Math 523 - Assignment 2

Emir Sevinc 260682995

February 8, 2019

A4

a)

Since we're using the log link, we will have that $g(\mu_i) = \eta_i$. Since $g(x) = log(x)$ and $\eta_i = X_i\beta$ we have $log(E[Y/X]) = X\beta \implies E[Y/X] = exp(X_i\beta)$, and the model will look like this:

$log(E[Y/X]) = X\beta = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_{L-1}X_{L-1} \implies E[Y/X] = exp[\beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_{L-1}X_{L-1}]$, where L-1 additional parameters to the β_0 are present so we have L parameters in total, and the predictor is such that X_i is 1 if X belongs to level i, and 0 else.

b)

Let's code X as X = 0 if it belongs to group A, and 1 if it belongs to group B, so we're using group A as the reference group. The model now has L = 2 so a single slope parameter, so it looks like this: $\log[E(Y/X) = \beta_0 + \beta_1 X]$, and the design matrix X is as follows:

$$\begin{array}{cc} \text{X} & \begin{array}{cc} 1 & X \end{array} \\ \begin{array}{c} \\ \\ \\ na \\ \\ \\ na+nb \end{array} & \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ . & 0 \\ . & . \\ . & 1 \\ 1 & 1 \\ 1 & 1 \end{pmatrix} \end{array}$$

That is, since the groups are ordered, the second columns consists of zeroes until the n_A'th row, and ones from that row to the $n = n_A + n_B$'th row. We have that for the negative binomial model, $E(Y_i) = \mu_i = \mu$, $Var[Y_i] = \frac{\mu_I(\mu_i+\theta_z)}{\theta_z}$ (From assignment 1). These will be used later.

Since we're using the log link, we have $g(x) = log(x) \implies g'(x) = \frac{1}{x}$. Now we set up our likelihood:

$l(\beta) = \sum_{i=1}^{n} \frac{y_i\theta_i - b(\theta_i)}{a(\phi)} + c(\theta_i, y_i)$. We have from the previous assignment that the negative binomial model can be written in exponential family form as
$exp[ylog(\frac{\mu}{\mu+\theta_z}) + \theta_z log(1 - \frac{\mu}{\mu+\theta_z}) + log(\frac{\Gamma(y+\theta_z)}{\Gamma(y+1)\Gamma(\theta_z)})]$ where $\theta_i = log(\frac{\mu_i}{\mu_i+\theta_z})$, $b(\theta_i) = -\theta_z log(\frac{\theta_z}{\mu_I+\theta_z})$ and $a(\phi) = 1$. So our likelihood will look like:

$l(\beta) = \sum_{i=1}^{n}[y_i log(\frac{\mu_i}{\mu_i+\theta_z}) + \theta_z log(1 - \frac{\mu_i}{\mu_i+\theta_z}) + log(\frac{\Gamma(y+\theta_z)}{\Gamma(y+1)\Gamma(\theta_z)})]$
. We know from class that a general score equation can be written as follows:

$\frac{\partial l}{\partial \beta_j} = \sum_{i=1}^{n} \frac{y_i-\mu_i}{Var(Y_i)} * \frac{1}{g'(\mu_i)} * X_{ij}$. Plugging in our known values and using $g'(\mu_i) = \frac{1}{\mu_i}$ we get:
$\frac{\partial l}{\partial \beta_j} = \sum_{i=1}^{n} \frac{(y_i-\mu_i)\theta_z}{\mu_i(\mu_i+\theta_z)} * \mu_i * X_{ij}$
$\frac{\partial l}{\partial \beta_j} = \sum_{i=1}^{n} \frac{(y_i-\mu_i)\theta_z}{(\mu_i+\theta_z)} * X_{ij}$

To optimise these will be set to 0, so we have our score equations as:
$\frac{\partial l}{\partial \beta_0} = \sum_{i=1}^{n} \frac{(y_i-\mu_i)\theta_z}{(\mu_i+\theta_z)} * X_{io}$. But We know that X_{i0} is corresponding to the intercept so the design matrix has all 1's in that column, thus this is equal to

$\sum_{i=1}^{n} \frac{(y_i-\mu_i)\theta_z}{(\mu_i+\theta_z)} = 0 \implies \sum_{i=1}^{n} \frac{(y_i-\mu_i)}{(\mu_i+\theta_z)} = 0$

We also have

$\frac{\partial l}{\partial \beta_1} = \sum_{i=1}^{n} \frac{(y_i-\mu_i)\theta_z}{\mu_i(\mu_i+\theta_z)} * \mu_i * X_{i1} = 0 \implies \sum_{i=1}^{n} \frac{(y_i-\mu_i)}{\mu_i(\mu_i+\theta_z)}\mu_i * X_{i1} = 0$. We can partition the sum as follows: $\sum_{i=1}^{n} \frac{(y_i-\mu_i)}{\mu_i(\mu_i+\theta_z)}\mu_i * X_{i1} = \sum_{i=1}^{n_A} \frac{(y_i-\mu_i)}{\mu_i(\mu_i+\theta_z)}\mu_i * X_{i1} + \sum_{i=n_{A+1}}^{n_A+n_B} \frac{(y_i-\mu_i)}{\mu_i(\mu_i+\theta_z)}\mu_i * X_{i1}$ since $n_A + n_B = n$. But as we can see from the design matrix, the column corresponding to the X_{i1}s is zero upto n_A, and 1 after that, so this simply becomes

$\sum_{i=1}^{n_A} \frac{(y_i-\mu_i)}{\mu_i(\mu_i+\theta_z)}\mu_i * 0 + \sum_{i=n_{A+1}}^{n_A+n_B} \frac{(y_i-\mu_i)}{\mu_i(\mu_i+\theta_z)}\mu_i * 1$
$= \sum_{i=n_{A+1}}^{n} \frac{(y_i-\mu_i)}{(\mu_i+\theta_z)}$

So our score equations are:

$\sum_{i=1}^{n} \frac{(y_i-\mu_i)}{(\mu_i+\theta_z)} = 0$
$\sum_{i=n_{A+1}}^{n} \frac{(y_i-\mu_i)}{(\mu_i+\theta_z)} = 0$

c)

For equation 1 Partitioning once again we get $\sum_{i=1}^{n} \frac{(y_i-\mu_i)}{(\mu_i+\theta_z)} = \sum_{i=1}^{n_A} \frac{(y_i-\mu_i)}{(\mu_i+\theta_z)} + \sum_{i=n_{A+1}}^{n_A+n_B} \frac{(y_i-\mu_i)}{(\mu_i+\theta_z)}$, but by the second score equation, the second term is zero (as $n = n_A + n_B$), so we must have $\sum_{i=1}^{n_A} \frac{(y_i-\mu_i)}{(\mu_i+\theta_z)} = 0$. Since our model has the form $\mu_i = exp(\beta_0 + \beta_1 X_{i1})$ we can plug this in the score equations, solve and simplify using the fact that $X_i 1 = 0$ for $1 \leq i \leq n_A$ and 1 else. So we get

$\sum_{i=1}^{n_A} \frac{y_i - e^{\beta_0+\beta_1 X_{i1}}}{(e^{\beta_0+\beta_1 X_{i1}}+\theta_z)} = 0 \implies$

$\sum_{i=1}^{n_A} \frac{y_i - e^{\beta_0}}{(e^{\beta_0}+\theta_z)} = 0 \implies$

$\frac{1}{(e^{\beta_0}+\theta_z)} \sum_{i=1}^{n_A} y_i - e^{\beta_0} = 0 \implies$

$\sum_{i=1}^{n_A} y_i - e^{\beta_0} = 0 \implies$

$\sum_{i=1}^{n_A} y_i = \sum_{i=1}^{n_A} e^{\beta_0} \implies$

$\sum_{i=1}^{n_A} y_i = n_A e^{\beta_0} \implies$

$e^{\beta_0} = \frac{1}{n_A} \sum_{i=1}^{n_A} y_i \implies$

$\hat{\beta}_0 = log(\frac{1}{n_A} \sum_{i=1}^{n_A} y_i)$ And for equation 2,

$0 = \sum_{i=n_{A+1}}^{n} \frac{y_i - e^{\beta_0+\beta_1 X_{i1}}}{(e^{\beta_0+\beta_1 X_{i1}}+\theta_z)} \implies$

$0 = \sum_{i=n_{A+1}}^{n} \frac{y_i - e^{\beta_0+\beta_1}}{e^{\beta_0+\beta_1}+\theta_z} \implies$

$0 = \frac{1}{e^{\beta_0+\beta_1}+\theta_z} \sum_{i=n_{A+1}}^{n} y_i - e^{\beta_0+\beta_1} \implies$

$\sum_{i=n_{A+1}}^{n} y_i - e^{\beta_0+\beta_1} = 0 \implies$

$\sum_{i=n_{A+1}}^{n} y_i - \sum_{i=n_{A+1}}^{n} e^{\beta_0+\beta_1} = 0 \implies$

$\sum_{i=n_{A+1}}^{n} y_i = \sum_{i=n_{A+1}}^{n} e^{\beta_0+\beta_1} \implies$

$\sum_{i=n_{A+1}}^{n} y_i = n_B e^{\beta_0+\beta_1} = n_B e^{\beta_0} e^{\beta_1} \implies$

$\sum_{i=n_{A+1}}^{n} y_i = n_B e^{log(\frac{1}{n_A}\sum_{i=1}^{n_A} y_i)} e^{\beta_1} \implies$

$\sum_{i=n_{A+1}}^{n} y_i = (\frac{n_B}{n_A}\sum_{i=1}^{n_A} y_i) e_1^{\beta} \implies$

$e^{\beta_1} = \frac{n_A}{n_B}\frac{\sum_{i=n_{A+1}}^{n} y_i}{\sum_{i=1}^{n_A} y_i} \implies$

$\beta_1 = log(\frac{n_A}{n_B}\frac{\sum_{i=n_{A+1}}^{n} y_i}{\sum_{i=1}^{n_A} y_i}) \implies$

$\hat{\beta}_1 = log(\frac{1}{n_B}\sum_{i=n_{A+1}}^{n} y_i) - log(\frac{1}{n_A}\sum_{i=1}^{n_A} y_i)$. Note that the term getting subtracted is equal to our estimate of β_0 so
$\hat{\beta}_1 = log(\frac{1}{n_B}\sum_{i=n_{A+1}}^{n} y_i) - \hat{\beta}_0$

So our parameter estimates are:
$\hat{\beta}_0 = log(\frac{1}{n_A}\sum_{i=1}^{n_A} y_i)$
$\hat{\beta}_1 = log(\frac{1}{n_B}\sum_{i=n_{A+1}}^{n} y_i) - \hat{\beta}_0$

the fitted means will be $\hat{\mu}_i = e^{\hat{\beta}_0+\hat{\beta}_1 X_{i1}}$ in general, but since the value of X_{i1} is 0 for $i = 1, 2, ..., n_A$ and 1 for $i = n_{A+1}, ..., n$ the fitted means will be:

$\hat{\mu}_i = e^{\hat{\beta}_0} = \frac{1}{n_A}\sum_{i=1}^{n_A} y_i$ for $i = 1, ..., n_A$ and
$\hat{\mu}_i = e^{\hat{\beta}_0+\hat{\beta}_1} = e^{\hat{\beta}_0 + log(\frac{1}{n_B}\sum_{i=n_{A+1}}^{n} y_i) - \hat{\beta}_0}$
$= \frac{1}{n_B}\sum_{i=n_{A+1}}^{n} y_i$ for $i = n_{A+1}, ..., n$

If the canonical links were used, the score equations would be equal to
$\sum_{i=1}^{n} y_i X_{ij} = \sum_{i=1}^{n} \mu_i X_{ij}$. So we would get
$\sum_{i=1}^{n} y_i X_{i0} = \sum_{i=1}^{n} \mu_i X_{i0} \implies \sum_{i=1}^{n} y_i = \sum_{i=1}^{n} \mu_i$ and
$\sum_{i=1}^{n} y_i X_{i1} = \sum_{i=1}^{n} \mu_i X_{i1} \implies \sum_{i=n_{A+1}}^{n} y_i = \sum_{i=n_{A+1}}^{n} \mu_i$ as our score equations. Partitioning the sums for eq1,

$\sum_{i=1}^{n_A} y_i + \sum_{i=n_{A+1}}^{n} y_i = \sum_{i=1}^{n_A} \mu_i + \sum_{i=n_{A+1}}^{n} \mu_i$, but since both of the added terms are equal by the second score equation we must have
$\sum_{i=1}^{n_A} y_i = \sum_{i=1}^{n_A} \mu_i$
. The canonical link being used implies that $log(\frac{\mu_i}{\mu_i+\theta_z}) = X_i\beta$ and since we know the inverse of the canonical from assignment 1 (via the b') to be $\frac{\theta_z e^{\theta}}{1-e^{\theta}}$ we must have, using the $X\beta_i$ we know from earlier, $\mu_i = \frac{\theta_z e^{\beta_0+\beta_1 X_{i1}}}{1-e^{\beta_0+\beta_1 X_{i1}}}$
.

Plugging these in the score equations and using the known values of X_{i1} for our intervals we get
$\sum_{i=1}^{n_A} y_i = \sum_{i=1}^{n_A} \frac{\theta_z e^{\beta_0+\beta_1 X_{i1}}}{1-e^{\beta_0+\beta_1 X_{i1}}} \implies$
$\sum_{i=1}^{n_A} y_i = \sum_{i=1}^{n_A} \frac{\theta_z e^{\beta_0}}{1-e^{\beta_0}} = n_A \frac{\theta_z e^{\beta_0}}{1-e^{\beta_0}}$
and
$\sum_{i=n_{A+1}}^{n} y_i = \sum_{i=n_{A+1}}^{n} \frac{\theta_z e^{\beta_0+\beta_1}}{1-e^{\beta_0+\beta_1}} \implies$
$\sum_{i=n_{A+1}}^{n} y_i = n_B \frac{\theta_z e^{\beta_0+\beta_1}}{1-e^{\beta_0+\beta_1}}$
From
$\sum_{i=1}^{n_A} y_i = n_A \frac{\theta_z e^{\beta_0}}{1-e^{\beta_0}}$ we will have $(1-e^{\beta_0})\sum_{i=1}^{n_A} y_i = n_A\theta_z e^{\beta_0} \implies e^{\beta_0}[n_A\theta_z + \sum_{i=1}^{n_A} y_i] = \sum_{i=1}^{n_A} y_i \implies$
$e^{\beta_0} = \frac{\sum_{i=1}^{n_A} y_i}{n_A\theta_z + \sum_{i=1}^{n_A} y_i} \implies \hat{\beta}_0 = log(\frac{\sum_{i=1}^{n_A} y_i}{n_A\theta_z + \sum_{i=1}^{n_A} y_i})$

And from $\sum_{i=n_{A+1}}^{n} y_i = n_B \frac{\theta_z e^{\beta_0+\beta_1}}{1-e^{\beta_0+\beta_1}}$ we will have

$n_A\theta_z e^{\beta_0+\beta_1} = (\sum_{i=n_{A+1}}^{n} y_i)(1-e^{\beta_0+\beta_1}) \implies [n_B\theta_z + \sum_{i=n_{A+1}}^{n} y_i]e^{\beta_0}e^{\beta_1} = \sum_{i=n_{A+1}}^{n} y_i$. From this we can isolate β_1

:

$e_1^{\beta} = \frac{\sum_{i=n_{A+1}}^{n} y_i}{(n_B\theta_z+\sum_{i=n_{A+1}}^{n} y_i)/\beta_0} \implies \beta_1 = log((\frac{\sum_{i=n_{A+1}}^{n} y_i}{n_B\theta_z+\sum_{i=n_{A+1}}^{n} y_i})/\frac{\sum_{i=1}^{n_A} y_i}{n_A\theta_z+\sum_{i=1}^{n_A} y_i})$

$= log(\frac{\sum_{i=n_{A+1}}^{n} y_i}{n_B\theta_z+\sum_{i=n_{A+1}}^{n} y_i}) - log(\frac{\sum_{i=1}^{n_A} y_i}{n_A\theta_z+\sum_{i=1}^{n_A} y_i})$

$= log(\frac{\sum_{i=n_{A+1}}^{n} y_i}{n_B\theta_z+\sum_{i=n_{A+1}}^{n} y_i}) - \hat{\beta}_0$

So our parameter estimates this time are: $\hat{\beta}_0 = log(\frac{\sum_{i=1}^{n_A} y_i}{n_A\theta_z+\sum_{i=1}^{n_A} y_i})$

$\hat{\beta}_1 = log(\frac{\sum_{i=n_{A+1}}^{n} y_i}{n_B\theta_z+\sum_{i=n_{A+1}}^{n} y_i}) - \hat{\beta}_0$

Since our means are now $\mu_i = \frac{\theta_z e^{\beta_0+\beta_1}}{1-e^{\beta_0+\beta_1}}$, using our known X_{i1} values we get:

$\hat{\mu}_i = \frac{\theta_z e^{\hat{\beta}_0}}{1-e^{\hat{\beta}_0}} = \theta_z \frac{\sum_{i=1}^{n_A} y_i}{n_A\theta_z+\sum_{i=1}^{n_A} y_i} / \frac{n_A\theta_z}{n_A\theta_z+\sum_{i=1}^{n_A} y_i} = \frac{\theta_z \sum_{i=1}^{n_A} y_i}{n_A\theta_z} = \frac{\sum_{i=1}^{n_A} y_i}{n_A}$ for i = 1, ..., n_A and

$\hat{\mu}_i = \frac{\theta_z e^{\hat{\beta}_0+\hat{\beta}_1}}{1-e^{\hat{\beta}_0+\hat{\beta}_1}} = \theta_z e^{\hat{\beta}_0+log(\frac{\sum_{i=n_{A+1}}^{n} y_i}{n_B\theta_z+\sum_{i=n_{A+1}}^{n} y_i})-\hat{\beta}_0} / (\frac{n_B\theta_z}{n_B\theta_z+\sum_{i=n_{A+1}}^{n} y_i}) = \frac{\sum_{i=n_{A+1}}^{n} y_i}{n_B}$ for i = $n_{A+1}, ..., n$

It is noteworthy that the parameter estimates are different, but we got the same fitted means.

d)

We have that our design matrix (nx2) is:

$$\begin{array}{ll} X & \begin{array}{cc} 1 & X \end{array} \\ \\ \\ \\ na & \\ \\ \\ na+nb & \end{array} \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ . & 0 \\ . & . \\ . & 1 \\ 1 & 1 \\ 1 & 1 \end{pmatrix}$$

The end result of this computation will make use of the terms W_{ii} but we dont have to know them until the end, so we will only note that W is an nxn matrix with $W = diag[w_1, w_2, ..., w_n]$

First we need to multiply X^T(2xn) with W(nxn) where X^T :

$$\begin{array}{c} \text{Xtr} \\ \begin{array}{ccccccc} & & & n_A & n_{A+1} & & n \end{array} \\ \begin{pmatrix} 1 & 1 & ... & 1 & 1 & ... & 1 \\ 0 & 0 & ... & 0 & 1 & ... & 1 \end{pmatrix} \end{array}$$

Multiplying this out with the W matrix guves:

$$\begin{array}{c} \text{Xtr} \\ \begin{array}{ccccccc} & & & n_A & n_{A+1} & & n \end{array} \\ \begin{pmatrix} w_{11} & w_{22} & ... & w_{n_A n_A} & w_{n_{A+1} n_{A+1}} & ... & w_{nn} \\ 0 & 0 & ... & 0 & w_{n_{A+1} n_{A+1}} & ... & w_{nn} \end{pmatrix} \end{array}$$

Now we multiply this with our nx2 design matrix to get:

$$\text{F} \quad \begin{pmatrix} \sum_{i=1}^{n} w_{ii} & \sum_{i=n_{A+1}}^{n} w_{ii} \\ \sum_{i=n_{A+1}}^{n} w_{ii} & \sum_{i=n_{A+1}}^{n} w_{ii} \end{pmatrix}$$

So the entries of W will only change according to the link function used. In general, we have that $w_{ii} = (\frac{\partial \mu_i}{\partial \eta_i})^2 \frac{1}{Var(Y_i)}$

Since $g(\mu_i) = \eta_i$, $\mu_i = g^{-1}(\eta_i)$. So $\frac{\partial \mu_i}{\partial \eta_i} = \frac{1}{g'(\mu_i)}$ as we have seen in class, and we know from assignment 1 that $Var[Y_i] = \frac{\mu_i(\mu_i+\theta_z)}{\theta_z}$

So $w_{ii} = \frac{1}{(g'(\mu_i))^2} \frac{\theta_z}{\mu_i(\mu_i+\theta_z)}$

If we are using the log link, then $g(\mu) = log(\mu) \implies g'(\mu) = \frac{1}{\mu} \implies \frac{1}{(g'(\mu))^2} = \mu^2$

Thus $w_{ii} = \frac{\mu_i \theta_z}{\mu_i+\theta_z}$ so the Fisher Information matrix is:

$$\text{F} \quad \begin{pmatrix} \sum_{i=1}^{n} \frac{\mu_i\theta_z}{\mu_i+\theta_z} & \sum_{i=n_{A+1}}^{n} \frac{\mu_i\theta_z}{\mu_i+\theta_z} \\ \sum_{i=n_{A+1}}^{n} \frac{\mu_i\theta_z}{\mu_i+\theta_z} & \sum_{i=n_{A+1}}^{n} \frac{\mu_i\theta_z}{\mu_i+\theta_z} \end{pmatrix}$$

If we use the canonical link, $g = log(\frac{\mu}{\mu+\theta_z}) \implies g' = \frac{\mu+\theta_z}{\mu} * \frac{\theta_z}{(\mu+\theta_z)^2} = \frac{\theta_z}{\mu(\mu+\theta_z)} \implies \frac{1}{(g')^2} = (\frac{\mu(\mu+\theta_z)}{\theta_z})^2 = (Var[Y_i])^2$

Thus $w_{ii} = (Var[Y_i])^2 \frac{1}{Var[Y_i]} = Var[Y_i] = \frac{\mu_i(\mu_i+\theta_z)}{\theta_z}$ so our Fisher Information matrix is now:

$$\text{F} \quad \begin{pmatrix} \sum_{i=1}^{n} \frac{\mu_i(\mu_i+\theta_z)}{\theta_z} & \sum_{i=n_{A+1}}^{n} \frac{\mu_i(\mu_i+\theta_z)}{\theta_z} \\ \sum_{i=n_{A+1}}^{n} \frac{\mu_i(\mu_i+\theta_z)}{\theta_z} & \sum_{i=n_{A+1}}^{n} \frac{\mu_i(\mu_i+\theta_z)}{\theta_z} \end{pmatrix}$$

Note that these are different.

e)

We will need to differentiate the score with respect to β to get the Hessian.

$\frac{\partial l}{\partial \beta_j} = \sum_{i=1}^{n} \frac{(y_i-\mu_i)\theta_z}{(\mu_i+\theta_z)} * X_{ij}$ This is a function of μ which is a function of η and η is a function of β, so we need: $\frac{\partial l^2}{\partial \beta_j \partial \beta_k} = \frac{\partial}{\partial \mu_i}[\sum_{i=1}^{n} \frac{(y_i-\mu_i)\theta_z}{(\mu_i+\theta_z)} * X_{ij}] * \frac{\partial \mu_i}{\partial \eta_i} * \frac{\partial \eta_i}{\partial \beta_k}$

We know that $\frac{\partial \mu_i}{\partial \eta_i} = \frac{1}{g'(\mu_i)} = \mu_i$ since we're using $g(x) = log(x)$, and $\frac{\partial \eta_i}{\partial \beta_k} = X_{ik}$

Now $\frac{\partial}{\partial \mu_i}[\sum_{i=1}^{n} \frac{(y_i-\mu_i)\theta_z}{(\mu_i+\theta_z)} * X_{ij}] = X_{ij}\theta_z \sum_{i=1}^{n} \frac{\partial}{\partial \mu_i} \frac{(y_i-\mu_i)}{(\mu_i+\theta_z)}$

$\frac{\partial}{\partial \mu_i}\frac{(y_i-\mu_i)}{(\mu_i+\theta_z)} = \frac{-1*(\mu_i+\theta_z)-(y_i-\mu_i)}{(\mu_i+\theta_z)^2} = \frac{-y_i-\theta_z}{(\mu_i+\theta_z)^2}$
So $\frac{\partial l^2}{\partial\beta_j\partial\beta_k} = \sum_{i=1}^n X_{ij}\theta_z\frac{-y_i-\theta_z}{(\mu_i+\theta_z)^2}X_{ik}$. Taking the minus of this, we get that the observed information model is

$$F$$
$$\begin{pmatrix} \sum_{i=1}^n \theta_z\frac{y_i+\theta_z}{(\mu_i+\theta_z)^2}\mu_i & \sum_{i=n_{A+1}}^n \theta_z\frac{y_i+\theta_z}{(\mu_i+\theta_z)^2}\mu_i \\ \sum_{i=n_{A+1}}^n \theta_z\frac{y_i+\theta_z}{(\mu_i+\theta_z)^2}\mu_i & \sum_{i=n_{A+1}}^n \theta_z\frac{y_i+\theta_z}{(\mu_i+\theta_z)^2}\mu_i \end{pmatrix}$$

This is not the same as the Fisher Info matrix because we didn't use the canonical link.

f)

Asymptotic variance is given by the diagonal elements of the inverse of the Fisher information matrix. We had that when the log link was used the Fisher Info was:

$$F$$
$$\begin{pmatrix} \sum_{i=1}^n \frac{\mu_i\theta_z}{\mu_i+\theta_z} & \sum_{i=n_{A+1}}^n \frac{\mu_i\theta_z}{\mu_i+\theta_z} \\ \sum_{i=n_{A+1}}^n \frac{\mu_i\theta_z}{\mu_i+\theta_z} & \sum_{i=n_{A+1}}^n \frac{\mu_i\theta_z}{\mu_i+\theta_z} \end{pmatrix}$$

This is of the form:

$$M$$
$$\begin{pmatrix} M & N \\ M & N \end{pmatrix}$$

So its invere would look like:

$$Minv$$
$$\begin{pmatrix} \frac{N}{MN-N^2} & \frac{-N}{MN-N^2} \\ \frac{-N}{MN-N^2} & \frac{M}{MN-N^2} \end{pmatrix}$$

So we can easily get the diagonal entires as $\frac{N}{MN-N^2}$ and $\frac{M}{MN-N^2}$. Thus we will have $Var[\hat{\beta}_0] = \frac{\sum_{i=n_{A+1}}^n \frac{\mu_i\theta_z}{\mu_i+\theta_z}}{\sum_{i=1}^n \frac{\mu_i\theta_z}{\mu_i+\theta_z}*\sum_{i=n_{A+1}}^n \frac{\mu_i\theta_z}{\mu_i+\theta_z}-(\sum_{i=n_{A+1}}^n \frac{\mu_i\theta_z}{\mu_i+\theta_z})^2}$ and $Var[\hat{\beta}_1] = \frac{\sum_{i=1}^n \frac{\mu_i\theta_z}{\mu_i+\theta_z}}{\sum_{i=1}^n \frac{\mu_i\theta_z}{\mu_i+\theta_z}*\sum_{i=n_{A+1}}^n \frac{\mu_i\theta_z}{\mu_i+\theta_z}-(\sum_{i=n_{A+1}}^n \frac{\mu_i\theta_z}{\mu_i+\theta_z})^2}$

g)

$D(y,\hat{\mu}) = 2\sum_{i=1}^n \omega_i[y_i(\tilde{\theta}_i-\hat{\theta}_i) - b(\tilde{\theta}_i) + b(\hat{\theta}_i)]$
We had for the negative binomial model that $\theta = \log(\frac{\mu}{\mu+\theta_z})$ For the saturated model we will have $\tilde{\mu}_i = y_i$. so $\tilde{\theta}_i = log(\frac{\hat{\mu}}{\hat{\mu}+\theta_z}) = log(\frac{y_i}{y_i+\theta_z})$ and $\hat{\theta}_i = log(\frac{\hat{\mu}}{\hat{\mu}+\theta_z})$
We also had for the model that $b(\theta) = -\theta_z log(1-e^\theta)$, thus we will have: $b(\hat{\theta}_i) = -\theta_z log(1-e^{log(\frac{\hat{\mu}}{\hat{\mu}+\theta_z})}) = -\theta_z log(1-\frac{\hat{\mu}}{\hat{\mu}+\theta_z}) = -\theta_z log(\frac{\theta_z}{\hat{\mu}+\theta_z})$ and $b(\tilde{\theta}_i) = -\theta_z log(1-e^{\tilde{\theta}_i}) = -\theta_z log(1-e^{log(\frac{y_i}{y_i+\theta_z})}) = -\theta_z log(1-\frac{y_i}{y_i+\theta_z}) = -\theta_z log(\frac{\theta_z}{y_i+\theta_z})$

So we can write the deviance as $2\sum_{i=1}^n y_i(log(\frac{y_i}{y_i+\theta_z}) - log(\frac{\hat{\mu}}{\hat{\mu}+\theta_z})) + \theta_z log(\frac{\theta_z}{y_i+\theta_z}) - \theta_z log(\frac{\theta_z}{\hat{\mu}+\theta_z})$
$= 2\sum_{i=1}^n y_i log(y_i) - y_i log(y_i+\theta_z) - y_i log(\hat{\mu}) + y_i log(\hat{\mu}+\theta_z) + \theta_z log(\theta_z) - \theta_z log(y_i+\theta_z) - \theta_z log(\theta_z) + \theta_z log(\hat{\mu}+\theta_z)$

$= 2\sum_{i=1}^n log(y_i+\theta_z)(-y_i-\theta_z) + log(\hat{\mu}+\theta_z)(y_i+\theta_z) + y_i log(y_i) - y_i log(\hat{\mu}_i)$
$= 2\sum_{i=1}^n -log(y_i+\theta_z)(y_i+\theta_z) + log(\hat{\mu}+\theta_z)(y_i+\theta_z) + y_i[log(\frac{y_i}{\hat{\mu}_i})]$

$= 2\sum_{i=1}^{n} y_i + \theta_z[log(\hat{\mu} + \theta_z) - log(y_i + \theta_z)] + y_i[log(\frac{y_i}{\hat{\mu}_i})]$
$= 2\sum_{i=1}^{n}(y_i + \theta_z)[log\frac{(\hat{\mu}+\theta_z)}{(y_i+\theta_z)}] + y_i[log(\frac{y_i}{\hat{\mu}_i})]$

A5)

The poisson distribution with paraneter μ can be written as an exponential family as $exp[-\mu + log\frac{\mu^y}{y!}]$
$= exp[-\mu + ylog(\mu) - log(y!)]$, so the log likelihood we will work with is: $\sum_{i=1}^{n}[-\mu + y_i log(\mu) - log(y!)] = -n\mu + log(\mu)\sum_{i=1}^{n} y_i - nlog(y!) = -n\mu + log(\mu)n\bar{y} - nlog(y!)$
So we have that the score (first derivative wrt μ) will be $\frac{\partial}{\partial\mu} - n\mu + log(\mu)n\bar{y} - nlog(y!) = -n + \frac{n\bar{y}}{\mu} = \frac{n\bar{y}-n\mu}{\mu} = \frac{n(\bar{y}-\mu)}{\mu}$
So the Hessian(second derivative) will be given by $\frac{\partial}{\partial\mu}\frac{n(\bar{y}-\mu)}{\mu} = \frac{-n\mu-[n(\bar{y}-\mu)]}{\mu^2} = \frac{-n\mu-n\bar{y}+n\mu}{\mu^2} = \frac{-n\bar{y}}{\mu^2}$
Finally the information will be given by the (negative) expectation of the Hessian, that is $E[\frac{n\bar{y}}{\mu^2}] = \frac{n}{\mu^2}E[\bar{y}] = \frac{n\mu}{\mu^2} = \frac{n}{\mu}$ So now we have everything we need for the algorithms. For fisher scoring, for a parameter B:
$B^{t+1} = B^t + (I^t)^{-1}u^t$ where I is the information, and u^t is the score. Thus plugging those in we will have:
$\mu^{t+1} = \mu^t + \frac{\mu^t}{n}\frac{n(\bar{y}-\mu^t)}{\mu^t} = \mu^t + \bar{y} - \mu = \bar{y}$

On the other hand, Newton-Rhapson will iterate

$B^{t+1} = B^t - (H^t)^{-1}u^t$ where H is the hessian. So plugging in what we know:
$\mu^{t+1} = \mu^t + \frac{(\mu^t)^2}{n\bar{y}}\frac{n(\bar{y}-\mu^t)}{\mu^t}$
$= \mu^t + \frac{\mu^t}{\bar{y}}(\bar{y} - \mu^t)$
$= \mu^t + \mu^t - \frac{(\mu^t)^2}{\bar{y}}$ so
$\mu^{t+1} = 2\mu^t - \frac{(\mu^t)^2}{\bar{y}}$. To have the previous result, we would need to have
$2\mu^t - \frac{(\mu^t)^2}{\bar{y}} = \bar{y} \implies 2\mu^t\bar{y} - (\mu^t)^2 = \bar{y}^2 \implies \bar{y}^2 - 2\mu^t\bar{y} + (\mu^t)^2 = 0 \implies (\bar{y}^2 - \mu^t)^2 = 0 \implies \mu^t = \bar{y}$. So for Newton Rhapson, if $\mu^t = \bar{y}$, we can have the Fisher Scoring result $\mu^{t+1} = \bar{y}$

A6)

```
awards <- read.csv("awards.csv")
attach(awards)
```

a)

```
fit1<-glm(numawards~1+math,family=poisson(link=log))
summary(fit1)
```

```
##
## Call:
## glm(formula = numawards ~ 1 + math, family = poisson(link = log))
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.1853  -0.9070  -0.6001   0.3246   2.9529
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -5.333532   0.591261  -9.021   <2e-16 ***
```

```
## math         0.086166   0.009679   8.902   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 287.67  on 199  degrees of freedom
## Residual deviance: 204.02  on 198  degrees of freedom
## AIC: 384.08
##
## Number of Fisher Scoring iterations: 6
```

β_0 is estimated as -5.333532 and β_1 is estimated as 0.086166 . The low p valuesuggests math to be a significant predictor.

```r
confint(fit1,level=0.95)
```

```
##                   2.5 %    97.5 %
## (Intercept) -6.52038334 -4.200322
## math         0.06737466  0.105356
```

So the CI is found as [0.06737466,0.105356]

b)

```r
fit2<-glm(numawards~as.factor(prog),family=poisson(link=log),x=TRUE)
summary(fit2)
```

```
##
## Call:
## glm(formula = numawards ~ as.factor(prog), family = poisson(link = log),
##     x = TRUE)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.4142  -0.6928  -0.6325   0.0000   3.3913
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)       -1.6094     0.3333  -4.828 1.38e-06 ***
## as.factor(prog)2   1.6094     0.3473   4.634 3.59e-06 ***
## as.factor(prog)3   0.1823     0.4410   0.413    0.679
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 287.67  on 199  degrees of freedom
## Residual deviance: 234.46  on 197  degrees of freedom
## AIC: 416.51
##
## Number of Fisher Scoring iterations: 6
```

The model has 3 parameters, and their estimated values are printed above. as.factor(prog)3 has a high corresponding p value thus does not appear to be significant, but the intercept and as.factor(prog)2 have low p values and thus are significant.

We confirm with the Wald test:

```r
I <- t(fit2$x)%*%diag(fit2$weights)%*%fit2$x

# Inverse Fisher information matrix (i.e. as. covariance matrix of the MLEs)

I.inv <- solve(I)

sd <- sqrt(diag(I.inv))

# computing the Wald statistics (testing whether beta_j = 0 for j=1,2)

beta <- fit2$coefficients
p.val <- pchisq((beta/sd)^2,df=1,lower.tail=FALSE)

p.val
```

```
##      (Intercept) as.factor(prog)2 as.factor(prog)3
##     1.376940e-06     3.590060e-06     6.792649e-01
```

So we see once again p values implying the significance of the intercept, as.factor(prog)2 but NOT of as.factor(prog)3

```r
#Wald Stat Testing

library(Matrix)
as.numeric(rankMatrix(I.inv))
```

```
## [1] 3
```

```r
W <- t(matrix(beta,nrow=3))%*%I%*%matrix(beta,nrow=3)
pchisq(W,df=3,lower.tail=FALSE)
```

```
##              [,1]
## [1,] 2.403968e-10
```

The test suggests that not all predictor parameters are 0, and:

```r
z <- qnorm(0.975)
c.upper <- beta+z*sd
c.lower <- beta-z*sd
CI <- cbind(c.lower,c.upper)
colnames(CI)<-c("2.5 %","97.5 %")
CI
```

```
##                       2.5 %     97.5 %
## (Intercept)      -2.2627592 -0.9561166
## as.factor(prog)2  0.9286927  2.2901831
## as.factor(prog)3 -0.6819413  1.0465844
```

As we can see from the Wald confidence intervals too, the estimate for as.factor(prog)3 contains 0, but the rest do not. Now we perform a likehlihood ratio test:

```r
anova(fit2, test = "LRT")
```

```
## Analysis of Deviance Table
##
## Model: poisson, link: log
##
## Response: numawards
```

```
##
## Terms added sequentially (first to last)
##
##
##                  Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                               199     287.67
## as.factor(prog)  2   53.212       197     234.46 2.787e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

As we can see as.factor(prog) has high significance.

c)

```r
fit3 <- glm(numawards~1+math+as.factor(prog),family=poisson, x=TRUE)
summary(fit3)
```

```
##
## Call:
## glm(formula = numawards ~ 1 + math + as.factor(prog), family = poisson,
##     x = TRUE)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.2043  -0.8436  -0.5106   0.2558   2.6796
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)       -5.24712    0.65845  -7.969 1.60e-15 ***
## math               0.07015    0.01060   6.619 3.63e-11 ***
## as.factor(prog)2  1.08386    0.35825   3.025  0.00248 **
## as.factor(prog)3  0.36981    0.44107   0.838  0.40179
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 287.67  on 199  degrees of freedom
## Residual deviance: 189.45  on 196  degrees of freedom
## AIC: 373.5
##
## Number of Fisher Scoring iterations: 6
```

```r
fit4 <- glm(numawards~1+math*as.factor(prog),family=poisson, x=TRUE)
summary(fit4)
```

```
##
## Call:
## glm(formula = numawards ~ 1 + math * as.factor(prog), family = poisson,
##     x = TRUE)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.2295  -0.7958  -0.5298   0.2528   2.6826
##
```

```
## Coefficients:
##                        Estimate Std. Error z value Pr(>|z|)
## (Intercept)            -3.86179    2.49317  -1.549    0.121
## math                    0.04400    0.04721   0.932    0.351
## as.factor(prog)2       -0.44107    2.60299  -0.169    0.865
## as.factor(prog)3       -0.84473    2.86990  -0.294    0.768
## math:as.factor(prog)2  0.02841    0.04870   0.583    0.560
## math:as.factor(prog)3  0.02290    0.05421   0.422    0.673
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 287.67  on 199  degrees of freedom
## Residual deviance: 189.10  on 194  degrees of freedom
## AIC: 377.16
##
## Number of Fisher Scoring iterations: 6
```

The first model has 4 parameters, and all of them except for as.factor(prog)3 appear to be significant with low p values. The second model has 6 parameters but has very high p values across the board, so doesn't appear to be that useful. Both models attempt to relate the expected number of awards with math and prog.
On the first fit, the model fit is (since log is canonical for poission) $logE[Y/X] = \beta_0 + \beta_1 * math + \beta_2 X_2 + \beta_3 X_3 \implies E(Y/X) = exp(\beta_0 + \beta_1 * math + \beta_2 X_2 + \beta_3 X_3)$ where $X_2 = 1$ if the student is in program 2 and $X_2 = 0$ if not, and $X_3 = 1$ if the student is in program 3 and $X_3 = 0$ if not. For math in this model we have a positive parameter value of 0.07015, so each point of increase in the math score increases $X\beta$ by 0.07015, and this corresponds to an $e^{0.07015} = 1.072$ times (so about 7%) increase in the number of awards received. Since prog is a factor predictor, to interpret them properly we would need the following:
$i) : logE[Y/X = prog1] = \beta_0 + \beta_1 * math$ since program 1 appears to be used as the reference class,
$ii) : logE[Y/X = prog2] = \beta_0 + \beta_1 * math + \beta_2$
$iii) : logE[Y/X = prog3] = \beta_0 + \beta_1 * math + \beta_3$

Note that $\beta_2 = ii - i$ and $\beta_3 = iii - i$. So we have that

$\beta_2 = logE[Y/X = prog2] - logE[Y/X = prog1] \implies e^{\beta_2} = e^{logE[Y/X=prog2]} * e^{-logE[Y/X=prog1]} \implies e^{\beta_2} * e^{logE[Y/X=prog1]} = e^{logE[Y/X=prog2]} \implies e^{\beta_2} * E[Y/X = prog1] = E[Y/X = prog2]$. Our β_2 estimate is 1.08386 Thus what the model says is that $e^{1.08386} = 2.95607$ The expected number of awards earned by a student in program 2 would be about 2.95 as much as the expected number of awards of one in program 1. The interpretation of β_3 is similar,
$\beta_3 = logE[Y/X = prog3] - logE[Y/X = prog1] \implies e^{\beta_3} = e^{logE[Y/X=prog3]} * e^{-logE[Y/X=prog1]} \implies e^{\beta_3} * e^{logE[Y/X=prog1]} = e^{logE[Y/X=prog3]} \implies e^{\beta_3} * E[Y/X = prog1] = E[Y/X = prog3]$ so according to the model (if we are to take β_3 to be a significant parameter, which we likely wouldn't due to the p value) the expected number of awards earned by a student in program 3 would be $e^{0.36981} = 1.44746$ times the expected number earned by one in program 1. Overall, the model says that a student in programs 2 or 3 is significantly more likely to earn an award than a student in program 1, and there is a strong (positive) statistical relationship between the number of awards earned and the math score.

As for the interpretation for the second model it is of the from $logE[Y/X] = \beta_0 + \beta_1 * math + \beta_2 X_2 + \beta_3 X_3 + \beta_4 * math * X_2 + \beta_5 * math * X_3$ with interactions.
$i) : logE[Y/X = prog1] = \beta_0 + \beta_1 * math$ since program 1 appears to be used as the reference class,
$ii) : logE[Y/X = prog2] = \beta_0 + \beta_1 * math + \beta_2 + \beta_4 * math$
$ii) : logE[Y/X = prog3] = \beta_0 + \beta_1 * math + \beta_3 + \beta_5 * math$

This time $ii - i$ gives $\beta_2 + \beta_4 math$ a,d $iii - 1$ gives $\beta_3 + \beta_5 math$ so we have a similar formulation to the revious model:
$\beta_2 + \beta_4 math = logE[Y/X = prog2] - logE[Y/X = prog1] \implies e^{\beta_2 + \beta_4 math} = e^{logE[Y/X=prog2]} * e^{-logE[Y/X=prog1]} \implies e^{\beta_2 + \beta_4 math} * e^{logE[Y/X=prog1]} = e^{logE[Y/X=prog2]} \implies e^{\beta_2 + \beta_4 math} * E[Y/X =$

$prog1] = E[Y/X = prog2]$. So we can interpret the model in a similar way but this time math effects the proportion of the expectaitons too, that is an increase of a math score by one causes $\beta_2 + \beta_4 math$ to increase by $\beta_4 = 0.02841$ which is an $e^{0.02841} = 1.0288$ times an increase in $e^{\beta_2+\beta_4 math}$, and in turn with the additional effect of the prorgram the student is in (β_2), the expected value of the number of awards earned by a student in program 2 is $e^{-0.44107+0.02841*math}$ times as much as the expected value of the number of awards earned by a student in program 1. Note that this term depends on the math score as well, which is what differs it from the previous model.

Equivalently,
$\beta_3 + \beta_5 math = logE[Y/X = prog3] - logE[Y/X = prog1] \implies e^{\beta_3+\beta_5 math} = e^{logE[Y/X=prog3]} * e^{-logE[Y/X=prog1]} \implies e^{\beta_3+\beta_5 math} * e^{logE[Y/X=prog1]} = e^{logE[Y/X=prog3]} \implies e^{\beta_3+\beta_5 math} * E[Y/X = prog1] = E[Y/X = prog3]$

The interpretation is identical, an increase of a math score by one causes $\beta_3 + \beta_5 math$ to increase by $\beta_5 = 0.02290$ which is an $e^{0.02290} = 1.023$ times an increase in $e^{\beta_3+\beta_5 math}$, and in turn with the additional effect of the prorgram the student is in (β_3), the expected value of the number of awards earned by a student in program 2 is $e^{-0.84473+0.02290*math}$ times as much as the expected value of the number of awards earned by a student in program 1.

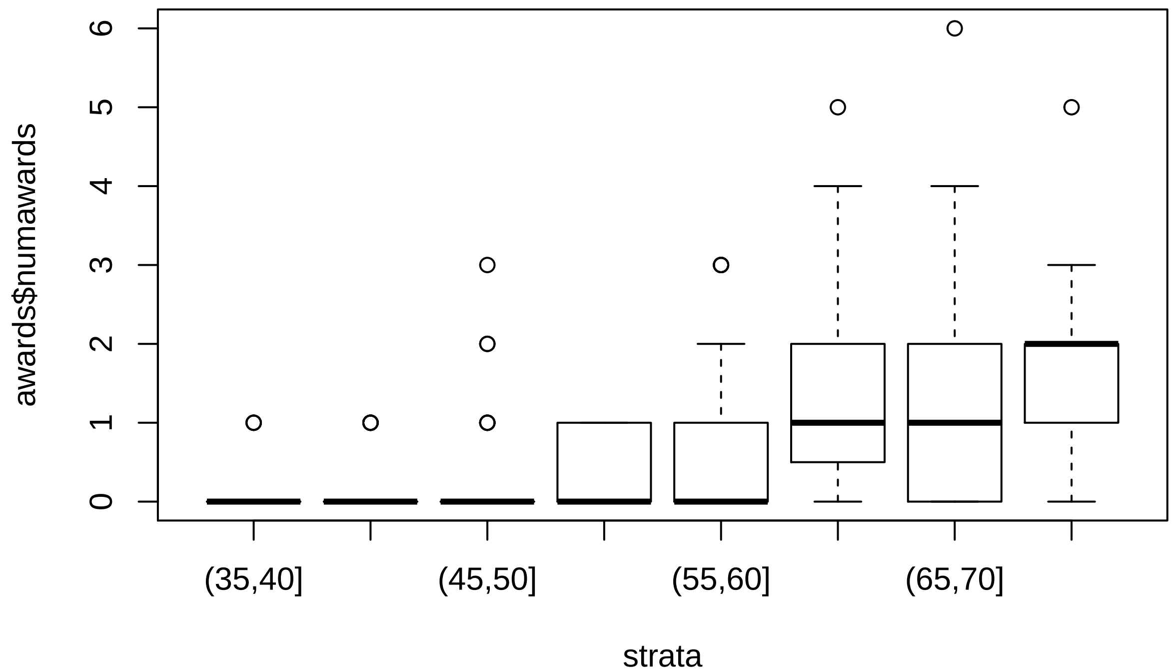
Note that in this model $E[Y/X = prog1]$ is able to be higher depending on the value of math.

This model however overall has a high lack of significance, due to the high p values.

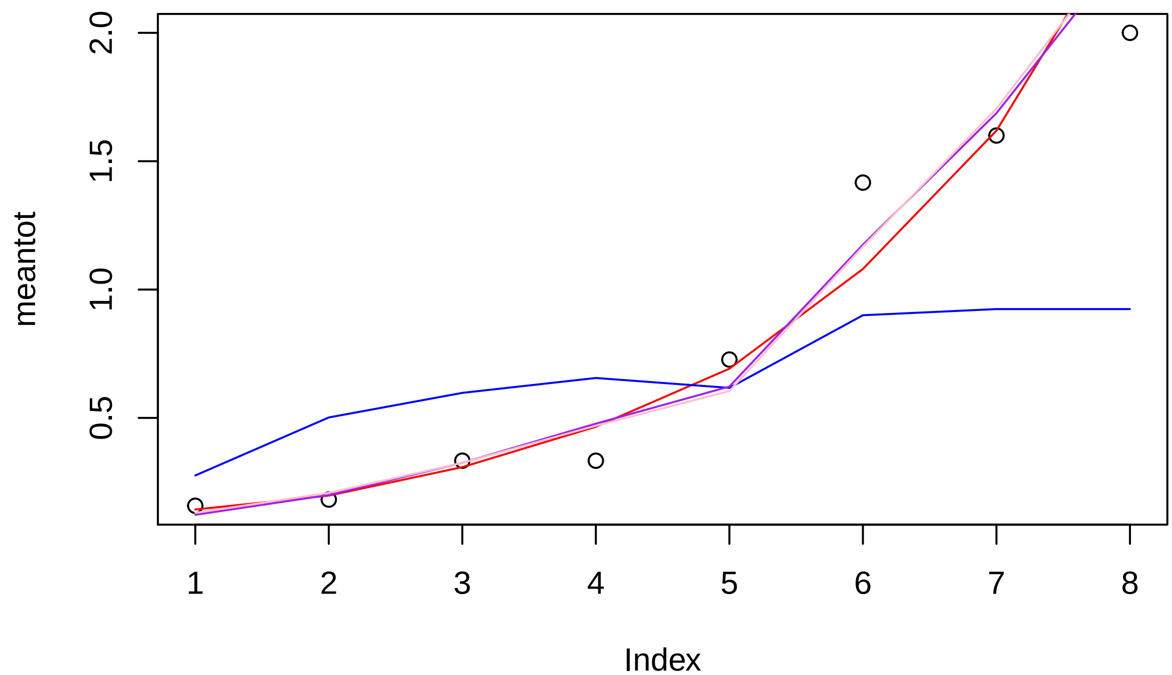
d)

We stratify the data as instructed and consider how well the models fit the means of each strata:

```
awards <- cbind(awards, model1=fit1$fitted.values,model2=fit2$fitted.values, model3=fit3$fitted.values,
strata <- cut(awards$math, breaks=c(35, 40, 45, 50, 55, 60, 65, 70, 75))
set_new <- split(awards, strata)
plot(awards$numawards~strata)
```




```r
meantot <- vector(mode="numeric", length=8)
model1_mean <- vector(mode="numeric", length=8)
model2_mean <- vector(mode="numeric", length=8)
model3_mean <- vector(mode="numeric", length=8)
model4_mean <- vector(mode="numeric", length=8)
i <- 1
for (val in set_new) {
  meantot[i] <- mean(val$numawards)
  model1_mean[i] <- mean(val$model1)
  model2_mean [i] <- mean(val$model2)
  model3_mean[i] <- mean(val$model3)
  model4_mean[i] <- mean(val$model4)
  mean(val$model3)
  mean(val$model4)
  i <- i+1 }
plot(meantot)
lines(model1_mean, col="Red")
lines(model2_mean, col="Blue")
lines(model3_mean, col="Purple")
lines(model4_mean, col="Pink")
```



Model 1 (red) and model 3 (purple) seem to be fitting well.

e)

Analysis of deviance:

```r
anova(fit1, fit2, fit3, fit4, test="Chi")
```

```
## Analysis of Deviance Table
##
## Model 1: numawards ~ 1 + math
## Model 2: numawards ~ as.factor(prog)
## Model 3: numawards ~ 1 + math + as.factor(prog)
## Model 4: numawards ~ 1 + math * as.factor(prog)
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1       198     204.02
## 2       197     234.46  1  -30.439
## 3       196     189.45  1   45.010 1.96e-11 ***
## 4       194     189.10  2    0.348   0.8403
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The third model has the lowest p value, and in fact seems to be the only model with the significantly low p value, so it would seem that the first model from part c) is the best suited one. The interpretation was provided in part c), the model says that a student in programs 2 or 3 is significantly more likely to earn an award than a student in program 1, and there is a strong statistical relationship between the number of awards earned and the math score.