C3M2: Peer Reviewed Assignment

Outline:

The objectives for this assignment:

1. Apply Poisson Regression to real data.
2. Learn and practice working with and interpreting Poisson Regression Models.
3. Understand deviance and how to conduct hypothesis tests with Poisson Regression.
4. Recognize when a model shows signs of overdispersion.

General tips:

1. Read the questions carefully to understand what is being asked.
2. This work will be reviewed by another human, so make sure that you are clear and concise in what your explanations and answers.

In [1]:
```
# Load the required packages
library(MASS)
```

Problem 1: Poisson Estimators

Let $Y_1, . . . , Y_n$i ~ Poisson(λ_i). Show that, if $\eta_i = \beta_0$, then the maximum likelihood estimator of λ_i is ˆλ_i = ¯Y, for all i = 1, . . . , n.

Solution

In order to show the above, we must find the solution that maximizes ˆλ = argmax l(λ; $y_1, . . . , y_n$)

Assuming independence, the probability mass function is given by: $\lambda^{y}e^{-\lambda}y!$

The joint PMF is $\Pi_{ni=1}\lambda^{y_ie^{-\lambda}}y_i!$ Take the log of both sides of this likelihood function:

log($\Pi_{ni=1}\lambda^{y_ie^{-\lambda}}y_i!$) = n∑i=1log($\lambda^{x_ie^{-\lambda}}x_i!$) = n∑i=1$y_i$log(λ) − n∑i=1λ − n∑i=1log(y_i!)
 = log(λ)n∑i=1y_i − n∑i=1log(y_i!) − nλ Then we differentiate to first order, set it equal to zero, and solve for λ. Note that the differentiation is with respect to λ and so things become simpler.
0 = $^{1}\lambda$n∑i=1y_i − n nλ = n∑i=1y_i λ = $\Sigma_{ni=1}{}^{y_i n}$ = ¯y Thus, the maximum likelihood estimator of λ_i is simply the sample mean: ^λ_i = ¯Y

Problem 2: Ships data

The ships dataset gives the number of damage incidents and aggregate months of service for different types of ships broken down by year of construction and period of operation.

The code below splits the data into a training set (80% of the data) and a test set (the remaining 20%).

In [2]:

```
data(ships)
ships = ships[ships$service != 0,]
ships$year = as.factor(ships$year)
ships$period = as.factor(ships$period)

set.seed(11)
n = floor(0.8 * nrow(ships))
index = sample(seq_len(nrow(ships)), size = n)

train = ships[index, ]
test = ships[-index, ]
head(train)
summary(train)
```

A data.frame: 6 × 5

	type	year	period	service	incidents
	<fct>	<fct>	<fct>	<int>	<int>
40	E	75	75	542	1
28	D	65	75	192	0
18	C	60	75	552	1
19	C	65	60	781	0
5	A	70	60	1512	6
32	D	75	75	2051	4

```
 type    year     period      service            incidents    
 A:5     60:7     60:11    Min.    :   45.0    Min.    : 0.00  
 B:5     65:8     75:16    1st Qu.:  318.5    1st Qu.: 0.50  
 C:6     70:8              Median : 1095.0    Median : 2.00  
 D:7     75:4              Mean   : 5012.2    Mean   :10.63  
 E:4                       3rd Qu.: 2202.5    3rd Qu.:11.50  
                           Max.   :44882.0    Max.   :58.00  
```

2. (a) Poisson Regression Fitting

Use the training set to develop an appropriate regression model for `incidents`, using `type`, `period`, and `year` as predictors (HINT: is this a count model or a rate model?).

Calculate the mean squared prediction error (MSPE) for the test set. Display your results.

In [12]:

```
# Your Code Here
mod1 = glm(incidents ~ type + period + year, data = train, family = 'poisson')
```

```r
mod1
summary(mod1)
#prediction
pred1 = predict(mod1, test, type = 'response')
MSPE = sum((test$incidents - pred1)^2)/length(test$incidents)
#MSPE2 = mean((test$incidents - pred1)^2)
MSPE
#MSPE2
```

```
Call:  glm(formula = incidents ~ type + period + year, family = "poisson",

    data = train)

Coefficients:
(Intercept)        typeB        typeC        typeD        typeE     period
75
     1.5644       1.6795      -2.0789      -1.1551      -0.5113        0.41
23
     year65       year70       year75
     0.4379       0.2260       0.1436

Degrees of Freedom: 26 Total (i.e. Null);  18 Residual
Null Deviance:      554.7
Residual Deviance: 109.2        AIC: 200.9

Call:
glm(formula = incidents ~ type + period + year, family = "poisson",
    data = train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-4.0775  -1.9869  -0.0418   0.7612   3.6618

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)   1.5644     0.2199   7.113 1.13e-12 ***
typeB         1.6795     0.1889   8.889  < 2e-16 ***
typeC        -2.0789     0.4408  -4.717 2.40e-06 ***
typeD        -1.1551     0.2930  -3.943 8.06e-05 ***
typeE        -0.5113     0.2781  -1.839   0.0660 .
period75      0.4123     0.1282   3.216   0.0013 **
year65        0.4379     0.1885   2.324   0.0201 *
year70        0.2260     0.1916   1.180   0.2382
year75        0.1436     0.3147   0.456   0.6481
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 554.70  on 26  degrees of freedom
Residual deviance: 109.21  on 18  degrees of freedom
AIC: 200.92

Number of Fisher Scoring iterations: 6
```

131.077556337426

Solution

First a generalized linear model is fit and given the training set with poisson distribution. Taking predicted values from this model using the test data, we get a residual deviance of 109.21 and the mean squared prediction error came out to be 131.08.

2. (b) Poisson Regression Model Selection

Do we really need all of these predictors? Construct a new regression model leaving out `year` and calculate the MSE for this second model.

Decide which model is better. Explain why you chose the model that you did.

In [14]:

```
# Your Code Here


# Exclude 'year'
mod2 = glm(incidents ~ type + period, data = train, family = 'poisson')
mod2
summary(mod2)

pred2 = predict(mod2, test, type = 'response')
MSPE2 = sum((test$incidents - pred2)^2)/length(test$incidents)
MSPE2
```

```
Call:  glm(formula = incidents ~ type + period, family = "poisson",
    data = train)

Coefficients:
(Intercept)        typeB        typeC        typeD        typeE      period
75
     1.7190       1.7831      -2.0573      -1.1281      -0.4831        0.47
23

Degrees of Freedom: 26 Total (i.e. Null);  21 Residual
Null Deviance:      554.7
Residual Deviance: 115.6        AIC: 201.3

Call:
glm(formula = incidents ~ type + period, family = "poisson",
    data = train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-4.2377  -1.9003  -0.1372   0.6377   3.8906

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)   1.7190     0.1838   9.355  < 2e-16 ***
typeB         1.7831     0.1781  10.014  < 2e-16 ***
typeC        -2.0573     0.4394  -4.683 2.83e-06 ***
typeD        -1.1281     0.2918  -3.866 0.000111 ***
typeE        -0.4831     0.2767  -1.746 0.080787 .
period75      0.4723     0.1222   3.865 0.000111 ***
---
```

```
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 554.70  on 26  degrees of freedom
Residual deviance: 115.63  on 21  degrees of freedom
AIC: 201.34

Number of Fisher Scoring iterations: 6
```

275.122550627591

In [18]:

```
# Can compare nested poisson models with a chi-squared
#?pchisq
anova(mod1, mod2, test='Chisq')
```

A anova: 2 × 5

	Resid. Df	Resid. Dev	Df	Deviance	Pr(>Chi)
	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	18	109.2123	NA	NA	NA
2	21	115.6311	-3	-6.418801	0.09292038

The above analysis of variance table shows that the newer model without 'year' has a p-value of 0.09, which is greater than the typical alpha value of 0.05, which means there is not enough statistical evidence to reject the null hypothesis and that the simpler model is a better model.

2. (c) Deviance

How do we determine if our model is explaining anything? With linear regression, we had a F-test, but we can't do that for Poisson Regression. If we want to check if our model is better than the null model, then we're going to have to check directly. In particular, we need to compare the deviances of the models to see if they're significantly different.

Conduct two χ^2 tests (using the deviance). Let α = 0.05:

1. Test the adequacy of null model.
2. Test the adequacy of your chosen model agaisnt the saturated model (the model fit to all predictors).

What conclusions should you draw from these tests?

In [25]:

```
# Your Code Here
# Test if the model is better than the null model
q1 = sum((train$incidents - fitted(mod1))^2/fitted(mod1))
q1
mod1$df.residual
# Test chi_sq stat
1 - pchisq(q1, df = 18)
# Test against the saturated model
```

```r
mod3 = glm(incidents ~ ., data = train, family = 'poisson')
mod3
summary(mod3)
anova(mod1, mod3, test = 'Chisq')
```

98.4689210202285

18

4.22106793962485e-13

```
Call:  glm(formula = incidents ~ ., family = "poisson", data = train)

Coefficients:
(Intercept)        typeB        typeC        typeD        typeE       year
65  
 -1.2619390   -0.2351831   -1.7277785   -0.8340264   -0.4230541    2.24249
75  
     year70       year75     period75      service  
  2.9910064    2.2365673    0.8557250    0.0001153  

Degrees of Freedom: 26 Total (i.e. Null);  17 Residual
Null Deviance:	    554.7 
Residual Deviance: 27.82 	AIC: 121.5
```

```
Call:
glm(formula = incidents ~ ., family = "poisson", data = train)

Deviance Residuals: 
    Min       1Q   Median       3Q      Max  
-2.3603  -0.5990  -0.1924   0.2763   1.9552  

Coefficients:
              Estimate Std. Error z value Pr(>|z|)    
(Intercept) -1.262e+00  5.349e-01  -2.359  0.01831 *  
typeB       -2.352e-01  2.966e-01  -0.793  0.42790    
typeC       -1.728e+00  4.436e-01  -3.895 9.81e-05 ***
typeD       -8.340e-01  2.962e-01  -2.815  0.00487 ** 
typeE       -4.231e-01  2.804e-01  -1.509  0.13137    
year65       2.242e+00  3.419e-01   6.560 5.40e-11 ***
year70       2.991e+00  4.655e-01   6.426 1.31e-10 ***
year75       2.237e+00  5.053e-01   4.426 9.58e-06 ***
period75     8.557e-01  1.630e-01   5.249 1.53e-07 ***
service      1.153e-04  1.567e-05   7.358 1.87e-13 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 554.704  on 26  degrees of freedom
Residual deviance:  27.823  on 17  degrees of freedom
AIC: 121.53

Number of Fisher Scoring iterations: 5
```

A anova: 2 × 5

	Resid. Df	Resid. Dev	Df	Deviance	Pr(>Chi)

	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
1	18	109.21228	NA	NA	NA
2	17	27.82257	1	81.3897	1.853209e-19

The two χ^2 tests produced p-values of 4.22e-13 and 1.85e-19, respectively.
Thus, the model is a better one compared to the null model but not as good as the saturated model.

2. (d) Poisson Regression Visualizations

Just like with linear regression, we can use visualizations to assess the fit and appropriateness of our model. Is it maintaining the assumptions that it should be? Is there a discernable structure that isn't being accounted for? And, again like linear regression, it can be up to the user's interpretation what is an isn't a good model.

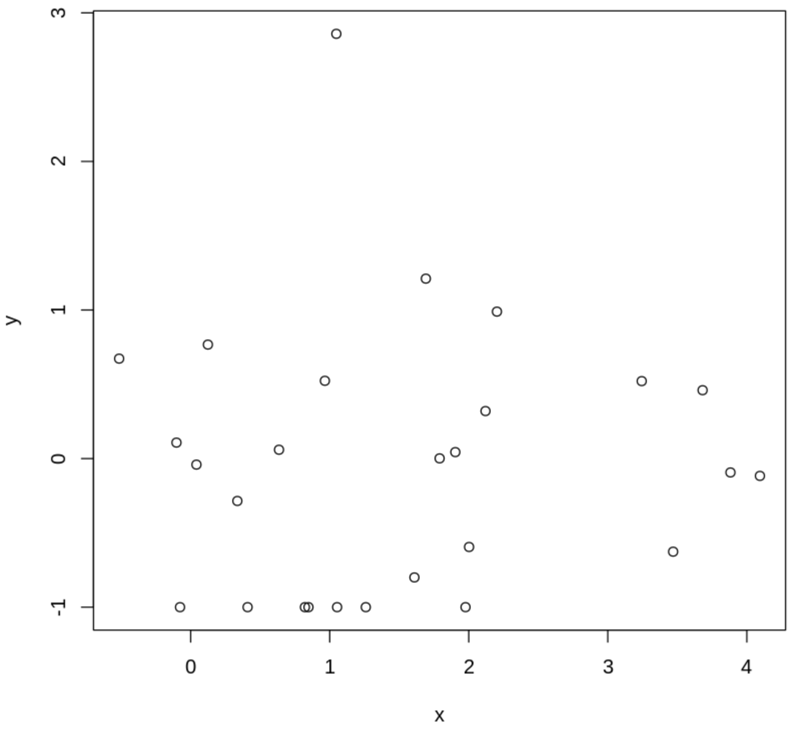
Plot the deviance residuals against the linear predictor η. Interpret this plot.

In [29]:
```
# Your Code Here

#mod1
#summary(mod1)

y = mod1$residual
x = mod1$linear.predictors

plot(x, y)
```



Apart from a single point anomaly (1.2, 2.8), the data points seem equally and consistently spread out in a roughly linear structure, indicating that there is evidence of being a good fit.

2. (e) Overdispersion

For linear regression, the variance of the data is controlled through the standard deviation σ, which is independent of the other parameters like the mean μ. However, some GLMs do not have this independence, which can lead to a problem called overdispersion. Overdispersion occurs when the observed data's variance is higher than expected, if the model is correct.

For Poisson Regression, we expect that the mean of the data should equal the variance. If overdispersion is present, then the assumptions of the model are not being met and we can not trust its output (or our beloved p-values)!

Explore the two models fit in the beginning of this question for evidence of overdisperion. If you find evidence of overdispersion, you do not need to fix it (but it would be useful for you to know how to). Describe your process and conclusions.

```
In [30]: # Your Code Here
         mod1
         mod2
```

```
Call:  glm(formula = incidents ~ type + period + year, family = "poisson", 

    data = train)

Coefficients:
(Intercept)        typeB        typeC        typeD        typeE     period
75  
     1.5644       1.6795      -2.0789      -1.1551      -0.5113       0.41
23  
     year65       year70       year75  
     0.4379       0.2260       0.1436  

Degrees of Freedom: 26 Total (i.e. Null);  18 Residual
Null Deviance:      554.7 
Residual Deviance: 109.2        AIC: 200.9

Call:  glm(formula = incidents ~ type + period, family = "poisson", 
    data = train)

Coefficients:
(Intercept)        typeB        typeC        typeD        typeE     period
75  
     1.7190       1.7831      -2.0573      -1.1281      -0.4831       0.47
23  

Degrees of Freedom: 26 Total (i.e. Null);  21 Residual
Null Deviance:      554.7 
Residual Deviance: 115.6        AIC: 201.3
```

Based on the dispersion parameter $\hat{}\phi = \Sigma_{ni=1}(y_i - \hat{}\lambda)^{2}\hat{}\lambda_{in-p+1}$, we can see that there will be overdispersion if the numerator, or residual deviance, is greater than the degrees of freedom. Both the models above have residual deviance values much greater than their degrees of freedom, creating overdispersion. We can use the quasilikelihood method described in the lecture, where we adjust the standard errors using the dispersion parameter to scale the variance.

In [ ]:

```

```