Gov 2001: Problem Set 4 Solutions

Due Wednesday, March 2 by 6pm

Problem 1: Maximizing a Poisson Likelihood

Suppose $Y_i \sim Poisson(\lambda)$. You take two draws from the distribution of Y_i, y_1 and y_2.

1.A) Write out the expression for the likelihood function, $L(\lambda \mid y_1, y_2)$

$$L(\lambda \mid y_1, y_2) \propto P(y_1, y_2|\lambda)$$
$$L(\lambda \mid y_1, y_2) \propto \left(\frac{\lambda^{y_1} e^{-\lambda}}{y_1!}\right)\left(\frac{\lambda^{y_2} e^{-\lambda}}{y_2!}\right)$$

1.B) Write out the expression for the log-likelihood function $\ell(\lambda) = \log\left[L(\lambda \mid y_1, y_2)\right]$

$$\ell(\lambda \mid y_1, y_2) = \log\left[L(\lambda \mid y_1, y_2)\right]$$
$$\ell(\lambda \mid y_1, y_2) = \log\left[\left(\frac{e^{-\lambda}\lambda^{y_1}}{y_1!}\right)\left(\frac{e^{-\lambda}\lambda^{y_2}}{y_2!}\right)\right]$$
$$\ell(\lambda \mid y_1, y_2) = \log\left(\frac{e^{-\lambda}\lambda^{y_1}}{y_1!}\right) + \log\left(\frac{e^{-\lambda}\lambda^{y_2}}{y_2!}\right)$$
$$\ell(\lambda \mid y_1, y_2) = \log\left(e^{-\lambda}\right) + \log\left(\lambda^{y_1}\right) - \log\left(y_1!\right) + \log\left(e^{-\lambda}\right) + \log\left(\lambda^{y_2}\right) - \log\left(y_2!\right)$$
$$\ell(\lambda \mid y_1, y_2) = -\lambda\left[\log(e)\right] + y_1\left[\log(\lambda)\right] - \log\left(y_1!\right) - \lambda\left[\log(e)\right] + y_2\left[\log(\lambda)\right] - \log\left(y_2!\right)$$
$$\ell(\lambda \mid y_1, y_2) = -2\lambda + \left[\log(\lambda)\right](y_1 + y_2) - \log\left(y_1!\right) - \log\left(y_2!\right)$$

1.C) Write out the expression for the first derivative of the log-likelihood function $\ell'(\lambda) = \frac{\partial}{\partial\lambda}\left(\log\left[L(\lambda \mid y_1, y_2)\right]\right)$

$$\ell'(\lambda \mid y_1, y_2) = \frac{\partial}{\partial\lambda}\left[\ell(\lambda \mid y_1, y_2)\right]$$
$$\ell'(\lambda \mid y_1, y_2) = \frac{\partial}{\partial\lambda}\left[-2\lambda + \left[\log(\lambda)\right](y_1 + y_2) - \log\left(y_1!\right) - \log\left(y_2!\right)\right]$$
$$\ell'(\lambda \mid y_1, y_2) = -2 + \frac{1}{\lambda}(y_1 + y_2)$$

1.D) Write out the expression for the second derivative of the log-likelihood function $\ell''(\lambda) = \frac{\partial^2}{\partial\lambda^2}\left(\log\left[L(\lambda \mid y_1, y_2)\right]\right)$

$$\ell''(\lambda \mid y_1, y_2) = \frac{\partial^2}{\partial\lambda^2}\left[\ell(\lambda \mid y_1, y_2)\right]$$
$$\ell''(\lambda \mid y_1, y_2) = \frac{\partial}{\partial\lambda}\left[\ell'(\lambda \mid y_1, y_2)\right]$$
$$\ell''(\lambda \mid y_1, y_2) = \frac{\partial}{\partial\lambda}\left[-2 + \frac{1}{\lambda}(y_1 + y_2)\right]$$
$$\ell''(\lambda \mid y_1, y_2) = -\frac{1}{\lambda^2}(y_1 + y_2)$$

1.E) Find the MLE for λ by setting $\ell(\lambda)' = 0$ and checking that $\ell(\lambda)'' < 0$.
The first step here is to set our score, $\ell'(\lambda \mid y_1, y_2) = 0$ and solve for $\hat{\lambda}$:

$$\ell'(\lambda \mid y_1, y_2) = -2 + \frac{1}{\lambda}(y_1 + y_2)$$
$$\ell'(\lambda \mid y_1, y_2) = 0$$
$$0 = -2 + \frac{1}{\lambda}(y_1 + y_2)$$
$$-\frac{1}{\lambda}(y_1 + y_2) = -2$$
$$\frac{1}{\lambda}(y_1 + y_2) = 2$$
$$\frac{1}{\lambda} = \frac{2}{(y_1 + y_2)}$$
$$\hat{\lambda} = \frac{(y_1 + y_2)}{2}$$

In part D, we found that $\ell''(\lambda \mid y_1, y_2) = -\frac{1}{\lambda^2}(y_1 + y_2)$. Note that λ is a rate of arrival, where $\lambda > 0$. Similarly, y_1 and y_2 represent positive integer event counts in a Poisson model, so their sum must also be positive. Since $\frac{1}{\lambda^2}(y_1 + y_2)$ must then be positive (1 over the square of a positive value multiplied by another positive value), the negative sign in front ensures that our second derivative will be negative at our MLE. Our Hessian in this case is:

$$
\begin{aligned}
H(\lambda \mid y_1, y_2) &= -\frac{1}{\hat{\lambda}^2}(y_1 + y_2) \\
H(\lambda \mid y_1, y_2) &= -\frac{1}{\left(\frac{(y_1+y_2)}{2}\right)^2}(y_1 + y_2) \\
H(\lambda \mid y_1, y_2) &= -\frac{1}{\frac{(y_1+y_2)^2}{4}}(y_1 + y_2) \\
H(\lambda \mid y_1, y_2) &= -\frac{1}{\frac{(y_1+y_2)}{4}} \\
H(\lambda \mid y_1, y_2) &= -\frac{4}{(y_1 + y_2)}
\end{aligned}
$$

Which is always going to be negative for $y_1 > 0$ and $y_2 > 0$, so we know that our log likelihood function reaches a maximum at $\hat{\lambda}$.

1.F) Suppose $y_1 = 2$ and $y_2 = 4$. What is the value of your MLE, using the analytical solution you provided in part E?

$$
\begin{aligned}
\hat{\lambda} &= \frac{(y_1 + y_2)}{2} \\
\hat{\lambda} &= \frac{(2 + 4)}{2} \\
\hat{\lambda} &= 3
\end{aligned}
$$

1.G) Before we begin using `optim` to maximize likelihood and log-likelihood functions, we'll use this simple case to program our own numerical optimization. Recall from section that one common approach to numerical optimization is the Newton-Raphson method, where:

$$\lambda^{(i+1)} = \lambda^i - \frac{\ell'(\lambda^i)}{\ell''(\lambda^i)}$$

For an arbitrary starting value in the support of λ, $\lambda^{(0)}$, the expression above can be iterated until resulting values of $\lambda^{(i+1)}$ converge to our MLE. Write out the update formula for the Newton-Raphson algorithm that would calculate our MLE for λ.

$$
\begin{aligned}
\lambda^{(i+1)} &= \lambda^i - \frac{\ell'(\lambda^i)}{\ell''(\lambda^i)} \\
\lambda^{(i+1)} &= \lambda^i - \frac{-2 + \frac{1}{\lambda}(y_1 + y_2)}{-\frac{1}{\lambda^2}(y_1 + y_2)}
\end{aligned}
$$

1.H) Write a function that implements your formula from part G in R. Your function should take λ and your data as inputs and return an updated value of λ.

```
nr <- function(lambda,y){
   return(lambda - (-2 + (1/lambda) * (sum(y)))/(-(1/lambda^2)*sum(y)))
 }
```

1.I) Suppose again that $y_1 = 2$ and $y_2 = 4$. In R, use the update formula provided in part G and your function from part H to calculate your MLE numerically. Set $\lambda^{(0)}$ to 1.0. Does the algorithm converge to your answer in part G?

```
y <- c(2, 4)
 diff <- 10
 init.lambda <- 1
 lambdas <- NULL
 while(diff > 0.0001){
   temp <- init.lambda
   init.lambda <- nr(init.lambda, y)
   lambdas <- c(lambdas, init.lambda)
   diff <- init.lambda - temp
 }

 lambdas

 [1] 1.666667 2.407407 2.882945 2.995433 2.999993 3.000000
```

This converges to our analytical solution for $\hat{\lambda}$ in just a few iterations!

1.J) Finally, use `optim` with the following arguments `control(fnscale = -1), method = "BFGS", hessian = TRUE` to calculate your MLE for λ. You can continue to assume that $y_1 = 2$ and $y_2 = 4$. Do your results match your results from F? Comment on the relationship between your hand-coded algorithm and this approach.

Hint: you'll need to write a function for your log-likelihood from part B.

```
# First, let's write out our Poisson log likelihood function

  ll.1j <- function(lambda, y){
    -2 * lambda + sum(y) * log(lambda)
  }

  # Note that I'm dropping all portions of this that don't have
  # lambda in them. These get absorbed into the proportionality
  # constant and don't affect the maximum

# Now let's run optim in this log likelihood function
  opt.1j <- optim(par = 1, fn = ll.1j,
```

```
            y = y,
            control = list(fnscale = -1),
            method = "BFGS",
            hessian = T)

opt.1j$par # Shows our MLE of 3!
```

These results match the results from our Newton-Raphson algorithm and our analytical solution. BFGS is a variation on Newton Raphson, so we should expect the same results from these two algorithms unless we're dealing with a functional form that's particularly irregular or starting one of these at a point far away from the maximum (Newton-Raphson has a hard time finding it's way out of a flat portion of your function).

Problem 2: Maximizing a Normal Likelihood

$Y_i \sim N(\beta, 1)$ where β is constant across all Y_i. You receive the following data which consists of three realizations of Y: $(y_1 = -1, y_2 = 0, y_3 = 1)$. Assume all Y are independent of one another.

2.A) Find an expression for the log-likelihood of β conditional on your data. Remove any constants that will not affect the maximization of this function. Show your derivation.

$$\begin{aligned}
\ln L(\beta|\mathbf{y}) &= \ln[k(y)p(\mathbf{y}|\beta)] \\
&= \ln[k(y)p(y_1|\beta)p(y_2|\beta)p(y_3|\beta)] \\
&= \ln k(y) + \ln\left(\prod_{i=1}^{3} \frac{1}{\sqrt{2\pi}} e^{\frac{-(y_i-\beta)^2}{2}}\right) \\
&= \ln k(y) + 3\ln\frac{1}{\sqrt{2\pi}} + \sum_{i=1}^{3} \frac{-(y_i-\beta)^2}{2}.
\end{aligned}$$

At this point we can dispense with any additive constants.

$$\begin{aligned}
\ln L(\beta|\mathbf{y}) &\doteq \sum_{i=1}^{3} \frac{-(y_i-\beta)^2}{2} \\
&= -\frac{1}{2}\left((-1-\beta)^2 + (0-\beta)^2 + (1-\beta)^2\right) \\
&= -\frac{1}{2}\left((1+2\beta+\beta^2) + \beta^2 + (1-2\beta+\beta^2)\right) \\
&= -\frac{1}{2}(2+3\beta^2) \\
&= -1 - \frac{3}{2}\beta^2
\end{aligned}$$

2.B) Find the maximum likelihood estimate of β analytically.

$$\frac{\partial \ln L(\beta|\mathbf{y})}{\partial \beta} = -3\beta.$$

At a critical point this is equal to zero, so we can set this equal to zero and solve for β yielding $\hat{\beta} = 0$ as a proposed maximum likelihood estimate. You may have also answered $\hat{\beta} = \frac{1}{3}\sum_{i=1}^{3} y_i = \bar{y}$ if you stopped at the beginning of the second set of equations above.

2.C) Confirm that this is a maximum by checking the second derivative.

We confirm that it is a maximum by checking that the second derivative is negative at the critical point:

$$\frac{\partial^2 \ln L(\beta|\mathbf{y})}{\partial \beta^2} = -3.$$

2.D) $-\frac{\partial^2 \ln L(\beta|\mathbf{y})}{\partial \beta^2}$, i.e. the negation of the 2nd derivative you calculated for part C, is an estimate of what is sometimes called 'the information'. Using the concept of the likelihood ratio (which on the log scale is a difference in log-likelihoods), explain in 3 sentences or less why this quantity might be a useful summary of the precision of the maximum likelihood estimate of a parameter.

The likelihood ratio is used to compare the plausibility of proposed estimates for a parameter. A large, negative second derivative at the maximum of the log-likelihood function (assuming a reasonably well-behaved likelihood) is indicative of a rapidly decreasing likelihood function in the neighborhood of the maximum, which implies larger likelihood ratios (or log-likelihood differences) for the MLE in comparison to parameter values nearby. In contrast, a small, negative second derivative implies a flatter likelihood and smaller likelihood ratios (smaller log-likelihood differences) when one compares the MLE with surrounding values.

2.E) Create a function in R for the log-likelihood from part A.

```
ll.sty.norm <- function(par, data){
  loglike <- -sum(.5*(data-beta)^2)
  return(loglike)
}
```

2.F) Confirm your answer from part B by maximizing your function from part E using `optim` with the following arguments `control(fnscale = -1), method = "BFGS", hessian = TRUE`. The hessian matrix is the matrix of second derivatives of a function, and note that in this case, it is 1 by 1 because our function has only one argument. Note that `optim` also provides the value of the likelihood at its maximum as `value`.

The `optim()` function locates the maximum at zero, and calculates

```
data <- c(-1,0,1)

opt <- optim(par =  .3, ll.sty.norm, data = data, control =
             list(fnscale = -1), method = "BFGS", hessian = TRUE)
```

2.G) Use the Taylor Series expansion below to approximate the log-likelihood for β in the vicinity of its maximum likelihood estimate.

Given $f(a)$, $f'(a)$, and $f''(a)$ where a is some value in the domain of f then

$$f(x) \approx f(a) + f'(a)(x-a) + \frac{f''(a)}{2}(x-a)^2.$$

Hint: for your problem f is the log-likelihood function and a is $\hat{\beta}_{MLE}$.

All derivatives are evaluated at the maximum:

$$\begin{aligned}
\ln L(\beta|\mathbf{y}) &\approx \ln L(\hat{\beta}|\mathbf{y}) + \frac{\partial \ln L(\beta|\mathbf{y})}{\partial \beta}(\beta - \hat{\beta}) + \frac{\partial^2 \ln L(\beta|\mathbf{y})}{\partial \beta^2}\frac{(\beta-\hat{\beta})^2}{2} \\
&= -1 + 0 \cdot (\beta - 0) - \frac{3}{2}(\beta - 0)^2 \\
&= -1 - \frac{3}{2}(\beta^2)
\end{aligned}$$

2.H) What do you think of this approximation for $\ln L(\beta|\mathbf{y})$? Based on your answer, would you consider your answer from part D a statement of the exact curvature or the approximate curvature of the likelihood around the maximum?
This approximation is exact, because the log-likelihood of a normal distribution is a quadratic function.

Problem 3

For this problem you will work with presidential election data. To begin, download the presidential.Rdata data file and load it into R using the following line of code:

```
load("presidential.Rdata")
```

These data, which are stored in matrix format, contain the dependent variable, Dvote, which is the fraction that the Democratic presidential candidate won in each state. The rest of the covariates include the following pieces of information for each state and year:

- year - Year of the race (from 1948 to 1992)
- n1 - Support for Dem candidate in Sept nationwide poll
- n2 - Presidential approval in July nationwide poll * Inc
- n3 - Presidential approval in July nationwide poll * Presinc
- n4 - 2nd Quarter GDP nationwide growth
- s1 - Dem share in the state the last election
- s2 - Dem share of state vote two elections ago
- s3 - Home state indicator variable
- s4 - Home state of VP candidate indicator
- s5 - Democratic minority in state legislature
- s6 - State economic growth in past year * Inc
- s7 - Measure of state ideology
- s8 - State ideological compatibility with candidates
- r1 - South indicator variable (i.e., the state is in the South)
- r2 - South in 1964*-1
- r3 - Deep South in 1964*-1
- r4 - New England in 1964
- r5 - North Central in 1972
- r6 - West in 1976*-1

Note that all variables are constructed so that an increase in the variable would be expected to increase Democratic vote share. Also, for the following problems, you may assume that (1) the observations are independent across states i and years t, and (2) the democratic vote share is normally distributed.

You may assume that σ^2 is constant from year to year and across states. That is, $Y_{it} \sim N(\mu_{it}, \sigma^2)$.

3.A) Parameterize the systematic component as $\mu_{it} = X_{it}\beta$. Using this parameterization, derive the log likelihood, $lnL(\beta, \sigma|y)$. Show the steps of your derivation.

$$
\begin{aligned}
lnL(\beta,\sigma^2|y) &= ln\left[\prod \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y_i-\mu_{it})^2}{2\sigma^2}}\right] \\
&= \sum ln\left[\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y_i-X_{it}\beta)^2}{2\sigma^2}}\right] \\
&= \sum ln\left[\frac{1}{\sqrt{2\pi\sigma^2}}\right] + ln\left[e^{-\frac{(y_i-X_{it}\beta)^2}{2\sigma^2}}\right] \\
&= \sum -\frac{1}{2}ln(2\pi\sigma^2) - \frac{(y_i - X_{it}\beta)^2}{2\sigma^2} \\
&= \sum -\frac{1}{2}[ln(2\pi\sigma^2) + \frac{(y_i - X_i\beta)^2}{\sigma^2}] \\
&= \sum -\frac{1}{2}[ln(\sigma^2) + ln(2\pi) + \frac{(y_i - X_i\beta)^2}{\sigma^2}] \\
&= \sum -\frac{1}{2}[ln\sigma^2 + \frac{(y_i - X_i\beta)^2}{\sigma^2}] - \frac{1}{2}ln(2\pi) \\
&\propto \sum -\frac{1}{2}[ln\sigma^2 + \frac{(y_i - X_i\beta)^2}{\sigma^2}]
\end{aligned}
$$

3.B) Write a function in `R` that calculates this log likelihood. Your function should take as its inputs proposed parameter values for β, a vector for y, and a matrix for X.

```
load("presidential.RData")

ll.normal <- function(par,y,x){
  betas <- par[-length(par)]
  sigma2 <- exp(par[length(par)]) # the very last parameter is the variance

  covariates <- as.matrix(cbind(1, x))
  xb <- covariates %*% betas

  return(-1/2 * (sum(log(sigma2) + (y -(xb))^2 / sigma2)))
}
```

3.C) Check that your function works, and then use the `optim` function to maximize the log likelihood for all covariates in the dataset. (Hint: For this problem, you'll want to give your `optim` function starting values of zero and use the method `BFGS`.) Report the MLE coefficient estimates for the Intercept, n_1, n_2, s_2, and r_3.

Be sure that you constrain your parameter for σ^2 to be positive by reparameterizing it in your function as e^{σ^2}.

Note: your model should use all covariates in the dataset, but you will only report a subset of them here. You can use the `lm()` function to check your work.

The table below reports the results, along with the standard errors. To get the standard errors we extract the Hessian matrix from the optim function, multiply it by -1 and take its inverse. The variances are located on the diagonals of this matrix, so taking the square root of the diagonals gives us the standard errors.

Table 1: **Parameter MLEs for OLS in Optim**

	coef	se
Intercept	-1.0531	0.2969
year	0.0007	0.0002
n1	0.3726	0.0258
n2	-0.0129	0.0057
n3	0.0478	0.0078
n4	0.0284	0.0018
s1	0.2950	0.0352
s2	0.2708	0.0285
s3	0.0425	0.0079
s4	0.0131	0.0077
s5	0.0350	0.0097
s6	0.0012	0.0004
s7	0.0339	0.0055
s8	0.0413	0.0173
r1	0.0588	0.0076
r2	0.0834	0.0161
r3	0.1845	0.0256
r4	0.0791	0.0112
r5	0.0688	0.0127
r6	0.0633	0.0091
e^{σ^2}	-6.6528	0.0597

```
# Test the Function:
  ll.normal(par = rep(0,ncol(data)+1), y = data[,"Dvote"], x = data[,-1])

  opt.3c <- optim(par = rep(0, ncol(data)+1), fn = ll.normal,
                  y = data[,"Dvote"], x = data[,-1],
                  control = list(fnscale = -1), method = "BFGS",
                  hessian = TRUE)

  res <- data.frame(coef = opt.3c$par, se = sqrt(diag(solve(-opt.3c$hessian))))

  row.names(res) <- c("Intercept",colnames(data)[2:20], "$e^{\\sigma^2}$")
```

```
library(xtable)

print(xtable(res, digits = 4, caption="{\\bf Parameter MLEs for OLS in Optim}",
 align = "rcc"),
      include.rownames = T,
      caption.placement="top",
      sanitize.text.function=function(x){x},
      file = "tab3c.tex")
```

3.D) Extract and save the variance-covariance matrix from your optim output, and use this to calculate the standard errors associated with your coefficient estimates from Part C. Report the standard errors for the Intercept and n_1.

See the table.

```
sqrt(diag(solve(-opt.3c$hessian)))
```

Problem 4: Likelihood Ratio Tests

A referee is challenging your model specification from Problem 3. He claims that your model would be a lot simpler and elegant if you just didn't include the regional variables: `r1`, `r2`, `r3`, `r4`, `r5`, and `r6` (conveniently coded in your dataset from problem 3!).

4.A) Specify the restricted and unrestricted models associated with the referee's claims.

The unrestricted model is just the model specific with all the covariates included:

$$Y = X\beta$$

The restricted model has the regional variables, r1, r2, r3, r4, r5, and r6 missing.

4.B) Using `R` (and your function from Problem 3), calculate the likelihood ratio test statistic as a function of the two likelihoods. Report this value.

Likelihood ratio statistic: **305.4489**

```
# Get values of normal log likelihood function

# Full model
  unrestricted <- optim(par = rep(0, ncol(data)+1),
                        fn = ll.normal,
                        y = data[,"Dvote"],
                        x = data[,-1],
                        control = list(fnscale = -1),
```

```
                          method = "BFGS",
                          hessian = TRUE)
  unrestricted$value

# Restricted model
  restricted <- optim(par = rep(0, 15),
                      fn = ll.normal,
                      y = data[,"Dvote"],
                      x = data[,2:14],
                      control = list(fnscale = -1),
                      method = "BFGS",
                      hessian = TRUE)
  restricted$value

# Calculate test statistic
  r <- 2*(unrestricted$value-restricted$value)
  r
  [1] 305.4489
```

4.C) Calculate the probability of seeing this likelihood ratio test statistic under the null hypothesis that the restricted model is true. Do you accept the referee's suggestion of getting rid of the regional variables?

To do this we use a χ^2 test, with the degrees of freedom equal to the number of restrictions (6 in this case). To do this we can use the `pchisq()` function.

```
1 - pchisq(r, df = 6)
```

The probability of seeing this test statistic under the null that the restrictions are valid is extremely low, basically zero. So we reject the referee's suggestion.

R code

Please submit all your code for this assignment as a .R file. Your code should be clean, commented, and executable without error.