STAT 778 HW 3

Tom Wallace

April 3, 2018

Program Organization and Compilation

Source code is contained in `hw3.c`. The program requires the GNU Scientific Library (GSL), an open-source numerical library. It can be obtained from `www.gnu.org/software/gsl`; or, it can installed from most standard Linux package managers. An example command to achieve the latter is:

```
sudo apt-get install gsl-bin libgsl-dev
```

Compilation of `hw3.c` is best achieved in two steps. First, use the below command to compile the program but not link it. You may need to change the argument passed to the `-I` flag to wherever the `gsl` header files live on your computer.

```
gcc -I/usr/include -c hw3.c
```

This command should create an object file `hw3.o`. Link this object file to relevant libraries with the following command. You may need to change the argument passed to the `-L` flag to wherever `libgsl` lives on your computer.

```
gcc -L/usr/lib hw3.o -o hw3 -lgsl -lgslcblas -lm
```

Once successfully compiled, the program can be executed. It requires one argument, the name of the input file. An example command is:

```
./hw3 HW2_2018.dat
```

Technical Notes

This program finds maximum partial likelihood estimates (MPLE) for coefficients $\beta_j \ldots \beta_k = \boldsymbol{\beta}$ in the Cox proportional hazards model. This section provides the details of the computation of these estimates.

The log-likelihood function for the Cox proportional hazards model is:

$$\log L(\boldsymbol{\beta}) = \sum_{i=1}^{n} \delta_i \left(\boldsymbol{\beta'x}_i - \log \sum_{l \in R(t_i)} \exp\left(\boldsymbol{\beta' x}_l\right) \right) \tag{1}$$

with:

$t_i ... t_n$	$=$	unique observation times
δ_i	$=$	indicator function, which returns 0 if observation t_i is right-censored
$\boldsymbol{x}_i$	$=$	vector of covariates associated with individual observed at t_i
$\boldsymbol{\beta}$	$=$	vector of coefficients associated with covariates
$R(t_i)$	$=$	risk set at t_i

The first order partial derivatives are:

$$\frac{\partial \log L}{\partial \beta_j} = \sum_{i=1}^{n} x_j - \frac{\sum_{l \in R(t_i)} x_{lj} \exp(\boldsymbol{\beta' x}_l)}{\sum_{l \in R(t_i)} \exp(\boldsymbol{\beta' x}_l)} \tag{2}$$

It can be shown that these functions are convex, and so the their roots—i.e., the value of β_k for which the above function equals 0—is the MPLE estimator $\hat{\beta}_k$. The Newton-Raphson algorithm is used to numerically estimate the roots (there is no closed-form solution). This algorithm also requires the derivatives of the function to be solved. Thus, we also need the second-order partial derivatives of the log likelihood function.

$$\frac{\partial^2 \log L}{\beta_j \beta_k} = \sum_{i=1}^{n} \frac{1}{\sum_{l \in R(t_i)} \exp(\boldsymbol{\beta}' \boldsymbol{x}_l)} \left(\sum_l x_{lj} x_{lk} \exp(\boldsymbol{\beta}' \boldsymbol{x}_l) - \frac{\left(\sum_l x_{lj} \exp(\boldsymbol{\beta}' \boldsymbol{x}_l)\right)\left(\sum_l x_{lk} \exp(\boldsymbol{\beta}' \boldsymbol{x}_l)\right)}{\sum_l \exp(\boldsymbol{\beta}' \boldsymbol{x}_l)} \right) \tag{3}$$

The program uses these expressions to find the first- and second-order partial derivatives of the log likelihood function and supplies them to the Newton-Raphson algorithm.

Verification and Validation