Project1

Erle E. Sandø, Simon Liabø

2022-09-23

Problem 1

a)

The pdf for a Poisson distribution is $f(x) = \frac{e^{-\lambda_i}\lambda_i^{y_i}}{y_i!}$ where the expected value λ is given by the cononical link function $\lambda_i = e^{\eta_i}$ and opposite $\eta_i = ln(\lambda_i)$.

To find the log likelihood function we first need the likelihood function $L(\beta)$.

$$L(\beta) = \prod_{i=1}^{n} \frac{e^{-\lambda_i}\lambda_i^{y_i}}{y_i!}$$

.

Log likelihood function:

$$\begin{aligned} l(\beta) &= \ln(\prod_{i=1}^{n} \frac{e^{-\lambda_i}\lambda_i^{y_i}}{y_i!}) \\ &= \sum_{i=1}^{n} \ln(\frac{e^{-\lambda_i}\lambda_i^{y_i}}{y_i!}) \\ &= \sum_{i=1}^{n} \ln e^{-\lambda_i} + \ln \lambda_i^{y_i} - \ln y_i! \\ &= \sum_{i=1}^{n} -\lambda_i + y_i \ln \lambda_i - \ln y_i! \\ &= \sum_{i=1}^{n} -e^{x_i^T\beta} + y_i x_i^T \beta - \ln y_i! \end{aligned} \tag{1}$$

Fisher score function:

$$\begin{aligned}
s(\beta) &= \frac{\partial l}{\partial \beta} \\
&= \frac{\partial}{\partial \beta}(\sum_{i=1}^{n} -e^{x_i^T \beta} + y_i x_i^T \beta - \ln y_i!) \\
&= \sum_{i=1}^{n} \frac{\partial}{\partial \beta}(-e^{x_i^T \beta} + y_i x_i^T \beta - \ln y_i!) \\
&= \sum_{i=1}^{n} -x_i e^{x_i^T \beta} + y_i x_i \\
&= \sum_{i=1}^{n} x_i (y_i - e^{x_i^T \beta})
\end{aligned} \tag{2}$$

Observed Fisher information: # noe feil her som gjør at koden ikke kjøres

$$\begin{aligned}
H(\beta) &= -\frac{\partial^2 l(\beta)}{\partial \beta \partial \beta^T} \\
&= -\frac{\partial s(\beta)}{\partial \beta^T} \\
&= -\frac{\partial}{\partial \beta^T}(\sum i = 1^n (x_i (y_i - e^{\eta_i}))) \\
&= \sum i = 1^n (\frac{\partial}{\partial \beta^T}(x_i (y_i - e^{\eta_i}))) \\
&= \sum i = 1^n (x_i \frac{\partial}{\partial \beta^T} e^{\eta_i}) \\
&= \sum i = 1^n (x_i \frac{\partial \eta_i}{\partial \beta^T} e^{\eta_i}) \\
&= \sum i = 1^n (x_i x_i^T \lambda_i)
\end{aligned} \tag{3}$$

Expected Fisher information:

$$\begin{aligned}
F(\beta) &= E[s_i(\beta) \cdot s_i^T(\beta)] \\
&= E[(y_i - \lambda_i) x_i \cdot (y_i - \lambda_i) x_i^T)] \\
&= E[x_i x_i^T (y_i - \lambda_i)^2] \\
&= x_i x_i^T E[(y_i - \lambda_i)^2] \\
&= x_i x_i^T \cdot Var y_i \\
&= x_i x_i^T \lambda_i
\end{aligned} \tag{4}$$

b)

```
score = function(y, X, beta)
  {
  eta = as.vector(X %*% beta)
  lmdba = exp(eta)
  score = apply((y - lmdba) * X,2,sum)
  score
}
expected_fisher = function(X, beta)
  {
  eta = as.vector(X %*% beta)
  W = diag(exp(eta))
  t(X) %*% W %*% X
}
log_likelihood = function(y, X, beta, lmbda = exp(as.vector(X %*% beta)))
  {
  sum(ifelse(lmbda==0,0,y*log(lmbda) ) - lmbda)
}
myglm = function(formula, data, start=rep(0, ncol(model.matrix(formula, data))))
  {
  X = model.matrix(formula, data)
  response = as.character(formula)[2]
  y = data[[response]]
  beta = start

  s=1
  while (s > (1e-10)) {
    eta = as.vector(X %*% beta)
    lmbda = exp(eta)

    score_val = score(y, X, beta)
    f = expected_fisher(X, beta)

    beta = beta + solve(f) %*% score_val
    s = sum(score_val^2)
    }
  #vcov
  cov_mat = solve(f)

  #coefficients
  sd_err = sqrt(diag(cov_mat))
  coeff = cbind(beta, sd_err)
  colnames(coeff) = c("Estimate", "Std.Error")
  rownames(coeff) = paste0("beta_", seq_along(beta)-1)

  #deviance
  dev = 2 * (log_likelihood(y, X, beta) - log_likelihood(y, X, beta, lmbda = y))

  list(coefficients = coeff, deviance = dev, vcov = cov_mat)
}
```

c)

```
n = 1000
k = 2
#simulate data
beta = rnorm(k+1)
X = cbind(matrix(1,n),matrix(rnorm(n * k), nrow = n, ncol = k))
eta = as.vector(X %*% beta)
```

```
lmd = exp(eta)
y = rpois(n,lmd)
data_sim = as.data.frame(cbind(y,X[,2:3]))
#fit models
model_myglm = myglm(y~., data = data_sim)
model_glm = glm(y~., data = data_sim, family = poisson(link=log))
#evaluate
coeff_diff = mean((model_myglm$coefficients[,1] -model_glm$coefficients)^2)
coeff_diff
```

```
## [1] 3.370121e-30
```

```
vcov_diff = mean( (model_myglm$vcov - vcov(model_glm))^2)
vcov_diff
```

```
## [1] 2.148087e-22
```

The model looks good. The results are very close to those obtained with glm() and vcov().

Problem 2

```
load(url("https://www.math.ntnu.no/emner/TMA4315/2022h/hoge-veluwe.Rdata"))
```

In problem 2 we will consider a data frame containing data on the bird Great tit in the national park of Hoge Veluwe. The data was collected on 135 female birds in the summer of 2005.

The response variable is the number of fledglings leaving the nest, which relies on the time of initiate breeding and the number of fledglings for each bird, plus the timing of food resources. The number of fledglings follow a poisson distribution with expectation $\lambda_i(t_i)$. This dependence is explained by a gaussian function

$$\lambda_0 \exp(-\frac{(t_i-\theta)^2}{2\omega^2})$$

.

a)

In the expression above λ_0 is the number of fledglings when it is the highest. Θ is the mean time, E(t), that is the time when there is the most fledglings. (?) ω represents how much variance there is in the number of fledglings.

b)

A generalized linear model needs a random component, a systematic component and a link function which can give the relations between the GLM parameters contained in β and $(\lambda_0, \theta, \omega)$.

In this situation the random component is y_i which as said follows a poisson distribution. The systematic component is $\eta_i = t_i^T\beta$, and the relation can be explained by $\eta_i = \ln(\lambda_i)$ which is a canonical link function.

The link function gives the relation between β and $(\lambda_0, \theta, \omega)$:

$$\eta_i = \beta_0 + \beta_1 t_i + \beta_2 t_i^2 = \log(\lambda_i) = \log(\lambda_0 + \exp(-\frac{(t_i-\theta)^2}{2\omega^2})) = \log(\lambda_0) - \frac{(t_i-\theta)^2}{2\omega^2}$$