HW4 Suggested Solutions

1.

Our data is about military coups in sub-Saharan African countries, it consists ofthe following variables:

miltcoup: number of military coups from independence to 1989

oligarchy: number years country ruled by military oligarchy from independence to 1989

pollib: degree of political liberalisation, this variable is ordinal

parties: number of legal political parties since 1993

pctvote: voter turnout in the last election

popn: population in millions in 1989

size: area in 1000 km square

numelec: total number of legislative and presidential elections

numregim: number of regime types

There are some missing data in this data set. We use the simplest approach to impute the data, which is using the median of that variable to impute.

We write the response (number of military coups from independence to 1989) as Y and the other variables as $\boldsymbol{X}$. We make the following modeling assumption:

$$Y|\boldsymbol{X} = \boldsymbol{x} \sim \text{Poisson}(\mu_{\boldsymbol{x}})$$

(so $\mu_{\boldsymbol{x}} = E(Y|\boldsymbol{X} = \boldsymbol{x})$)

$$\log(\mu_{\boldsymbol{x}}) = \beta_0 + \beta_1 oligarchy + \beta_2 pollib + \beta_3 parties + \beta_4 pctvote$$
$$+\beta_5 popn + \beta_6 size + \beta_7 numelec + \beta_8 numregim$$

```
##
## Call:
## glm(formula = miltcoup ~ ., family = poisson, data = africa_imputed)
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -9.468e-01  7.846e-01  -1.207 0.227540
## oligarchy    1.014e-01  2.921e-02   3.472 0.000517 ***
## pollib      -5.164e-01  2.351e-01  -2.196 0.028073 *
## parties      2.045e-02  1.026e-02   1.993 0.046316 *
## pctvote      8.622e-03  9.104e-03   0.947 0.343620
## popn         5.424e-03  5.820e-03   0.932 0.351297
## size        -5.818e-05  2.093e-04  -0.278 0.781082
## numelec      3.929e-02  5.293e-02   0.742 0.457995
```

```
## numregim     1.712e-01  1.951e-01   0.877 0.380328
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 93.971  on 46  degrees of freedom
## Residual deviance: 43.644  on 38  degrees of freedom
## AIC: 135.32
##
## Number of Fisher Scoring iterations: 6
```

From the model summary, we can see that this model fits quite well because the residual deviance is 43.644 on 38 degrees of freedom. If we perform the following hypothesis test:

$$H_0 : \text{The model fits}, H_1 : \text{The model does not fit},$$

then the p-value is $P(\chi^2_{38} > 43.644) = 0.2439842 >> 0.05$, indicating that we cannot reject H_0, so the model fits the data.

However, this model is not simple, there are many non-significant predictors. So we use the AIC to decide which predictor we should remove. The AIC of a model is defined as:

$$AIC = 2k - 2\log(\hat{L})$$

(k is the number of parameters estimated, $\hat{L}$ is the maximum likelihood of the model (i.e. the likelihood function evaluated at the maximum likelihood estimator))

```
## Single term deletions
##
## Model:
## miltcoup ~ oligarchy + pollib + parties + pctvote + popn + size +
##     numelec + numregim
##           Df Deviance    AIC
## <none>         43.644 135.32
## oligarchy  1   55.779 145.46
## pollib     1   48.424 138.10
## parties    1   47.245 136.93
## pctvote    1   44.531 134.21
## popn       1   44.495 134.18
## size       1   43.723 133.40
## numelec    1   44.200 133.88
## numregim   1   44.411 134.09
```

We can see that based on the AIC, removing size should improve the model the most, so we fit our second model:

$$\log(\mu_{\boldsymbol{x}}) = \beta_0 + \beta_1 oligarchy + \beta_2 pollib + \beta_3 parties + \beta_4 pctvote$$
$$+\beta_5 popn + \beta_6 numelec + \beta_7 numregim$$

```
##
## Call:
```

```
## glm(formula = miltcoup ~ oligarchy + pollib + parties + pctvote +
##     popn + numelec + numregim, family = poisson, data = africa_imputed)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.969771   0.780534  -1.242 0.214072
## oligarchy    0.100346   0.028936   3.468 0.000525 ***
## pollib      -0.511859   0.234079  -2.187 0.028765 *
## parties      0.020710   0.010234   2.024 0.043011 *
## pctvote      0.008597   0.009127   0.942 0.346237
## popn         0.005207   0.005782   0.901 0.367794
## numelec      0.036944   0.052364   0.706 0.480478
## numregim     0.172583   0.195377   0.883 0.377055
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 93.971  on 46  degrees of freedom
## Residual deviance: 43.723  on 39  degrees of freedom
## AIC: 133.4
##
## Number of Fisher Scoring iterations: 6
```

Again, this model fits well, the p-value is $P(\chi^2_{39} > 43.723) = 0.2777831 >> 0.05$. However, it is still not simple enough. Again, using the AIC, we find a predictor to remove:

```
## Single term deletions
##
## Model:
## miltcoup ~ oligarchy + pollib + parties + pctvote + popn + numelec +
##     numregim
##           Df Deviance    AIC
## <none>         43.723 133.40
## oligarchy  1   55.794 143.47
## pollib     1   48.456 136.14
## parties    1   47.424 135.10
## pctvote    1   44.600 132.28
## popn       1   44.518 132.20
## numelec    1   44.225 131.91
## numregim   1   44.500 132.18
```

We can see that removing numelec will decrease AIC the most. We now fit our third model:

$$\log(\mu_{\boldsymbol{x}}) = \beta_0 + \beta_1 oligarchy + \beta_2 pollib + \beta_3 parties + \beta_4 pctvote$$
$$+\beta_5 popn + \beta_6 numregim$$

```
##
## Call:
## glm(formula = miltcoup ~ oligarchy + pollib + parties + pctvote +
##     popn + numregim, family = poisson, data = africa_imputed)
##
```

```
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.734150   0.687959  -1.067 0.285908
## oligarchy    0.090300   0.025175   3.587 0.000335 ***
## pollib      -0.592600   0.209090  -2.834 0.004594 **
## parties      0.022831   0.009680   2.359 0.018344 *
## pctvote      0.009849   0.008919   1.104 0.269495
## popn         0.006065   0.005702   1.064 0.287498
## numregim     0.213985   0.185424   1.154 0.248486
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 93.971  on 46  degrees of freedom
## Residual deviance: 44.225  on 40  degrees of freedom
## AIC: 131.91
##
## Number of Fisher Scoring iterations: 6
```

Again, this model fits well, the p-value is $P(\chi^2_{40} > 44.225) = 0.2978188 >> 0.05$. However, it is still not simple enough. Again, using the AIC, we find a predictor to remove:

```
## Single term deletions
##
## Model:
## miltcoup ~ oligarchy + pollib + parties + pctvote + popn + numregim
##           Df Deviance    AIC
## <none>         44.225 131.91
## oligarchy  1   57.163 142.84
## pollib     1   51.557 137.24
## parties    1   49.234 134.91
## pctvote    1   45.434 131.11
## popn       1   45.325 131.00
## numregim   1   45.555 131.24
```

We can see that removing popn will decrease AIC the most. We now fit our next model:

$$\log(\mu_{\boldsymbol{x}}) = \beta_0 + \beta_1 oligarchy + \beta_2 pollib + \beta_3 parties + \beta_4 pctvote + \beta_5 numregim$$

```
##
## Call:
## glm(formula = miltcoup ~ oligarchy + pollib + parties + pctvote +
##     numregim, family = poisson, data = africa_imputed)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.369111   0.579713  -0.637  0.52431
## oligarchy    0.104818   0.021259   4.930  8.2e-07 ***
## pollib      -0.627628   0.204009  -3.076  0.00209 **
## parties      0.019628   0.009211   2.131  0.03310 *
## pctvote      0.009433   0.008869   1.064  0.28749
## numregim     0.124329   0.164880   0.754  0.45082
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 93.971  on 46  degrees of freedom
## Residual deviance: 45.325  on 41  degrees of freedom
## AIC: 131
##
## Number of Fisher Scoring iterations: 5
```

Again, this model fits well, the p-value is $P(\chi^2_{41} > 45.325) = 0.2963708 >> 0.05$. However, we can see that the p-value for numregim is still quite high, so we check if dropping it will decrease the AIC:

```
## Single term deletions
##
## Model:
## miltcoup ~ oligarchy + pollib + parties + pctvote + numregim
##           Df Deviance    AIC
## <none>         45.325 131.00
## oligarchy  1   68.479 152.16
## pollib     1   53.996 137.68
## parties    1   49.402 133.08
## pctvote    1   46.446 130.13
## numregim   1   45.893 129.57
```

Dropping numregim indeed will decrease the AIC the most, now we fit the model:

$$\log(\mu_{\boldsymbol{x}}) = \beta_0 + \beta_1 oligarchy + \beta_2 pollib + \beta_3 parties + \beta_4 pctvote$$

```
##
## Call:
## glm(formula = miltcoup ~ oligarchy + pollib + parties + pctvote,
##     family = poisson, data = africa_imputed)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.088645   0.440563  -0.201  0.84054
## oligarchy    0.108935   0.020290   5.369 7.92e-08 ***
## pollib      -0.644692   0.204523  -3.152  0.00162 **
## parties      0.020841   0.008960   2.326  0.02001 *
## pctvote      0.010909   0.008745   1.247  0.21225
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 93.971  on 46  degrees of freedom
## Residual deviance: 45.893  on 42  degrees of freedom
## AIC: 129.57
##
## Number of Fisher Scoring iterations: 5
```

Again, this model fits well, the p-value is $P(\chi^2_{42} > 45.893) = 0.3139866 >> 0.05$. There is still one non-significant predictor. We check if removing it would decrease the AIC:

```
## Single term deletions
##
## Model:
## miltcoup ~ oligarchy + pollib + parties + pctvote
##           Df Deviance    AIC
## <none>         45.893 129.57
## oligarchy  1   73.891 155.57
## pollib     1   54.887 136.57
## parties    1   50.775 132.46
## pctvote    1   47.414 129.09
```

And indeed it will. So we remove pctvote from our model.

Now we fit what should be our final model:

$$\log(\mu_{\boldsymbol{x}}) = \beta_0 + \beta_1 oligarchy + \beta_2 pollib + \beta_3 parties$$

```
##
## Call:
## glm(formula = miltcoup ~ oligarchy + pollib + parties, family = poisson,
##     data = africa_imputed)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.203243   0.365896   0.555  0.57858
## oligarchy    0.107423   0.019945   5.386 7.21e-08 ***
## pollib      -0.570511   0.194837  -2.928  0.00341 **
## parties      0.018210   0.008564   2.126  0.03348 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 93.971  on 46  degrees of freedom
## Residual deviance: 47.414  on 43  degrees of freedom
## AIC: 129.09
##
## Number of Fisher Scoring iterations: 5
```

So our final model is:

$$\log(\mu_{\boldsymbol{x}}) = \beta_0 + \beta_1 oligarchy + \beta_2 pollib + \beta_3 parties$$

We have treated political liberalisation as a quantitative variable even though it is actually ordinal. So here we fit a model with possible quadratic effect:

```
##
## Call:
## glm(formula = miltcoup ~ oligarchy + pollib + parties, family = poisson,
##     data = africa_imputed)
```

```
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.370636   0.268754  -1.379   0.1679
## oligarchy    0.106630   0.021008   5.076 3.86e-07 ***
## pollib.L    -0.789257   0.316266  -2.496   0.0126 *
## pollib.Q    -0.030794   0.261792  -0.118   0.9064
## parties      0.018410   0.008726   2.110   0.0349 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 93.971  on 46  degrees of freedom
## Residual deviance: 47.401  on 42  degrees of freedom
## AIC: 131.08
##
## Number of Fisher Scoring iterations: 5
```

We see that the quadratic effect of political liberalisation is not significant (p-value is 0.9064), therefore, our simple model of:

$$\log(\hat{\mu_{\boldsymbol{x}}}) = \beta_0 + \beta_1 oligarchy + \beta_2 pollib + \beta_3 parties$$
$$= 0.203243 + 0.107423 oligarchy - 0.570511 pollib + 0.018210 parties$$

is adequate.

Its deviance is 47.414 on 43 degrees of freedom, which indicates a relatively good fit. We perform the hypothesis test:

$$H_0 : \text{The model fits}, H_1 : \text{The model does not fit},$$

the p-value is $P(\chi^2_{43} > 47.414) = 0.2973727 >> 0.05$, indicating that we cannot reject H_0, so the model fits the data.

Furthermore, each predictor is significant according to the Wald test.

According to our model:

If other predictors are held constant, an additional year of being ruled by a military oligarchy (from independence to 1989) is associated with a $exp(0.107423) = 1.113405$ (about 11.3%) increase in the mean number of military coups.

If other predictors are held constant, an additional point on the political liberalisation rating is associated with a $exp(-0.570511) = 0.5652365$ increase (i.e. about 43.5% decrease) in the mean number of military coups.

If other predictors are held constant, one more legal political parties in 1993 is associated with a $exp(0.018210) = 1.018377$ (about 1.8%) increase in the mean number of military coups.

2.

We write the response (numbers of revertant colonies) as Y and the predictor variable (dose level of quinoline) as X.

$$Y|X = x \sim \text{Poisson}(\mu_x)$$

(so $\mu_x = E(Y|X = x)$)

Since there is only one predictor, we start by fitting the most simple model:

$$\log(\mu_{\boldsymbol{x}}) = \beta_0 + \beta_1 x$$

```
##
## Call:
## glm(formula = colonies ~ dose, family = poisson, data = salmonella)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) 3.3219950  0.0540292  61.485   <2e-16 ***
## dose        0.0001901  0.0001172   1.622    0.105
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 78.358  on 17  degrees of freedom
## Residual deviance: 75.806  on 16  degrees of freedom
## AIC: 172.34
##
## Number of Fisher Scoring iterations: 4
```

Our only predictor, dose, is not significant at the 0.05 level. The deviance of the model is very high, 75.806 on 16 degrees of freedom. We can test if the model is adequate:

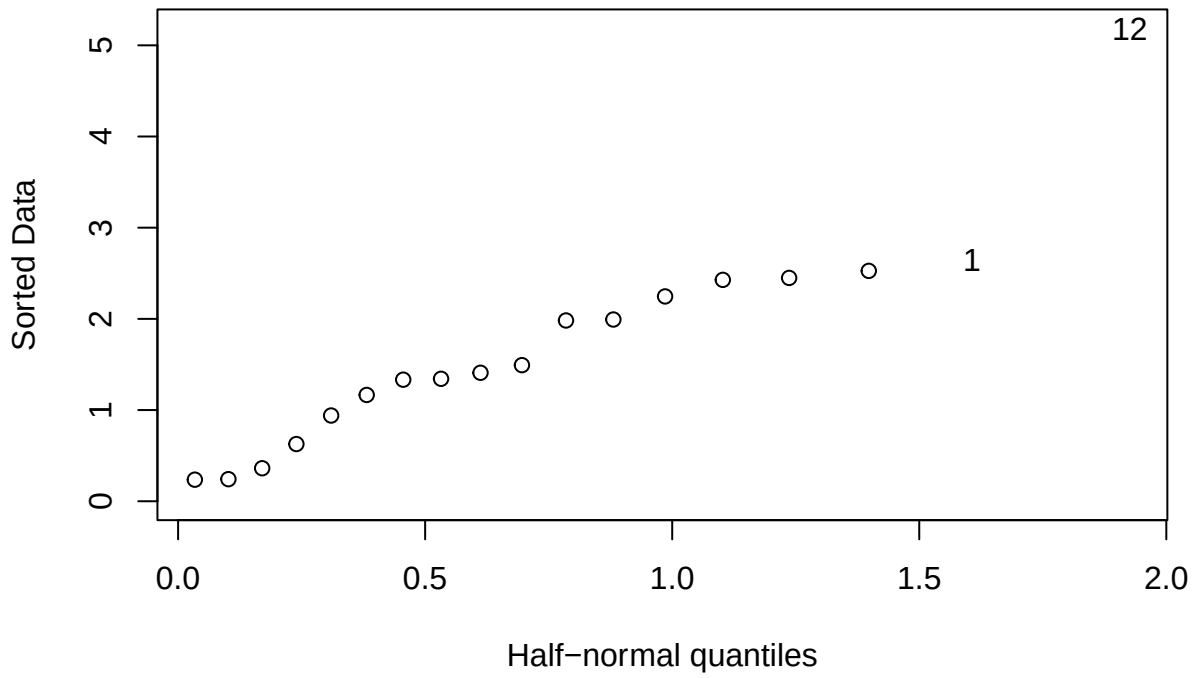
$$H_0 : \text{Model fits the data}, \quad H_1 : \text{Model does not fit the data}$$

The p-value is for this hypothesis test is virtually 0 ($P(\chi^2_{16} > 75.806) \approx 0$), indicating a lack of fit, so we reject H_0.

We could modify the model to allow a dispersion parameter that could vary:

$$\text{var}(Y|X = x) = \sigma^2 \mu_x,$$

but we check other possible causes of the high deviance first.



We can see that observation 12 is a potential outlier, so we remove it from the data and refit the model:

```
##
## Call:
## glm(formula = colonies ~ dose, family = poisson, data = salmonella_no_outlier)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) 3.2317358  0.0582053  55.523   <2e-16 ***
## dose        0.0002739  0.0001197   2.289   0.0221 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 51.379  on 16  degrees of freedom
## Residual deviance: 46.347  on 15  degrees of freedom
## AIC: 136.95
##
## Number of Fisher Scoring iterations: 4
```

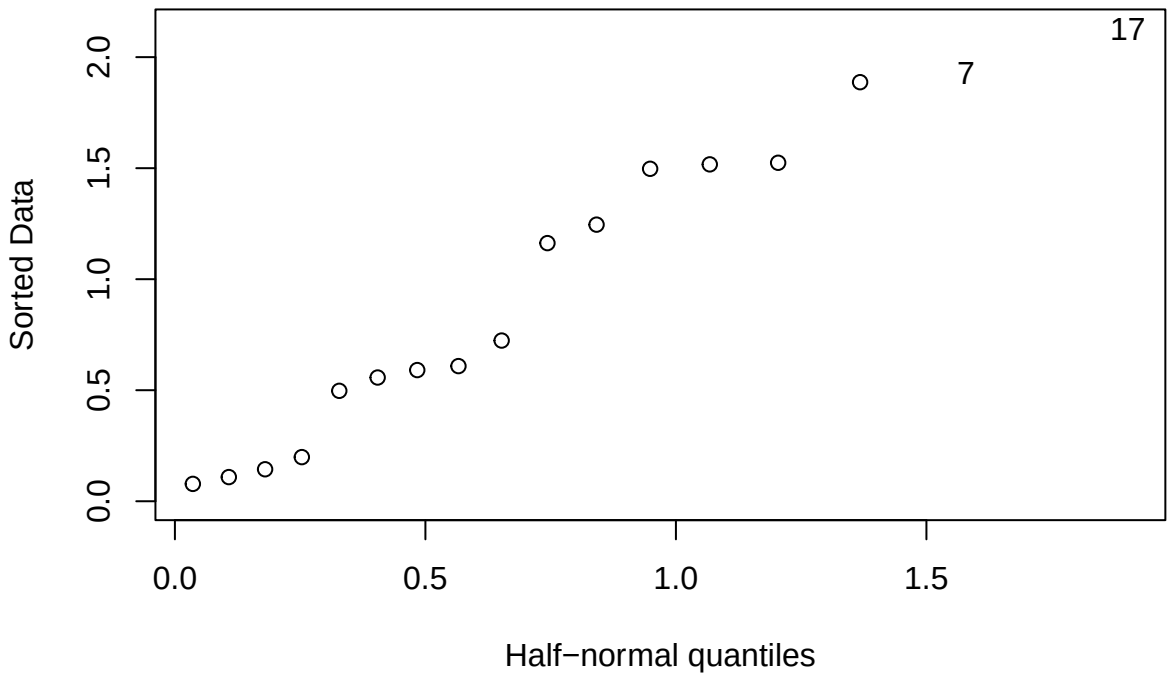
The deviance has decreased a lot, however, it is till too high. The p-value for the deviance based goodness-of-fit test is $(P(\chi^2_{15} > 46.347) = 0.00004681885 << 0.05$, therefore, the model does not fit the data. We already removed the potential outlier from the data, so now we explore the other possible cause of high deviance, which is an incorrectly specified mean structure $X\beta$. We fit the model:

$$\log(\mu_{\boldsymbol{x}}) = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \beta_4 x^4 + \beta_5 x^5$$

```
##
## Call:
## glm(formula = colonies ~ dose + I(dose^2) + I(dose^3) + I(dose^4) +
##     I(dose^5), family = poisson, data = salmonella_no_outlier)
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  3.076e+00  1.240e-01  24.798   <2e-16 ***
## dose        -3.058e-02  2.878e-02  -1.062    0.288
## I(dose^2)    1.545e-03  1.201e-03   1.287    0.198
## I(dose^3)   -1.625e-05  1.262e-05  -1.287    0.198
## I(dose^4)    4.622e-08  3.604e-08   1.282    0.200
## I(dose^5)   -3.148e-11  2.459e-11  -1.280    0.200
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 51.379  on 16  degrees of freedom
## Residual deviance: 23.465  on 11  degrees of freedom
## AIC: 122.07
##
## Number of Fisher Scoring iterations: 4
```

The deviance of this quite complicated model is still high, 23.465 on 11 degrees of freedom. The p-value for the deviance based goodness-of-fit test is ($P(\chi^2_{11} > 23.465) = 0.01518728 < 0.05$, therefore, the model does not fit the data.

Let's check for outliers in this model:



There is no outlier here.

We have ruled out both outliers and an incorrectly specified mean structure (we included powers of dose up to the fifth power, there is not much scope left for improvement) as explanations for high deviance, so we are left with the conclusion that there is overdispersion.

Now we go back to using the original data set (before we removed the twelfth observation). We want to introduce an overdispersion parameter σ^2 satisfying:

$$\text{var}(Y|X = x) = \sigma^2 \mu_x,$$

this method of dealing with over-dispersion assumes that σ^2 does not depend on x; since we have replicates, we can check if $\text{var}(Y|X = x) = \sigma^2 \mu_x$ is a reasonable modeling assumption.

For each covariate class, we use the (unbiased) sample variance to estimate $\text{var}(Y|X = x)$. The table below shows these estimated variances.

```
library(knitr)

covariate_classes = c(0,10,33,100,333,1000)
estimated_variances = c(var(c(15,21,29)), var(c(16,18,21)), var(c(16,26,33)), var(c(27,41,60)), var(c(33

est_var_df <- data.frame(
  `covariate class` = covariate_classes,
  `estimated variance` = estimated_variances
)
```

```r
# Display table using kable
kable(est_var_df, digits = 2, caption = "Estimated Variances by Covariate Class")
```

Table 1: Estimated Variances by Covariate Class

covariate.class	estimated.variance
0	49.33
10	6.33
33	73.00
100	274.33
333	16.33
1000	126.33

For each of the following model, we plot $\hat{\mu}_x$ against $\hat{\text{var}}(Y|X=x)$ (using the values in the table above).

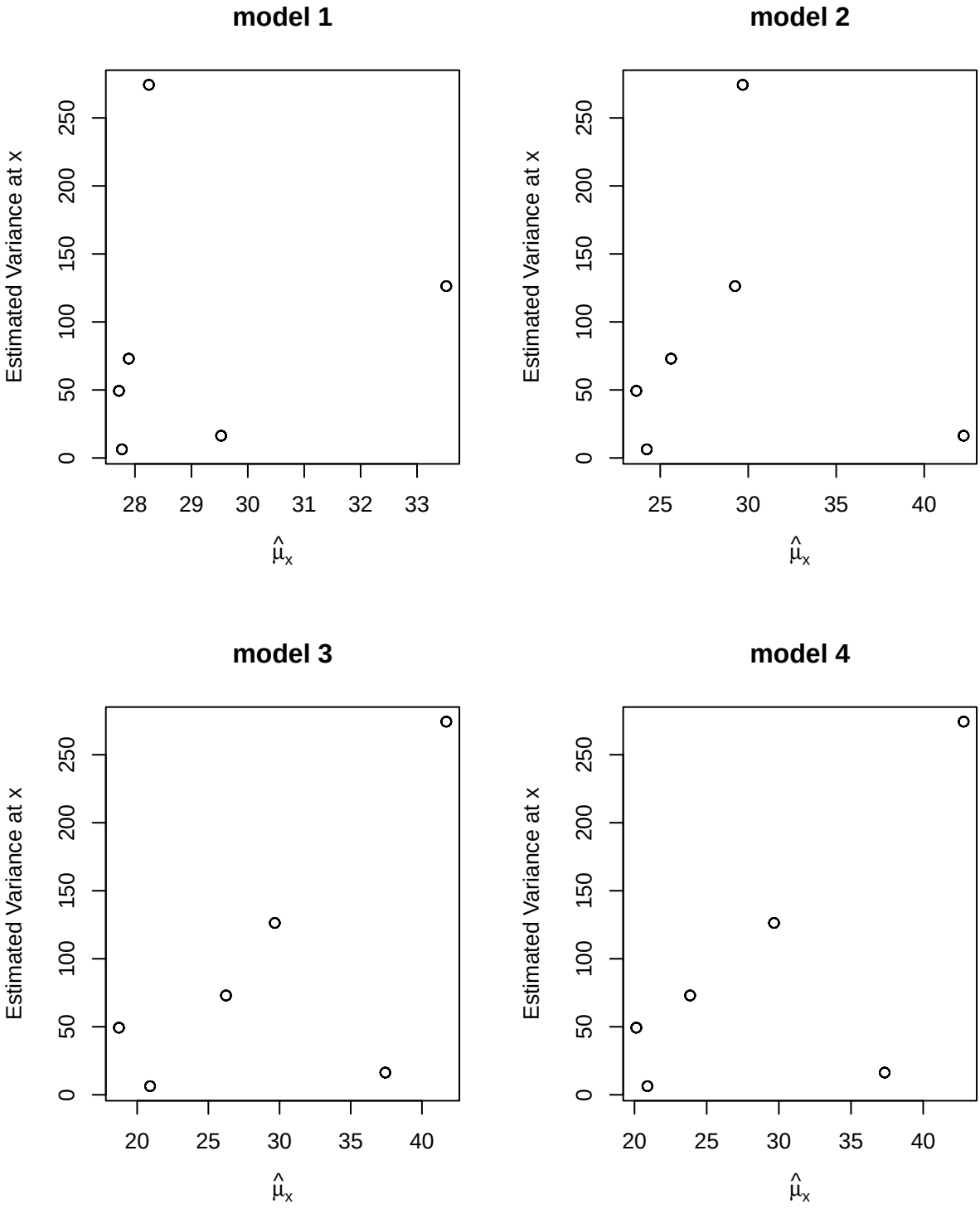
$$\text{model 1} : \log(\mu_{\boldsymbol{x}}) = \beta_0 + \beta_1 x$$

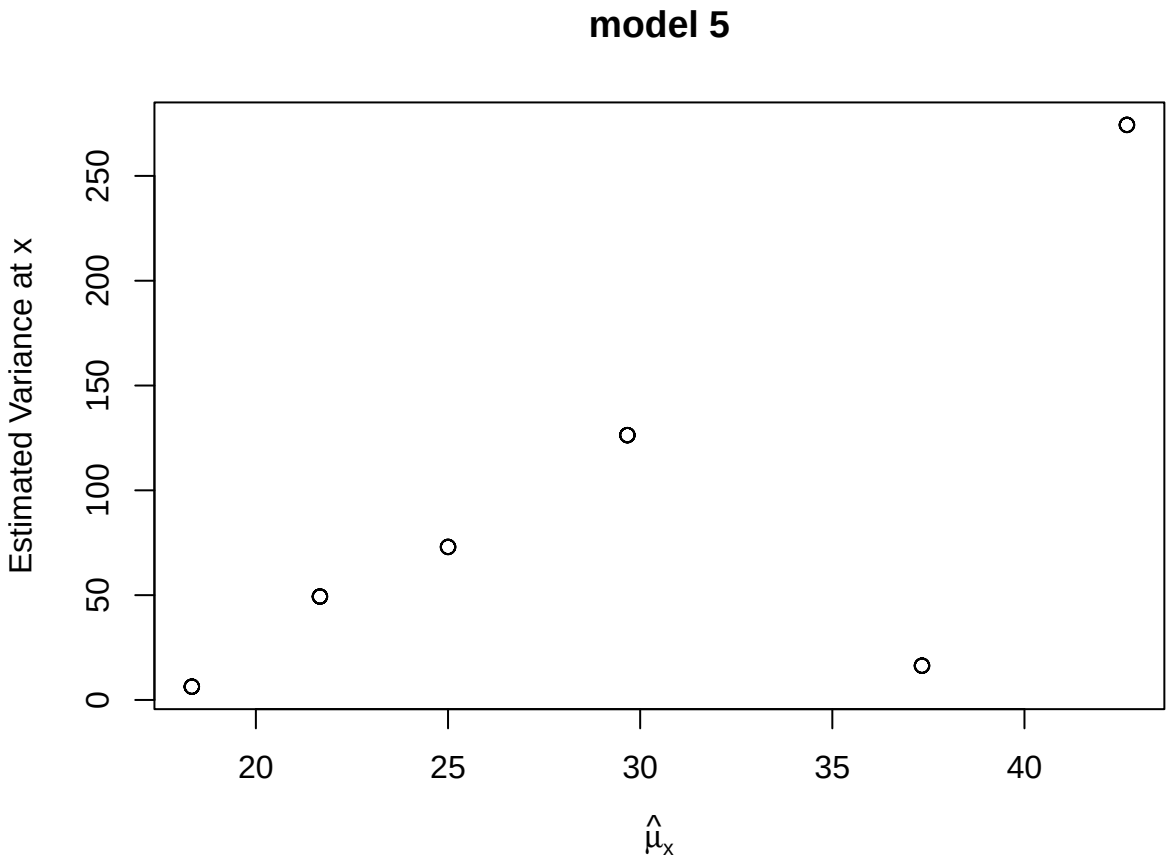
$$\text{model 2} : \log(\mu_{\boldsymbol{x}}) = \beta_0 + \beta_1 x + \beta_2 x^2$$

$$\text{model 3} : \log(\mu_{\boldsymbol{x}}) = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3$$

$$\text{model 4} : \log(\mu_{\boldsymbol{x}}) = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \beta_4 x^4$$

$$\text{model 5} : \log(\mu_{\boldsymbol{x}}) = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \beta_4 x^4 + \beta_5 x^5$$


model 1
Estimated Variance at x
0
50
100
150
200
250
28
29
30
31
32
33
$\hat{\mu}_x$
model 2
Estimated Variance at x
0
50
100
150
200
250
25
30
35
40
$\hat{\mu}_x$
model 3
Estimated Variance at x
0
50
100
150
200
250
20
25
30
35
40
$\hat{\mu}_x$
model 4
Estimated Variance at x
0
50
100
150
200
250
20
25
30
35
40
$\hat{\mu}_x$



For models 3,4, and 5, if we ignore the covariate class $dose = 333$ we can see from the plots that the assumption that $\text{var}(Y|X = x) \propto \sigma^2\mu_x$ is reasonable (because the points line up in a straight line). We re-fit all 3 models, for each model, we estimate the dispersion parameter by:

$$\hat{\sigma}^2 = \frac{\sum_i (y_i - \hat{\mu}_i)^2}{n - p}$$

The summary of each model with the estimated dispersion parameter are as follows (in the order of model 3,4,5):

```
##
## Call:
## glm(formula = colonies ~ dose + I(dose^2) + I(dose^3), family = poisson,
##     data = salmonella)
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  2.930e+00  1.453e-01  20.168  < 2e-16 ***
## dose         1.141e-02  3.314e-03   3.442 0.000577 ***
## I(dose^2)   -3.653e-05  1.229e-05  -2.973 0.002946 **
## I(dose^3)    2.558e-08  9.160e-09   2.793 0.005226 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## (Dispersion parameter for poisson family taken to be 2.61209)
##
##     Null deviance: 78.358  on 17  degrees of freedom
## Residual deviance: 36.055  on 14  degrees of freedom
## AIC: 136.59
##
## Number of Fisher Scoring iterations: 4

##
## Call:
## glm(formula = colonies ~ dose + I(dose^2) + I(dose^3) + I(dose^4),
##     family = poisson, data = salmonella)
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  3.002e+00  1.846e-01  16.261   <2e-16 ***
## dose         3.049e-03  1.380e-02   0.221    0.825
## I(dose^2)    7.357e-05  1.772e-04   0.415    0.678
## I(dose^3)   -3.093e-07  5.380e-07  -0.575    0.565
## I(dose^4)    2.331e-10  3.745e-10   0.622    0.534
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 2.715767)
##
##     Null deviance: 78.358  on 17  degrees of freedom
## Residual deviance: 34.989  on 13  degrees of freedom
## AIC: 137.53
##
## Number of Fisher Scoring iterations: 4

##
## Call:
## glm(formula = colonies ~ dose + I(dose^2) + I(dose^3) + I(dose^4) +
##     I(dose^5), family = poisson, data = salmonella)
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  3.076e+00  2.067e-01  14.878   <2e-16 ***
## dose        -3.038e-02  4.797e-02  -0.633    0.527
## I(dose^2)    1.518e-03  2.001e-03   0.759    0.448
## I(dose^3)   -1.555e-05  2.103e-05  -0.739    0.460
## I(dose^4)    4.375e-08  6.005e-08   0.729    0.466
## I(dose^5)   -2.969e-11  4.097e-11  -0.725    0.469
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 2.777999)
##
##     Null deviance: 78.358  on 17  degrees of freedom
## Residual deviance: 33.496  on 12  degrees of freedom
## AIC: 138.03
##
```

```
## Number of Fisher Scoring iterations: 4
```

Comparing all 3 models, we can see that model 3 fits the best. It has the lowest AIC and all its predictors are significant. Therefore, our final model is:

$$\log(\hat{\mu}_{\boldsymbol{x}}) = 2.93 + (1.41 \times 10^{-2})x + (-3.653 \times 10^{-5})x^2 + (2.558 \times 10^{-8})x^3,$$

$$\text{var}(Y|X = x) = 2.61209 \times \hat{\mu}_{\boldsymbol{x}},\ \ \hat{\sigma}^2 = 2.61209$$

3.

Denote age as X and marital status as M and count as Y. We model this data using Poisson count regression, the covariates we use are X and M, so that the data set has $8 \times 3 = 24$ covariate classes.

For this question, we use the mid-point of the age groups as the age covariate X (for the last interval, we assume a somewhat arbitrary upper bound of 80 years old, this is based on Denmark's life expectancy). This assumes a linear effect of moving up one age group, however, we can later add quadratic, cubic, etc. terms of this predictor to account for possible non-linear effects. For marital status, we used reference coding with single being the reference.

Our model is:

$$Y|X = x, M = m \sim \text{Poisson}(\mu_{x,m})$$

(so $\mu_{x,m} = E(Y|X = x, M = m)$)

$$\log(\mu_{x,m}) = \beta_0 + \beta_1 \text{married} + \beta_2 \text{divorced} + \sum \psi_i f_i(x) + \sum \phi_i g_i(x, \text{married}) + \sum \eta_i h_i(x, \text{married})$$

$$\log(\mu_{x,m}) = \beta_0 + \beta_1 \text{married} + \beta_2 \text{divorced} + \sum \psi_i f_i(x) + \sum \phi_i g_i(x, \text{married}) + \sum \eta_i h_i(x, \text{married})$$

We first fit the most basic model:

$$\log(\mu_{x,m}) = \beta_0 + \beta_1 \text{married} + \beta_2 \text{divorced} + \beta_3 x + \beta_4 (x \cdot \text{married}) + \beta_5 (x \cdot \text{divorced})$$

```
##
## Call:
## glm(formula = count ~ age + married + divorced + I(age * married) +
##     I(age * divorced), family = poisson, data = marital_counts)
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)        3.74192    0.32746  11.427  < 2e-16 ***
## age               -0.04967    0.01014  -4.898 9.66e-07 ***
## married           -1.11207    0.40989  -2.713  0.00667 **
## divorced          -3.95973    0.65580  -6.038 1.56e-09 ***
## I(age * married)   0.04650    0.01147   4.054 5.03e-05 ***
## I(age * divorced)  0.08111    0.01412   5.744 9.25e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 142.757  on 23  degrees of freedom
## Residual deviance:  60.151  on 18  degrees of freedom
## AIC: 152.19
##
## Number of Fisher Scoring iterations: 5
```

All predictors are highly significant, the p-values are all virtually zero. The deviance is 60.151 on 18 degrees of freedom, a hypothesis test testing goodness-of-fit would have p-value $P(\chi^2_{18} > 60.151) = 1.93 \times 10^{-6} < 0.05$, indicating that the model doesn't fit the data. We continue to fit another model to account for possible quadratic effects of age:

$$\log(\mu_{x,m}) = \beta_0 + \beta_1 \text{married} + \beta_2 \text{divorced} + \beta_3 L(x) + \beta_4 (L(x) \cdot \text{married}) + \beta_5 (L(x) \cdot \text{divorced})$$
$$+\beta_6 Q(x) + \beta_7 (Q(x) \cdot \text{married}) + \beta_8 (Q(x) \cdot \text{divorced})$$

We use orthogonal polynomials to code the effects of age, L being a linear polynomial, the second being a quadratic polynomial.

```
##
## Call:
## glm(formula = count ~ poly(age, 2) + married + divorced + married:poly(age,
##     2) + divorced:poly(age, 2), family = poisson, data = marital_counts)
##
## Coefficients:
##                          Estimate Std. Error z value Pr(>|z|)
## (Intercept)               1.6258     0.1892   8.593  < 2e-16 ***
## poly(age, 2)1            -4.3482     1.0905  -3.987 6.68e-05 ***
## poly(age, 2)2             0.4238     0.8579   0.494  0.62128
## married                   0.6246     0.2300   2.715  0.00662 **
## divorced                 -0.8574     0.3707  -2.313  0.02071 *
## poly(age, 2)1:married     3.3418     1.3102   2.551  0.01075 *
## poly(age, 2)2:married    -4.0452     1.0877  -3.719  0.00020 ***
## poly(age, 2)1:divorced    9.3722     1.8877   4.965 6.87e-07 ***
## poly(age, 2)2:divorced   -3.7884     1.4432  -2.625  0.00867 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 142.757  on 23  degrees of freedom
## Residual deviance:  11.623  on 15  degrees of freedom
## AIC: 109.66
##
## Number of Fisher Scoring iterations: 5
```

This model fits better, all predictors are significant except that quadratic term for age. The residual deviance is 11.623 on 15 degrees of freedom, the p-value for a goodness-of-fit test is $P(\chi^2_{15} > 11.186) = 0.7073116$, indicating that the model fits the data. The interaction terms 'poly(age, 2)2:married' and 'poly(age, 2)1:divorced' are highly significant, indicating that *age* is important for marriage status.

We continue to fit two more models, model 3 and model 4, to check whether higher order effects of *age* might improve the model further:

$$\text{model 3}: \log(\mu_{x,m}) = \beta_0 + \beta_1 \text{married} + \beta_2 \text{divorced} + \beta_3 L(x) + \beta_4 (L(x) \cdot \text{married}) + \beta_5 (L(x) \cdot \text{divorced})$$
$$+\beta_6 Q(x) + \beta_7 (Q(x) \cdot \text{married}) + \beta_8 (Q(x) \cdot \text{divorced})$$
$$+\beta_9 C(x) + \beta_{10} (C(x) \cdot \text{married}) + \beta_{11} (C(x) \cdot \text{divorced})$$

$$\text{model 4}: \log(\mu_{x,m}) = \beta_0 + \beta_1 \text{married} + \beta_2 \text{divorced} + \beta_3 L(x) + \beta_4 (L(x) \cdot \text{married}) + \beta_5 (L(x) \cdot \text{divorced})$$
$$+\beta_6 Q(x) + \beta_7 (Q(x) \cdot \text{married}) + \beta_8 (Q(x) \cdot \text{divorced})$$
$$+\beta_9 C(x) + \beta_{10} (C(x) \cdot \text{married}) + \beta_{11} (C(x) \cdot \text{divorced})$$
$$+\beta_{12} P_4(x) + \beta_{13} (P_4(x) \cdot \text{married}) + \beta_{14} (P_4(x) \cdot \text{divorced})$$

The summaries for model 3 and model 4 are below:

```
##
## Call:
## glm(formula = count ~ poly(age, 3) + married + divorced + married:poly(age,
##     3) + divorced:poly(age, 3), family = poisson, data = marital_counts)
##
## Coefficients:
##                         Estimate Std. Error z value Pr(>|z|)
## (Intercept)              1.61697    0.19439   8.318  < 2e-16 ***
## poly(age, 3)1           -4.47141    1.18833  -3.763 0.000168 ***
## poly(age, 3)2            0.05889    1.11189   0.053 0.957759
## poly(age, 3)3           -0.49085    0.84290  -0.582 0.560342
## married                  0.63386    0.23304   2.720 0.006529 **
## divorced                -1.11992    0.51045  -2.194 0.028238 *
## poly(age, 3)1:married    4.16345    1.42205   2.928 0.003414 **
## poly(age, 3)2:married   -3.46202    1.28191  -2.701 0.006920 **
## poly(age, 3)3:married    1.56447    1.04212   1.501 0.133296
## poly(age, 3)1:divorced 11.24053    2.78453   4.037 5.42e-05 ***
## poly(age, 3)2:divorced -4.82052    2.17163  -2.220 0.026434 *
## poly(age, 3)3:divorced  2.09865    1.57551   1.332 0.182845
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 142.757  on 23  degrees of freedom
## Residual deviance:   6.599  on 12  degrees of freedom
## AIC: 110.64
##
## Number of Fisher Scoring iterations: 4

##
## Call:
## glm(formula = count ~ poly(age, 4) + married + divorced + married:poly(age,
##     4) + divorced:poly(age, 4), family = poisson, data = marital_counts)
##
## Coefficients:
##                         Estimate Std. Error z value Pr(>|z|)
```

```
## (Intercept)             1.61446    0.19621   8.228  < 2e-16 ***
## poly(age, 4)1          -4.49373    1.21343  -3.703 0.000213 ***
## poly(age, 4)2           0.02562    1.15652   0.022 0.982327
## poly(age, 4)3          -0.57604    1.02280  -0.563 0.573299
## poly(age, 4)4          -0.12260    0.81872  -0.150 0.880967
## married                 0.57804    0.24256   2.383 0.017171 *
## divorced               -1.34624    0.74520  -1.807 0.070832 .
## poly(age, 4)1:married   4.31383    1.49360   2.888 0.003874 **
## poly(age, 4)2:married  -4.04582    1.42397  -2.841 0.004494 **
## poly(age, 4)3:married   1.65647    1.21902   1.359 0.174193
## poly(age, 4)4:married  -0.84306    1.02022  -0.826 0.408607
## poly(age, 4)1:divorced 12.64101    4.21193   3.001 0.002689 **
## poly(age, 4)2:divorced -6.09837    3.52585  -1.730 0.083699 .
## poly(age, 4)3:divorced  3.25035    2.67828   1.214 0.224902
## poly(age, 4)4:divorced -0.75150    1.71437  -0.438 0.661132
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 142.757  on 23  degrees of freedom
## Residual deviance:   3.612  on  9  degrees of freedom
## AIC: 113.65
##
## Number of Fisher Scoring iterations: 5
```

We can see that models 3 and 4 overfit, there are too many predictors. Many predictors are not significant, and no terms involving third or fourth degree polynomials are insignificant. We can be reasonably certain that the model would not benefit from higher order effects of *age*.

So we return to the model:

$$\log(\mu_{x,m}) = \beta_0 + \beta_1 \text{married} + \beta_2 \text{divorced} + \beta_3 L(x) + \beta_4 (L(x) \cdot \text{married}) + \beta_5 (L(x) \cdot \text{divorced}) \\ + \beta_6 Q(x) + \beta_7 (Q(x) \cdot \text{married}) + \beta_8 (Q(x) \cdot \text{divorced}),$$

which is our final model.

Its summary is as follows:

```
##
## Call:
## glm(formula = count ~ poly(age, 2) + married + divorced + married:poly(age,
##     2) + divorced:poly(age, 2), family = poisson, data = marital_counts)
##
## Coefficients:
##                         Estimate Std. Error z value Pr(>|z|)
## (Intercept)              1.6258     0.1892   8.593  < 2e-16 ***
## poly(age, 2)1           -4.3482     1.0905  -3.987 6.68e-05 ***
## poly(age, 2)2            0.4238     0.8579   0.494  0.62128
## married                  0.6246     0.2300   2.715  0.00662 **
## divorced                -0.8574     0.3707  -2.313  0.02071 *
## poly(age, 2)1:married    3.3418     1.3102   2.551  0.01075 *
## poly(age, 2)2:married   -4.0452     1.0877  -3.719  0.00020 ***
## poly(age, 2)1:divorced   9.3722     1.8877   4.965 6.87e-07 ***
```

```
## poly(age, 2)2:divorced  -3.7884     1.4432  -2.625  0.00867 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 142.757  on 23  degrees of freedom
## Residual deviance:  11.623  on 15  degrees of freedom
## AIC: 109.66
##
## Number of Fisher Scoring iterations: 5
```

Therefore, our final model is:

$$\log(\mu_{x,m}) = 1.6258 + 0.6246\text{married} - 0.8574\text{divorced} - 4.3482L(x) + 3.3418(L(x)\cdot\text{married}) + 9.3722(L(x)\cdot\text{divorced})$$
$$+0.4238Q(x) - 4.0452(Q(x)\cdot\text{married}) - 3.7884(Q(x)\cdot\text{divorced})$$

We can interpret this model separately for each marital status:

$$\log(\mu_{x,\text{single}}) = 1.6258 - 4.3482L(x) + 0.4238Q(x),$$

$$\log(\mu_{x,\text{married}}) = 2.2504 - 1.0064L(x) - 3.6214Q(x),$$

$$\log(\mu_{x,\text{divorced}}) = 0.7684 + 5.024L(x) - 3.3646Q(x),$$

For a 55 year old Dane, their X variable is coded as 55, and the corresponding orthogonal polynomial values are $L(55) = 0.12719164$ and $Q(55) = -0.21350422$. So:

$$\log(\hat{\mu}_{55,\text{single}}) = 1.6258 - 4.3482 \cdot 0.12719164 + 0.4238 \cdot -0.21350422 = 0.9822622,$$
$$\log(\hat{\mu}_{55,\text{married}}) = 2.2504 - 1.0064 \cdot 0.12719164 - 3.6214 \cdot -0.21350422 = 2.895579$$
$$\log(\hat{\mu}_{55,\text{divorced}}) = 0.7684 + 5.024 \cdot 0.12719164 - 3.3646 \cdot -0.21350422 = 2.125767 \implies$$

To estimate the probability that a 55 year old Danish person is divorced, we use the property on page 11 of chapter 4's slides.

So:

$$\hat{p}_{55,divorced} = \frac{\hat{\mu}_{55,divorced}}{\hat{\mu}_{55,divorced} + \hat{\mu}_{55,single} + \hat{\mu}_{55,married}} = \frac{\exp(0.9822622)}{\exp(0.9822622) + \exp(2.895579) + \exp(2.125767)} = 0.2875166$$

The actual values of the polynomials used for age can be found using the code:

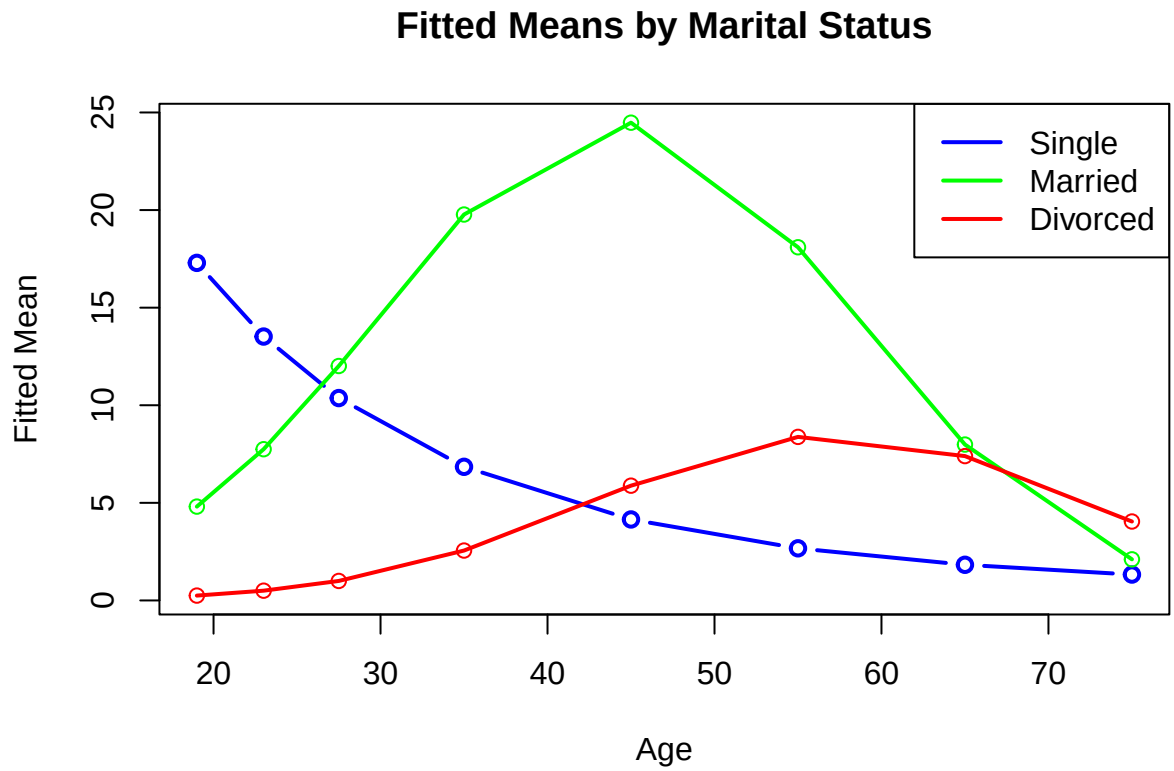
```
marital_counts = read.csv('/Users/Noppawee/Documents/NTHU/categorical_data/marital_counts')
age_poly <- poly(marital_counts$age, 2)
age_poly
```

```
##                 1           2
##  [1,] -0.25638105  0.25908017
##  [2,] -0.25638105  0.25908017
##  [3,] -0.25638105  0.25908017
##  [4,] -0.21376186  0.11544716
##  [5,] -0.21376186  0.11544716
##  [6,] -0.21376186  0.11544716
##  [7,] -0.16581527 -0.01890968
##  [8,] -0.16581527 -0.01890968
##  [9,] -0.16581527 -0.01890968
## [10,] -0.08590430 -0.17876643
## [11,] -0.08590430 -0.17876643
## [12,] -0.08590430 -0.17876643
## [13,]  0.02064367 -0.26732567
## [14,]  0.02064367 -0.26732567
## [15,]  0.02064367 -0.26732567
## [16,]  0.12719164 -0.21350422
## [17,]  0.12719164 -0.21350422
## [18,]  0.12719164 -0.21350422
## [19,]  0.23373960 -0.01730208
## [20,]  0.23373960 -0.01730208
## [21,]  0.23373960 -0.01730208
## [22,]  0.34028757  0.32128074
## [23,]  0.34028757  0.32128074
## [24,]  0.34028757  0.32128074
## attr(,"coefs")
## attr(,"coefs")$alpha
## [1] 43.06250 49.37728
##
## attr(,"coefs")$norm2
## [1]       1.000      24.000    8808.656 1973139.683
##
## attr(,"degree")
## [1] 1 2
## attr(,"class")
## [1] "poly"   "matrix"
```

Below we have made plots of predictions based on our final model. Notice that for single people, the log of the fitted mean $\log(\hat{\mu}_{Age}) = \hat{\eta}_{Age}$ closely resembles a linear function, but for married and divorced people, their curves are quadratic functions. This reflects the fact that in our final model, the quadratic effect for single people has a small estimated value.

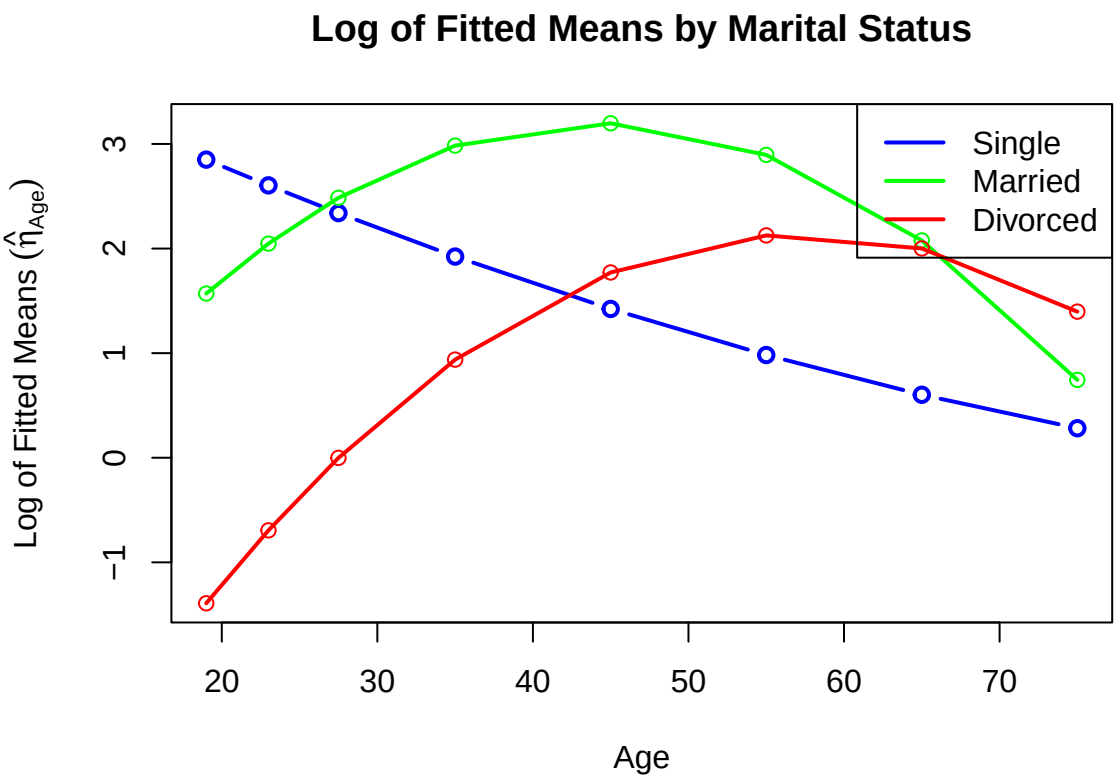
```
Denmark_covariate_classes = c(19,23,27.5,35,45,55,65,75)
fitted_mean_single = fitted(modp)[which(marital_counts$married == 0 & marital_counts$divorced == 0)]
fitted_mean_married = fitted(modp)[which(marital_counts$married == 1 & marital_counts$divorced == 0)]
fitted_mean_divorced = fitted(modp)[which(marital_counts$married == 0 & marital_counts$divorced == 1)]

log_fitted_mean_single = log(fitted_mean_single)
log_fitted_mean_married = log(fitted_mean_married)
log_fitted_mean_divorced = log(fitted_mean_divorced)

fitted_single_prob = fitted_mean_single/(fitted_mean_single + fitted_mean_married + fitted_mean_divorce
fitted_married_prob = fitted_mean_married/(fitted_mean_single + fitted_mean_married + fitted_mean_divor
fitted_divorced_prob = fitted_mean_divorced/(fitted_mean_single + fitted_mean_married + fitted_mean_div
```

```r
# Base plot with the first curve (e.g., single)
plot(
  Denmark_covariate_classes, fitted_mean_single,
  type = "b", col = "blue", lwd = 2,
  ylim = range(c(fitted_mean_single, fitted_mean_married, fitted_mean_divorced)),
  xlab = "Age", ylab = "Fitted Mean",
  main = "Fitted Means by Marital Status"
)

# Add the other two curves
lines(Denmark_covariate_classes, fitted_mean_married, col = "green", lwd = 2)
points(Denmark_covariate_classes, fitted_mean_married, col = "green")

lines(Denmark_covariate_classes, fitted_mean_divorced, col = "red", lwd = 2)
points(Denmark_covariate_classes, fitted_mean_divorced, col = "red")

# Add a legend
legend("topright",
       legend = c("Single", "Married", "Divorced"),
       col = c("blue", "green", "red"),
       lty = 1, lwd = 2)
```




```r
par(mar = c(5, 6, 4, 2))
# Base plot: log fitted means for single
plot(
```

```
  Denmark_covariate_classes, log_fitted_mean_single,
  type = "b", col = "blue", lwd = 2,
  ylim = range(c(log_fitted_mean_single, log_fitted_mean_married, log_fitted_mean_divorced)),
  xlab = "Age", ylab = expression(Log~of~Fitted~Means~(hat(eta)[Age])),
  main = 'Log of Fitted Means by Marital Status'
)

# Add lines for married and divorced
lines(Denmark_covariate_classes, log_fitted_mean_married, col = "green", lwd = 2)
points(Denmark_covariate_classes, log_fitted_mean_married, col = "green")

lines(Denmark_covariate_classes, log_fitted_mean_divorced, col = "red", lwd = 2)
points(Denmark_covariate_classes, log_fitted_mean_divorced, col = "red")

# Add a legend
legend("topright",
       legend = c("Single", "Married", "Divorced"),
       col = c("blue", "green", "red"),
       lty = 1, lwd = 2)
```




```
# Plot predicted probability for 'single'
plot(
  Denmark_covariate_classes, fitted_single_prob,
  type = "b", col = "blue", lwd = 2,
  ylim = c(0, 1),
```

```
  xlab = "Age", ylab = "Predicted Probability",
  main = "Predicted Probabilities by Marital Status"
)

# Add the other lines
lines(Denmark_covariate_classes, fitted_married_prob, col = "green", lwd = 2)
points(Denmark_covariate_classes, fitted_married_prob, col = "green")

lines(Denmark_covariate_classes, fitted_divorced_prob, col = "red", lwd = 2)
points(Denmark_covariate_classes, fitted_divorced_prob, col = "red")

# Add legend
legend("topright",
       legend = c("Single", "Married", "Divorced"),
       col = c("blue", "green", "red"),
       lty = 1, lwd = 2)
```