Hw 11

Rafael Espinosa Castañeda

1)

$$\mathcal{L}(\beta) = \log \prod_{i=1}^{N} p(y_i | \beta, x_i)$$

$$\mathcal{L}(\beta) = \sum_{j=1}^{3} \beta_j \sum_{y_i=j} x_i - \sum_{i=1}^{50} \log\left(1 + \sum_{j=1}^{3} \exp(x_i \beta_j)\right)$$

2) $$\frac{\partial \mathcal{L}}{\partial \beta_j} = \sum_{y_i=j} x_i - \sum_{i=1}^{50} \frac{x_i \exp(x_i \beta_j)}{1 + \sum_{j=1}^{3} \exp(x_i \beta_j)}$$

$$\frac{\partial \mathcal{L}}{\partial \beta_j \partial \beta_k} = -1(j=k) \sum_{i=1}^{50} \frac{x_i^2 \exp(x_i \beta_j)}{1 + \sum_{j=1}^{3} \exp(x_i \beta_j)} + \sum_{i=1}^{50} \frac{x_i^2 \exp(x_i (\beta_j + \beta_k))}{1 + \sum_{j=1}^{3} \exp(x_i \beta_j)^2}$$

3) Using R, we obtain

$$\hat{\beta} = (0.9406693, 1.7697685, 2.585144)$$

4) Using R, we obtain the predictive probabilities for y with $x = \bar{x}$

0.1772473, 0.2795248, 0.4375180, 0.1057099

5) So the predicted value is $y = 3$.

Problem 2

1) $S_0(t) = \int_t^\infty f_0(s)\, ds = \int_t^\infty \theta \exp(-s\theta)\, ds$

$$= -\cancel{\theta} \frac{\exp(-s\theta)}{\cancel{\theta}} \Bigg|_t^\infty = -\Big[\exp(-s\theta)\Big]_t^\infty$$

$$= -1 \left(-\exp(-t\theta) \right)$$

$$= \exp(-t\theta) \Rightarrow$$

$$h_0(t) = \frac{\theta \cancel{\exp(-t\theta)}}{\cancel{\exp(-t\theta)}} = \underline{\theta}$$

2) $f(t|x, \theta, \beta) = h(t|x)\, S(t)$; where

$$h(t|x) = h_0(t)\, e^{x\beta} = \theta\, e^{x\beta},$$

$$S(t) = e^{-\theta t}$$

$$f(t|x, \theta, \beta) = \theta\, e^{x\beta} e^{-\theta t}$$

$$= \theta\, e^{x\beta - \theta t}$$

3) $$I(1:N) = \prod_{i=1}^{N} \frac{\theta e^{x_i\beta}}{\sum_{j\geq i} \theta e^{x_j\beta}} = \prod_{i=1}^{N} \frac{e^{x_i\beta}}{\sum_{j\geq i} e^{x_j\beta}}$$

$$\Rightarrow \log\left(I(1:N)\right) = \log\left(\prod_{i=1}^{N} e^{x_i\beta}\right) - \log\left(\sum_{j\geq i} e^{x_j\beta}\right)$$

So, The log-likelihood function is

$$\mathcal{L}(\beta) = \sum_{i=1}^{N} \left[x_i\beta - \log\left(\sum_{j\geq i} e^{x_j\beta}\right)\right]$$

4) $$\frac{d\mathcal{L}}{d\beta} = \sum_{i=1}^{N} x_i - \sum_{i=1}^{N} \frac{\sum_{j\geq i} e^{x_j\beta} x_j}{\sum_{j\geq i} e^{x_j\beta}}$$

$$\frac{d^2\mathcal{L}}{d\beta^2} = -\sum_{i=1}^{N} \frac{\left(\sum_{j\geq i} e^{x_j\beta} x_j^2\right)\sum_{j\geq i} e^{x_j\beta} - \left(\sum_{j\geq i} e^{x_j\beta} x_j\right)^2}{\left(\sum_{j\geq i} e^{x_j\beta}\right)^2}$$

Using iteratively

$$\hat{\beta} = \hat{\beta} - \frac{dL}{d\beta} \Big/ \frac{d^2L}{d\beta^2} \quad \text{in}$$

Python

$$\hat{\beta} = 0{,}614407245791286$$

```python
def beta_hat(beta_0,x):
    x_sum= sum(x)

    beta=beta_0
    for t in range(100):
        total_sum=0
        for  i in range(len(x)):
            sum_num=0
            sum_den=0
            for j in range(i,len(x)):
                num=math.exp(x[j]*beta)*x[j]
                sum_num=sum_num+num
                den=math.exp(x[j]*beta)
                sum_den=sum_den+den
            total_sum=total_sum+ sum_num/sum_den
        derivative= x_sum-total_sum
        sum_factor1=0
        sum_factor2=0
        sum_factor3=0
        second_derivative=0
        for i in range (len(x)):
            sum_factor1=0
            sum_factor2=0
            sum_factor3=0
            for j in range(i,len(x)):
                factor1=math.exp(x[j]*beta)*x[j]**2
                sum_factor1=sum_factor1+factor1
                factor2=math.exp(x[j]*beta)
                sum_factor2=sum_factor2+factor2
                factor3=math.exp(x[j]*beta)*x[j]
                sum_factor3=sum_factor3+factor3
            numerator=sum_factor1*sum_factor2-sum_factor3**2
            denominator=sum_factor2**2
            second_derivative=second_derivative-numerator/denominator
        beta=beta-derivative/second_derivative
    return beta
```

Calculation First derivative

Calculation second derivative

updating $\hat{\beta}$ value.

```
# DataFrame with already sorted values( sorted
# in the excel file)
df=pd.read_csv("data2.csv")

x=df["x"]
b_0=0
beta=beta_hat(b_0,x)
print(beta)
```

5)

```
beta_samples=[]
for samples in range(100):
    t=[]


    for i in range(len(x)):
        t.append(0)

    for i in range(len(x)):
        t[i]=np.random.exponential(scale=1/math.exp(beta*x[i]))
        #t[i]=np.random.exponential()

    df=pd.DataFrame({"t":t,"x":x})

    df2=df.sort_values(by=["t"])
    x_sample=df2["x"]

    beta_samples.append(beta_hat(beta,x_sample))

print(beta_samples)
variance=np.var(beta_samples)
print(variance)
```

Created 100 samples for the as

$t_i = \theta \exp(x_i \beta)$, where $\theta = 1$

(rate $= \frac{1}{\text{scale}}$ in Python uses scale instead of rate)

After creating the samples and obtaining 100 estimates for $\hat{\beta}$,

we calculate the variance.

$$Var(\hat{\beta}) = 3{,}636 \times 10^{-32}$$

So

$$\beta \approx \text{Normal}\left(\hat{\beta}, \; Var(\hat{\beta})\right)$$

where

$$\hat{\beta} = 0{,}6144072457912 86$$