.

Question 6

We will use:

```
(Intercept)     region2     region3       type2       type3
 -3.0313238   0.2314097   0.4604585   0.3941889   0.5833108
```

(a) This corresponds to the reference class, represented by the intercept term, and computed by:

```
exp(coef(g.main)[1])
```

and yields `0.04825172` expected claims per year, which is equivalent to saying `1 / 0.04825172` = 1 claim every **20.72** years on average.

(b) This corresponds to `region3`, `type3`, and is computed by:

```
most.risky <- exp(cc[1] + cc[3] + cc[5])
```

and yields `0.1370301` expected claims per year, or `1 / 0.1370301` = 1 claim every **7.3** years on average.

(c) The safest class is the reference class, so the estimate is the same as (a).

Question 7

Combining the ingredients $\mathbf{X}$, β, offset term $\mathbf{o}$, and the inverse of the link function $g(\cdot)$ in `R` as $g(\mathbf{X}\beta + \mathbf{o})$, we have:

```
# computation
a <- (model.matrix(g.off) %*% coef(g.off) + g.off$offset)
b <- as.numeric(g.off$family$linkinv(a))

# comparison
all.equal(fitted(g.off), b, check.attributes = FALSE)
# [1] TRUE
```

Question 8

(a) Comparing the equations carefully, we see that `g.altr` is the special case where *region* forms a geometric progression. That is to say, we can write `g.altr` as:

$$\log(\mu_i) = \beta_0 + \beta_1(r_i - 1) + \beta_2 t_{i2} + \beta_3 t_{i3}$$

and now we no longer need to estimate separate parameters β_1 and β_2 for the covariates r_{i2} and r_{i3} as in `g.main`.

(b) We run:

```
> anova(g.altr, g.main)
```

which has output:

```
Analysis of Deviance Table

Model 1: n/expo ~ as.numeric(region) + type
Model 2: n/expo ~ region + type
  Resid. Df Resid. Dev Df  Deviance
1        50     44.941
2        49     44.940  1 0.0002148
```

The change in residual deviance is not significant for a χ^2_1 random variable, so `g.altr` is the preferred model.

(c) The restriction does make (some) logical sense if we believe that region 1 (the countryside) is the most safe place to drive, and region 3 (big city) is the most dangerous, with other regions being an intermediate level. We would expect more claims to arise in densely populated regions where people come into contact, so this is certainly plausible.

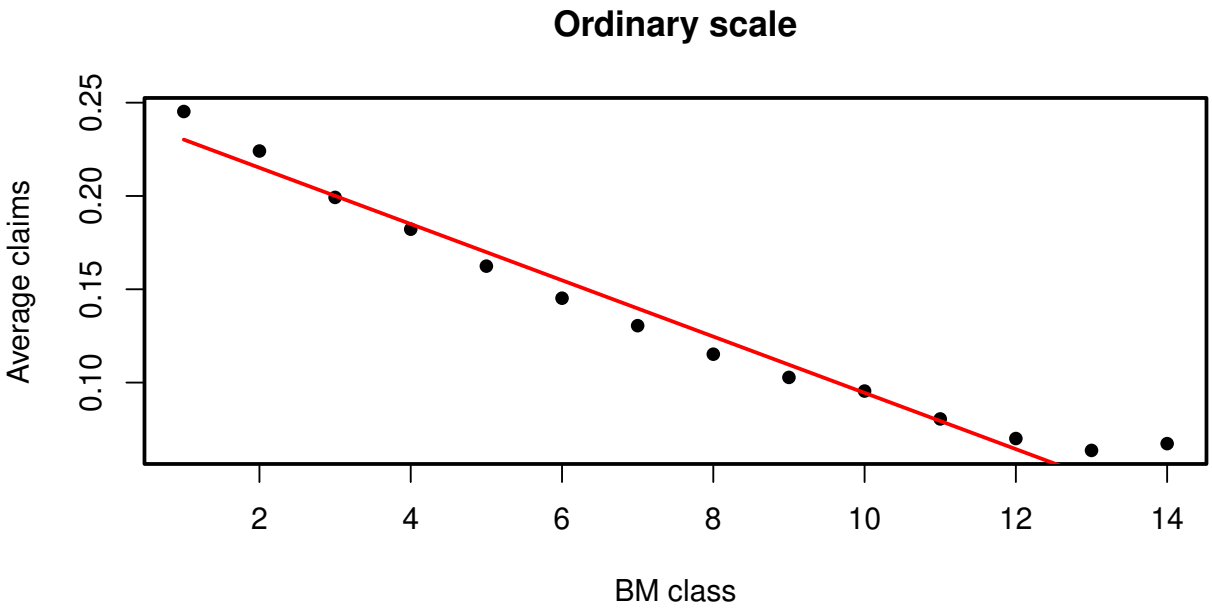
Question 9

Filling in the dots, we have:

```
# total claim numbers in each BM class
t1 <- tapply(nCl, B, sum)

# total exposure in each BM class
t2 <- tapply(Expo, B, sum)

# plot average number of claims t1/t2
par(mfrow=c(1,1),lwd=2) ## to get two plots next to each other, thick lines
plot(1:14, t1/t2, main="Ordinary scale", pch=16, xlab="BM class",
     ylab="Average claims")
lines(1:13, fitted(lm((t1/t2)[1:13]~I(1:13))), col="red")
```



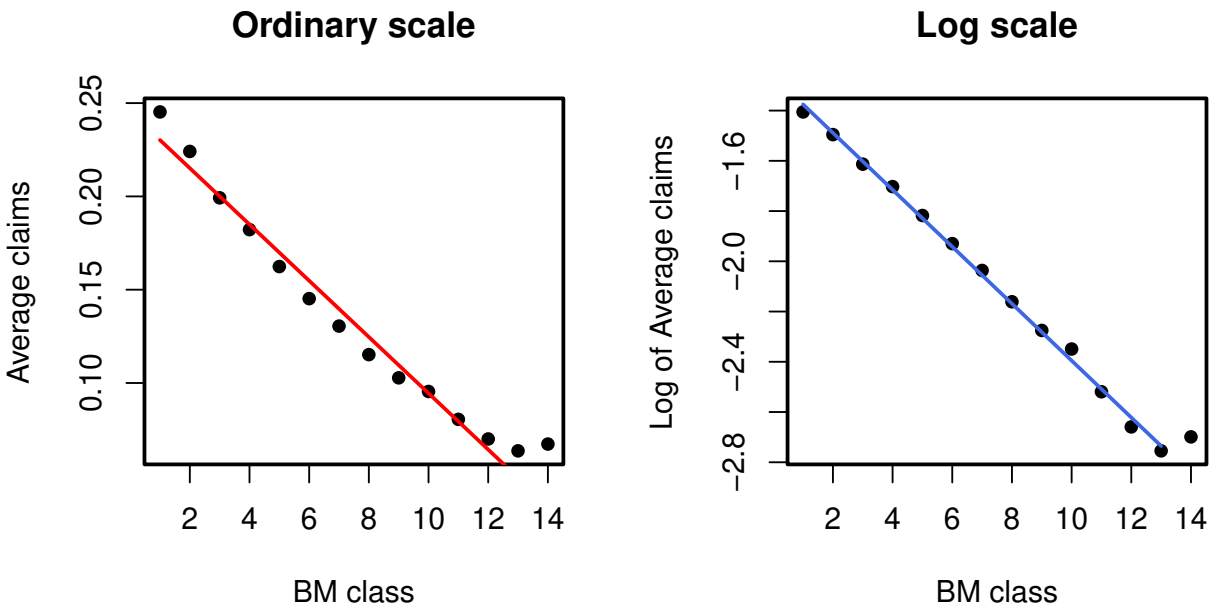
From the plot, it would appear that the average claim figures *do not* follow a linear trend.

Question 10

To remedy matters, we consider the situation in which there is a linear trend in the *logarithm* of the average claim figures:

```
# original scale, side-by-side
par(mfrow=c(1,2),lwd=2) ## to get two plots next to each other, thick lines
plot(1:14, t1/t2, main="Ordinary scale", pch=16, xlab="BM class",
     ylab="Average claims")
lines(1:13, fitted(lm((t1/t2)[1:13]~I(1:13))), col="red")

# generate plot of log(t1/t2)
plot(1:14, log(t1/t2), main="Log scale", pch=16, xlab="BM class",
     ylab="Log of Average claims")
lines(1:13, fitted(lm(log(t1/t2)[1:13]~I(1:13))), col="royalblue")
```



Comparing the plots, it does seem like the average claim numbers for the first 13 bonus-malus classes follow an exponential trend *i.e.* there is a linear trend in the *logarithm*.

Question 11

Loss ratio for each risk cell displayed by:

```
l <- list(Use=U, Age=A, Area=R, Mile=M)
risk.cells <- ftable(round(100*tapply(TotCl,l,sum)/tapply(TotPrem,l,sum)),
       row.vars=2, col.vars=c(1,3,4))
```

Table 1

```
> risk.cells
    Use    1                                   2
    Area   1           2           3           1           2           3
    Mile   1   2   3   1   2   3   1   2   3   1   2   3   1   2   3   1   2   3
Age
1        111  99  90 108 114 114 114 110  98 143 148 114 209 177 112 155 160 139
2         48  44  41  48  43  40  50  44  40  71  60  55  72  61  52  69  62  56
3         71  65  67  79  75  57  70  64  58  95  86  85 104  93  89  94  82  81
```

To identify *good risks* by the criterion stated in the question, run:

```
df.riskcells <- data.frame(risk.cells)
names(df.riskcells)[5] <- "Loss Ratio"
good.customers <- which(df.riskcells[5] <= 56)
groups <- df.riskcells[good.customers,]
```

Table 2

```
> groups
   Age Use Area Mile Loss Ratio
2    2   1    1    1         48
8    2   1    2    1         48
14   2   1    3    1         50
20   2   1    1    2         44
26   2   1    2    2         43
32   2   1    3    2         44
38   2   1    1    3         41
41   2   2    1    3         55
44   2   1    2    3         40
47   2   2    2    3         52
50   2   1    3    3         40
53   2   2    3    3         56
```

The `Age` column of **Table 2** makes clear the trend which can also be seen by looking across the second row of **Table 1**: drivers in age group 2 are the only age group that constitute *good risks*, regardless of the other factors (with some exceptions). Clearly, we would want to target this group of policyholders with our marketing.

Question 12

We need to consider how many cells have `Expo == 0`.

In model `g1`:

```
length(g1$coefficients) + g1$df.residual[1]  = 7524
```

Bonus-malus classes *(11, 12, 13, 14)* must necessarily be empty for age class 1, so let's consider how many cells correspond to this combination of factors.

We have:
4 empty BM classes, **3** Regions, **2** Usages, **11** Weight Classes and **3** Mileage classes.
Considering their product:

$$4 \times 3 \times 2 \times 11 \times 3 = 792.$$

Now, $7524 + 792 = 8316$, and that explains why ignoring the cells with `Expo == 0` leads to the degrees of freedom that we see in the the **ANOVA** output.

Question 13

(a) The following is a brute-force method for calculating the fitted value for cell 7785 under the three models g1, g2, g3:

```
# extract covariates in cell 7785
cov.7785 <- Cars[7785,][5:10]

# remove reference class covariates, add intercept term
cov.7785 <- c(1, as.numeric(cov.7785[cov.7785 != 1]))

# > cov.7785
[1]  1 14  2  2  2
[of the form (Int, B, WW, A, M)]

# model coefficients g1, no beta for factor M
g1beta <- exp(coef(g1))
relevant.beta1 <- c(g1beta[1], g1beta[9], g1beta[7],
                    g1beta[4], 0)

# model coefficients g2
g2beta <- exp(coef(g2))
relevant.beta2 <- c(g2beta[1], g2beta[9], g2beta[7],
                    g2beta[4], g2beta[10])

# model coefficients g3, drop mileage M
g3beta <- exp(coef(g3))
relevant.beta3 <- c(g3beta[1], g3beta[20], g3beta[7],
                     g3beta[4], 0)

# my fitted values
my.g1.fit <- t(cov.7785) %*% relevant.beta1
my.g2.fit <- t(cov.7785) %*% relevant.beta2
my.g3.fit <- t(cov.7785) %*% relevant.beta3

# summarise results
my.fit <- c(my.g1.fit, my.g2.fit, my.g3.fit)
names(my.fit) <- c("Model 1", "Model 2", "Model 3")

 > my.fit
 Model 1 Model 2 Model 3
 545.4   545.7   525.0
```

(b) Now, let's run:

```
r.fitted.vals <- list(fitted(g1), fitted(g2), fitted(g3))
length.r.fitted <- as.numeric(lapply(r.fitted.vals, length))

> length.r.fitted
[1] 7524 7524 7524
```

We again see that `R` has been assisting us in fitting the models by ignoring the cells that have `Expo == 0`. We understand why these cells are necessarily empty have done **Question 12**, but here the implication is that R does not fit values for the same set of cells, which is why the the length of the vector of fitted values is $8316 - 792 = 7524$.

Question 14

(a) *Please note*: in this question, all χ^2_k critical values are taken at the **95^{th} percentile**.

```
# (re)call
g1 <- glm(TotCl/Expo~R+A+U+W+Bminus1+Bis14, quasipoisson, wei=Expo)

# remove Bis14
g1.hat <- glm(TotCl/Expo~R+A+U+W+Bminus1, quasipoisson, wei=Expo)

> anova(g1.hat, g1)
Analysis of Deviance Table

Model 1: TotCl/Expo ~ R + A + U + W + Bminus1
Model 2: TotCl/Expo ~ R + A + U + W + Bminus1 + Bis14
  Resid. Df Resid. Dev Df Deviance
1      7516   38755743
2      7515   38616941  1   138802
```

Using: $\hat{\phi} = 5137$, and the fact that the decrease in deviance is 138802, we have that the decrease in *scaled* deviance is:

$$\frac{138802}{5137} = 27.02$$

Comparing this with the critical value of $\chi^2_1 = 3.841$, we see that `Bis14` *cannot* be removed from the model without getting a significantly worse fit.

(b)

```
# (re)call
g3 <- glm(TotCl/Expo~R+A+U+W+B, quasipoisson, wei=Expo)
g3.removeB <- glm(TotCl/Expo~R+A+U+W, quasipoisson, wei=Expo)
g3.removeW <- glm(TotCl/Expo~R+A+U+B, quasipoisson, wei=Expo)

# anova
anova(g3.removeB, g3)
anova(g3.removeW, g3)
```

```
> anova(g3.removeB, g3)
Analysis of Deviance Table

Model 1: TotCl/Expo ~ R + A + U + W
Model 2: TotCl/Expo ~ R + A + U + W + B
  Resid. Df Resid. Dev Df Deviance
1      7517   78902891
2      7504   38544506 13 40358385
```

```
> anova(g3.removeW, g3)
Analysis of Deviance Table

Model 1: TotCl/Expo ~ R + A + U + B
Model 2: TotCl/Expo ~ R + A + U + W + B
  Resid. Df Resid. Dev Df Deviance
1      7505   45495122
2      7504   38544506  1  6950616
```

We still use $\hat{\phi} = 5137$.

Starting with `B`, the decrease in deviance is 40358385. Therefore, the decrease in *scaled* deviance is:

$$\frac{40358385}{5137} = 7857$$

Comparing this with the critical value of $\chi^2_{13} = 22.36$, we see that `B` also *cannot* be removed from the model without getting a significantly worse fit.

Similarly for `W`, the decrease in deviance is 6950616. Therefore, the decrease in *scaled* deviance is:

$$\frac{6950616}{5137} = 1353$$

Comparing this with the critical value of $\chi^2_1 = 3.841$, we see that `W` also *cannot* be removed from the model without getting a significantly worse fit.

(c)

```
# (re)call
g1 <- glm(TotCl/Expo~R+A+U+W+Bminus1+Bis14, quasipoisson, wei=Expo)
g1.augmented <- glm(TotCl/Expo~R+A+U+WW+Bminus1+Bis14, quasipoisson, wei=Expo)

> anova(g1, g1.augmented)
Analysis of Deviance Table

Model 1: TotCl/Expo ~ R + A + U + W + Bminus1 + Bis14
Model 2: TotCl/Expo ~ R + A + U + WW + Bminus1 + Bis14
  Resid. Df Resid. Dev Df Deviance
1      7515   38616941
2      7506   38593888  9    23053
```

The decrease in deviance is 23053. Therefore, the decrease in *scaled* deviance is:

$$\frac{23053}{5137} = 4.488$$

Comparing this with the critical value of $\chi^2_9 = 16.92$, we see that it *would not help* to allow for separate coefficients for each weight class.

Question 15

(a) We run:

```
h1 <- glm(nCl/Expo~R+A+U+W+Bminus1+Bis14, family = poisson, weights = Expo)

> h1

Call:  glm(formula = nCl/Expo ~ R + A + U + W + Bminus1 + Bis14,
    family = poisson, weights = Expo)

Coefficients:
(Intercept)           R2           R3           A2           A3           U2
     -1.569        0.184        0.404       -0.771       -0.239        0.363
          W      Bminus1        Bis14
      0.878       -0.107        0.198

Degrees of Freedom: 7523 Total (i.e. Null);  7515 Residual
  (792 observations deleted due to missingness)
Null Deviance:     50200
Residual Deviance: 8240  AIC: Inf
```

(b)

```
h2 <- glm(TotCl/nCl~R+A+U+W+Bminus1+Bis14, family = Gamma(link = "log"),
          weights = nCl)

> h2

Call:  glm(formula = TotCl/nCl ~ R + A + U + W + Bminus1 + Bis14,
    family = Gamma(link = "log"), weights = nCl)

Coefficients:
(Intercept)           R2           R3           A2           A3           U2
    7.83242     -0.10150     -0.22954     -0.11239     -0.24754     -0.03967
          W      Bminus1        Bis14
   -0.00960     -0.00024     -0.09777

Degrees of Freedom: 6984 Total (i.e. Null);  6976 Residual
  (1331 observations deleted due to missingness)
Null Deviance:     13500
Residual Deviance: 11800  AIC: 1600000
```

(c) Assuming independence between claim numbers and sizes, we combine the multiplicative coefficients using:

```
h3.coef <- exp(h1$coefficients) * exp(h2$coefficients)
```

(d) Comparing the results from (c) with our earlier quasi-Poisson model for `TotCl/Expo` using the same covariates (`g1`), we have:

```
> h3.coef
(Intercept)          R2          R3          A2          A3          U2           W
   525.1108      1.0857      1.1901      0.4135      0.6145      1.3823      2.3827
    Bminus1       Bis14
     0.8979      1.1057

> exp(g1$coefficients)
(Intercept)          R2          R3          A2          A3          U2           W
   524.3017      1.0843      1.1916      0.4147      0.6184      1.3841      2.3722
    Bminus1       Bis14
     0.8979      1.1054
```

This is an intriguing result: by considering separate models for mean claim frequency and average claim sizes and then combining them, we have created a model for the risk premium in each cell (`TotCl/Expo`) which closely resembles the quasi-Poisson model we computed directly.