Computer exercise 4
Poisson Regression

When dealing with two or more variables, the functional relation between the variables is often of interest. For count data, one model that is frequently used is the Poisson regression model and applications are found in most sciences: technology, medicine etc. The Poisson regression model is also implemented in many packages for statistical analysis of data.

In this computer exercise you will learn more about:

(1) The Poisson regression model and how to estimate the model parameters

(2) Model selection, i.e. the number of explanatory variables to use

1 Preparatory exercises

1. Read chapter 7.1-7.3 in the textbook.

2. Try to explain the difference between linear regression and Poisson regression.

2 Road accident data

The Swedish Road Administration is the national authority that has the overall responsibility for the entire road transport system. One main issue is road safetybility and continuous work to improve road safety is performed. From their internet site http://www.trafikverket.se; it is possible to obtain a number of different statistics about road accidents[1]. We will in this exercise use traffic accident data from years 1950-2012. The data is used to fit a Poisson regression model to the number of people perished in traffic accidents, cf. Example 7.16 in the textbook. The estimated model is then used to predict the expected number of perished year 2016.

Start by download statistics about the number of people killed in road accidents reported by the police from the year 1950 to 2012. The data can be obtained from the mentioned website following the link: 1950-RoadData. However, the data obtained this way are in the format of an Excell spreadsheet that include also some description at the top. We modified this file to get it into more manageable format and now it is in `labfiles` named `'arsdata_1950.xls'`

Import the data to the Matlab workspace with the command

```
>> data = xlsread('arsdata_1950.xls');
```

[1] Another good source for all kinds of statistics about transport and communications is the Swedish Institute For Transport and Communications Analysis, `http://www.sika-institute.se/`.

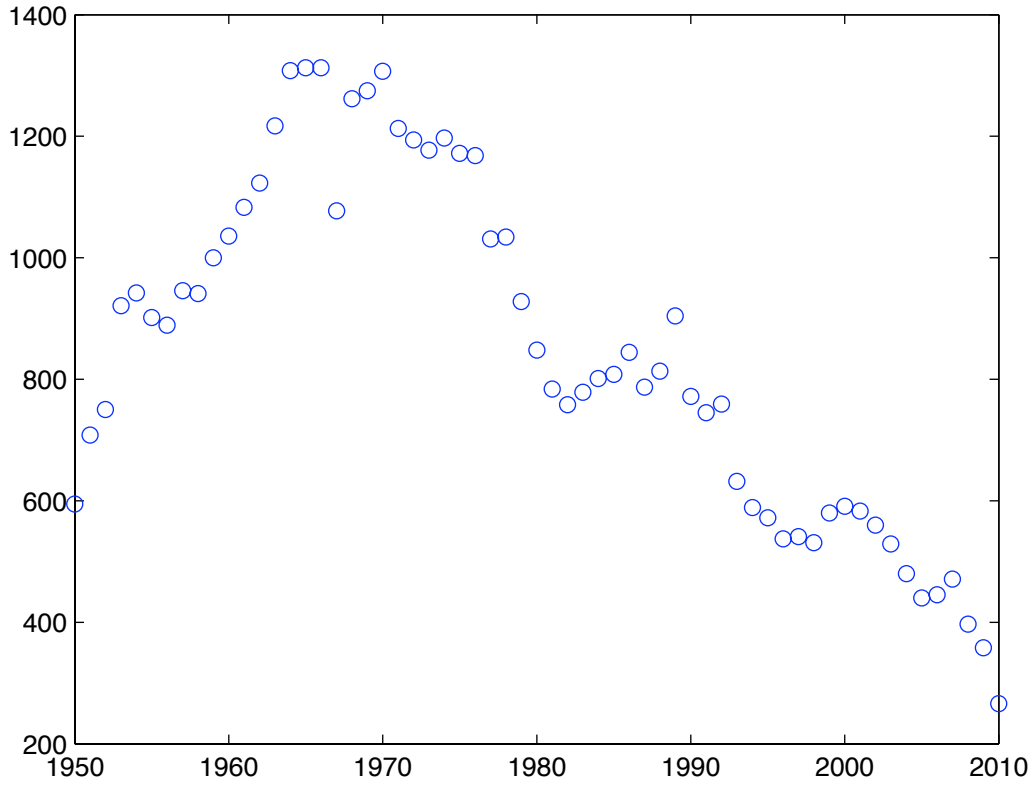

Figure 1: The number of people killed per year in road accidents in Sweden from year 1950 to year 2010. (Source: The Swedish Road Administration.)

The variable data now consists of 9 columns but we are only interested in columns {1, 2, 5, 6}, i.e. {year, number of people killed, number of cars, amount of sold petrol}. We store the data in a structure array

```
>> traffic = struct('year',data(:,1),'killed',data(:,2),'cars',data(:,5),...
             'petrol',data(:,6));
```

Plot the number of people killed each year

```
>> plot(traffic.year, traffic.killed, 'o')
```

Try also plotting the number of people killed vs. number of cars and the petrol consumption. Do you see any connections?

From the plot it can be seen that the trend of increasing number of people killed is broken around year 1965. And from year 1970 the number starts to decrease. Some natural questions arises. Why did the number of people killed increase in years 1950-1965? What was the reason for the brake of the increasing trend? (Hint: right-side driving (1967), front seat-belts in new cars (1969), mandatory use of front seat-belts (1975)).

3 The Poisson regression model

Lets say we have a sequence of count data, n_i, $i = 1, \ldots, k$, for some event, i.e. the number of perished in traffic accidents in a year. This count data is assumed to be observations from random variables $N_i \in \text{Po}(\mu_i)$, (called responses or dependent variables) with mean value $\mu_i = \mu_i(x_{i1}, \ldots, x_{ip})$. The variables, $x_{i1}, \ldots, x_{ip}$, are called explanatory variables[2] and are assumed to measure factors that influence the count data.

We restrict μ_i to be a log-linear function[3],

$$\mu_i = \exp(\beta_0 + \beta_1 x_{i1} + \ldots + \beta_p x_{ip}) \tag{1}$$

And thus the probability that $N_i = n$ is,

$$\mathsf{P}(N_i = n) = \frac{e^{-\mu_i}(\mu_i)^n}{n!} = \frac{e^{-e^{\beta_0+\beta_1 x_{i1}+\ldots+\beta_p x_{ip}}}(e^{\beta_0+\beta_1 x_{i1}+\ldots+\beta_p x_{ip}})^n}{n!},\ n = 0, 1, 2, \ldots. \tag{2}$$

3.1 Estimating model parameters $\beta_0, \ldots, \beta_p$

To simplify the notation we introduce $x_{i0} = 1$ and can now write (1) as,

$$\mathsf{E}[N_i] = \mu_i = \exp\left(\sum_{j=0}^{p} \beta_j x_{ij}\right), \tag{3}$$

where $N_i \in \text{Po}(\mu_i)$ for $i = 1, \ldots, k$.

The likelihood function is calculated as,

$$L(\beta) = \prod_{i=1}^{k} \mathsf{P}(N_i = n_i) = \prod_{i=1}^{k} \frac{\mu_i^{n_i}}{n_i!} e^{-\mu_i}. \tag{4}$$

where $\mu_i = \mu_i(\beta_\mathbf{p})$ is a function of $\beta_\mathbf{p} = (\beta_0, \ldots, \beta_p)$. The ML-estimates $\beta^*_\mathbf{p} = (\beta^*_0, \ldots, \beta^*_p)$ are the values of β that maximize the likelihood function $L(\beta)$. Often it is easier to maximize the log-likelihood function,

$$l(\beta) = -\sum_{i=1}^{k} \log(n_i!) + \sum_{i=1}^{k} n_i \log(\mu_i) - \sum_{i=1}^{k} \mu_i. \tag{5}$$

By setting the first order derivates of the log-likelihood equal to zero, we get a system of $(p+1)$ non-linear equations in β_j,

$$\frac{\partial l(\beta)}{\partial \beta_j} = \sum_{i=1}^{k} \frac{\partial \mu_i}{\partial \beta_j}\left(\frac{n_i}{\mu_i} - 1\right) = \sum_{i=1}^{k} (n_i - \mu_i) x_{ij} = 0,\ j = 0, \ldots, p. \tag{6}$$

Usually, the equation system must be solved with some numerical method, e.g. the Newton-Raphson algorithm, cf. Section 7.3.3. in the textbook. This is also the method implemented in the function `poiss_regress`, which was written for the purpose of this lab and can be found in `labfiles`. Use the command `>> type poiss_regress` to see the code.

Poisson regression model belongs to a class of models called generalized linear models. In a generalized linear model (GLM), the mean of the response, μ, is modeled as a monotonic (nonlinear) transformation

[2] Several other names exist in the literature: independent variables, regressor variables, predictor variables.

[3] Sometimes the model incorporates an extra term t_i: $\mu_i = t_i \exp(\beta_0 + \beta_1 x_{i1} + \ldots + \beta_2 x_{ip})$.

of a linear function of the explanatory variables, $g(\beta_0 + \beta_1 x_1 + \beta_2 x_2 ...)$. The inverse of the transformation function g is called the *canonical link function.* In Poisson regression this function is the log function, but in other GLM's different link functions are used, see `doc glmfit` for a list of supported link functions in the Matlab function `glmfit`[4]. Also, the response may take different distributions, such as the normal or the binomial distribution. Below, we will use related function `glmval` with the logarithmic link function to make predictions from the fitted model, see the code below.

4 Poisson regression of traffic data

We will now try to fit the Poisson regression model to the traffic data of the number of people killed in road accidents. Above, we could see that there was a break in the trend of increasing number people killed around year 1965-1975, mainly because of the improvement in car safety due to the use of safety belts. Because of this it seems reasonable to fit our model to data starting from year 1975.

```
>> traffic = struct('year',data(26:end,1),'killed',data(26:end,2),...
             'cars',data(26:end,5),'petrol',data(26:end,6));
```

Question 1: Which are the explanatory variables? And which is the response?

Redraw the plot from above for the reduced data set

```
>> plot(traffic.year,traffic.killed,'o')
>> figure(1), hold on
```

We start the analysis with one explanatory variable, `traffic.year`. Note usage of the prediction routine for the generalized linear models `glmval`

```
>> X1 = [traffic.year-mean(traffic.year)];
>> n = traffic.killed;
>> beta1 = poiss_regress(X1,n,1e-6);
>> my_fit = glmval(beta1, X1,'log');
>> plot(traffic.year, my_fit, 'b-')
```

Question 2: What is your estimate of β? Convince yourself that this is the solution to (6). You can utilize the following code for this purpose:

```
>> X0=ones(size(X1));
>> X=[X0 , X1];
>> mu=exp(X*beta1);
>> X'*(n-mu)
```

Does it appear to be the solution? Judging from the plot, is this model sufficient to describe the number of people killed in traffic accidents?

Although this simple model seems to capture the overall trend, adding further explanatory variables may improve the fit. Thus, we try adding the number of cars as a variable in our model.

[4] `glmfit` uses a method called weighted least squares to compute the β estimates.

```
>> X2 = [traffic.year-mean(traffic.year), traffic.cars-mean(traffic.cars)];
>> beta2 = poiss_regress(X2,n,1e-6);
>> my_fit = glmval(beta2, X2,'log');
>> plot(traffic.year, my_fit, 'g-')
```

Question 3: Have your estimates β_0^* and β_1^* changed? Does accounting for the number of cars improve the fit?

It seems reasonable also to add the quantity of sold petrol as this would reflect the total mileage of all cars[5].

```
>> X3 = [traffic.year-mean(traffic.year), traffic.cars-mean(traffic.cars),...
        traffic.petrol-mean(traffic.petrol)];
>> beta3 = poiss_regress(X3,n,1e-6);
>> my_fit = glmval(beta3, X3,'log');
>> plot(traffic.year, my_fit, 'r-')
```

Question 4: Have your estimates of β changed now? Use the command `format long` to display more digits. Which model do you choose?

4.1 Model selection - Deviance

It is not always easy to decide, just by looking at the plot, which model to choose. Even though adding more variables improves the fit, it also increases the uncertainty of the estimates. One method to choose complexity of the model is to use the deviance and a hypothesis test.

Let $\beta_{\mathbf{p}}^* = \{\beta_0^*, \beta_1^*, \ldots, \beta_p^*\}$ be the ML-estimates of the model parameters $\{\beta_0, \beta_1, \ldots, \beta_p\}$ of the full model with p explanatory variables and $\beta_{\mathbf{q}}^*$ the estimates of a simpler model where only q ($q < p$) of the explanatory variables have been used. Then for large k, and under suitable regularity conditions, the deviance

$$\text{DEV} = 2 \cdot (l(\beta_{\mathbf{p}}^*) - l((\beta_{\mathbf{q}}^*))) \tag{7}$$

is approximately $\chi^2(p-q)$ distributed if the less complex model is true. Thus, it is possible to test if the simpler model can be rejected compared to the full model.

Question 5: Use `chi2inv` to get the quantiles of the χ^2 distribution. Consider 5% significance level for your test.

The deviance for model 3 compared to model 2 is calculated as

```
>> DEV2 = 2*traffic.killed'*([X0,X3]*beta3-[X0,X2]*beta2)
```

[5] Assuming that the mean fuel consumption of a car has been constant over the years - a 1970 year model of a Volvo used about 10l per 100km which is approximately the same as for the 2000 year model. Of course, the year 2000 model has more than twice the horsepower.

Question 5: Is the improvement with model 3 significant compared to model 2? Repeat the test for model 2 against model 1 and also model 3 against model 1? Which model do you choose? Do you think that there was a sufficient number of explanatory variables used to explain the traffic deaths? Why?

5 Prediction

Now we want to use our model to predict the expected number of perished in traffic accidents six years from now, i.e. year 2016 . In order to do this we first must have an estimate of the number of cars that year. Start by plotting the number of cars vs. year,

```
>> figure(2)
>> plot(traffic.year, traffic.cars, 'o')
>> hold on
```

We will here use a simple linear model for the number of cars, y_i, year x_i

$$y_i = \beta_0 + \beta_1 \cdot x_i + \varepsilon_i \tag{8}$$

where the errors, $\varepsilon_i \in \mathrm{N}(0, (\sigma_\varepsilon)^2)$, are assumed to be independent and identically distributed. This is called a linear regression model. It is possible to estimate the parameters with the maximum likelihood method similar as for the Poisson regression model above.

Question 6: What is the likelihood function? Write it down.

In Matlab, the function `regress` computes the least-squares (LS) estimates of the linear regression model. In the case of ε_i being normally distributed, the LS method is equivalent to the ML method with exactly the same estimates.

```
>> phat = regress(traffic.cars,[ones(length(traffic.cars),1) [1975:2005]'])
>> plot(1975:2016, phat(1)+phat(2)*[1975:2016],'r')
>> cars_2016=phat(1)+phat(2)*2016;
```

Evaluate the fit by looking at the residuals.

```
>> res = traffic.cars-(phat(1)+phat(2)*traffic.year);
>> figure(3), plot(traffic.year,res,'o')
>> figure(4), normplot(res)
```

Question 7: Do the residuals conform to the requirements of the model errors ε_i?

Using the following code provide with prediction of petrol consumption for 2016.

```
>> phat = regress(traffic.petrol,[X0 [1975:2012]' ([1975:2012].^2)'])
>> plot(1975:2016, phat(1)+phat(2)*[1975:2016]+phat(3)*([1975:2016].^2),'r')
>> petrol_2016=phat(1)+phat(2)*2016+phat(3)*2016^2;
```

Notice that this time quadratic model had to be fit to the data.

Question 8: Are you satisfied with the obtained fits for the petrol and the number of cars?

However, for our purpose these rough estimates are sufficient. The expected number of perished can now be predicted using (1),

```
>>x=[1 2016-mean(traffic.year) cars_2016-mean(traffic.cars) petrol_2016-mean(traffic.petrol)]'
>>my_2016=exp(beta3'*x)    %----- Model 3 -----
```

Question 9:
Is the prediction reasonable? Comment.

Modify the code and predict the number of perished year 2013. Is the predicted number close to the "true" number which was 264 perished?