Homework 4

Louis Bensard

March 12, 2018

Problem J-2.3:

(a)

```
data = read.table('http://mathfaculty.fullerton.edu/mori/data/math534/serum_conc.txt', header=TRUE)
attach(data)

model <- function(alpha_1, lambda_1, alpha_2, lambda_2, t){

  E_1 = exp(-lambda_1*t)
  E_2 = exp(-lambda_2*t)
  f = alpha_1*E_1 + alpha_2*E_2 + exp(1)
  grad = cbind(E_1, -alpha_1*t*E_1, E_2, -alpha_2*t*E_2)
  attr(f,'gradient') <- grad
  return(f)
}


Result = nls(Concentration ~ model(alpha_1, lambda_1, alpha_2, lambda_2, Time),
             start=list(alpha_1=100, lambda_1=0.02, alpha_2=200, lambda_2=0.05),trace = TRUE,
             nls.control(maxiter = 50, tol = 1e-5, minFactor = 1/1024, printEval=TRUE))
```

```
## Warning in min(x): no non-missing arguments to min; returning Inf

## Warning in max(x): no non-missing arguments to max; returning -Inf

## 231329.2 :  1e+02 2e-02 2e+02 5e-02
## 165322 :   3649.2381888     0.4007787 -3400.0114242     0.4667008
## 3638.549 :   384.3248817     0.3681203 -136.8087595    0.4861802
## 2860.741 :  223.6343815   0.3269572  23.1456850   0.4911771
## 2722.359 :  236.5027530   0.2902003   8.9183250   1.1011997
## 2543.661 :  227.0596711   0.2789450  17.8750132   0.8591281
## 1868.352 :  210.9306596   0.2588181  33.2112609   0.9181446
## 942.6622 :  190.7676906   0.2289185  52.1856133   1.0299582
## 206.0498 :  172.469038   0.194188  69.307326   1.203755
## 45.25495 :  163.667019   0.170934  78.092532   1.397188
## 44.52444 :  164.4062140   0.1727339  77.4877186   1.3935490
## 44.52417 :  164.3565437   0.1726581  77.5274507   1.3923631
## 44.52417 :  164.3651625   0.1726686  77.5231205   1.3926456
## 44.52417 :  164.3634247   0.1726666  77.5239309   1.3925867
## 44.52417 :  164.363781   0.172667  77.523766   1.392599
```

```
summary(Result)
```

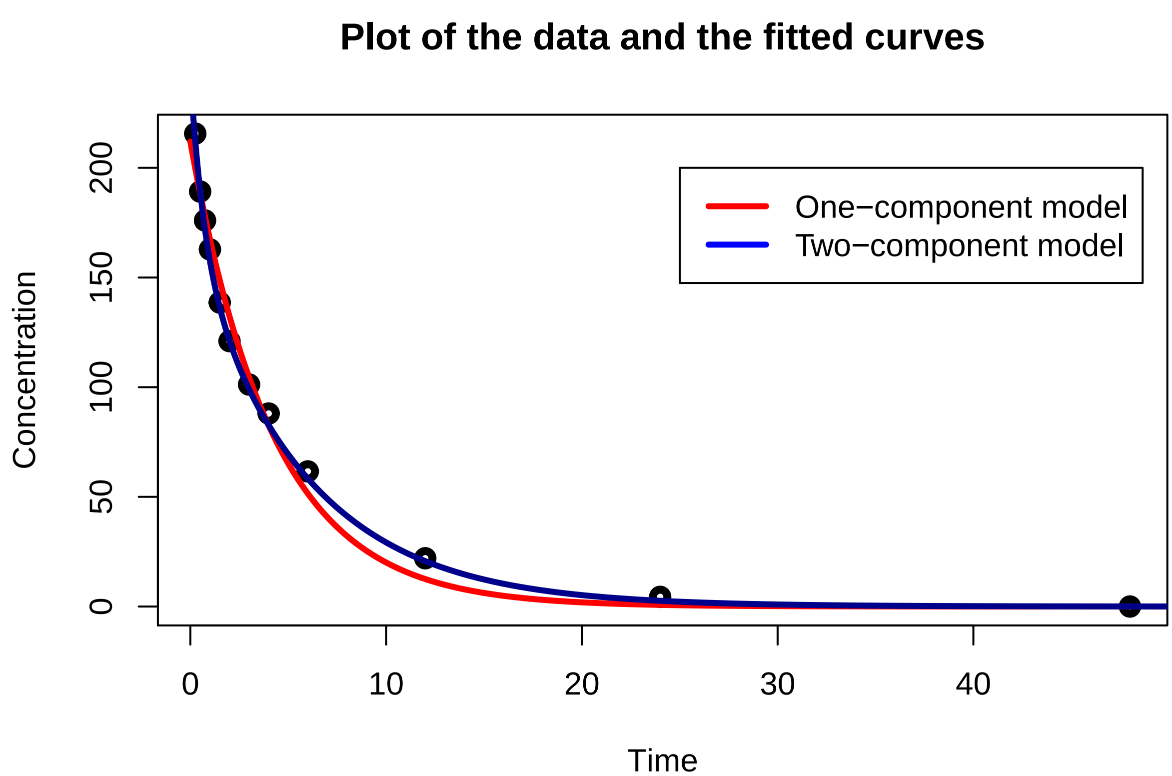
```
##
## Formula: Concentration ~ model(alpha_1, lambda_1, alpha_2, lambda_2, Time)
##
```

```
## Parameters:
##            Estimate Std. Error t value Pr(>|t|)
## alpha_1  164.36378    7.99697  20.553 3.29e-08 ***
## lambda_1   0.17267    0.01057  16.339 1.98e-07 ***
## alpha_2   77.52377    7.02524  11.035 4.05e-06 ***
## lambda_2   1.39260    0.25861   5.385 0.000658 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.359 on 8 degrees of freedom
##
## Number of iterations to convergence: 14
## Achieved convergence tolerance: 3.455e-06
```

(b)

```r
plot(Time, Concentration, lwd=4, main='Plot of the data and the fitted curves')

t = seq(0,50, 0.1)
lines(t, 211.9203*exp(-0.2357*t), col='red', lwd=3)
lines(t, 164.3638*exp(-0.1727*t) + 77.5238*exp(-1.3926*t), col = 'darkblue', lwd=3)
legend(25, 200, c("One-component model", "Two-component model"),col=c("red", "blue"), lty=1, lwd=3)
```



The two component model does seem to fit better the data than the one component model.

Problem J-2.4:

(a)

```
data = read.table('C:/Users/Louis/Documents/UPMC/M1/Spring 2018/MATH 534/Homework 4/data_j-2.4.txt', hea
attach(data)

X = as.matrix(cbind(Dose, Number_tested))
y = as.matrix(Tumor_incidence)

beta_0=c(0.1,.001,1e-5)
maxit = 30
tolgrad = 1e-9 ; tolerr = 1e-6
#converged solution used to compute the convergence ratio
beta_star = c(4.5945e-02, 1.6271e-03, 1.0192e-05)

MLE_beta = GN(y, X, beta_0, list_f_Jac_W_bin, Wt = 1, maxit, IRLS = TRUE, tolgrad, tolerr, beta_star)
```

```
##
##  it      beta_0            beta_1            beta_2            ||grad||     MRE      CV ratio
##   1    0.100000000000    0.001000000000    0.000010000000    3.4e+05    5.4e-02    4.0e+00
##   2    0.046764840575    0.001434258033    0.000010906535    1.9e+04    7.1e-04    5.7e-01
##   3    0.046228237059    0.001605337160    0.000010273248    1.1e+03    2.6e-04    4.8e-01
##   4    0.045986795559    0.001623570252    0.000010206146    9.9e+01    3.9e-05    5.0e-01
##   5    0.045951197971    0.001626526332    0.000010194705    1.7e+01    6.0e-06    6.6e-01
##   6    0.045945636306    0.001626980803    0.000010192955    2.5e+00    9.3e-07    1.5e+00
##   7    0.045944775691    0.001627051275    0.000010192684    3.9e-01    1.4e-07    3.3e+00
##   8    0.045944642346    0.001627062190    0.000010192642    6.1e-02    2.2e-08    3.0e+00
##   9    0.045944621690    0.001627063881    0.000010192635    9.4e-03    3.5e-09    3.0e+00
##  10    0.045944618490    0.001627064143    0.000010192634    1.5e-03    5.4e-10    3.0e+00
##  11    0.045944617994    0.001627064184    0.000010192634    2.3e-04    8.3e-11    3.0e+00
##  12    0.045944617917    0.001627064190    0.000010192634    3.5e-05    1.3e-11    3.0e+00
##  13    0.045944617905    0.001627064191    0.000010192634    5.4e-06    2.0e-12    3.0e+00
##  14    0.045944617903    0.001627064191    0.000010192634    8.4e-07    3.1e-13    3.0e+00
##  15    0.045944617903    0.001627064191    0.000010192634    1.3e-07    4.8e-14    3.0e+00
##  16    0.045944617903    0.001627064191    0.000010192634    2.0e-08    7.4e-15    3.0e+00
##  17    0.045944617903    0.001627064191    0.000010192634    3.0e-09    1.2e-15    3.0e+00
##  18    0.045944617903    0.001627064191    0.000010192634    1.7e-09    1.9e-16    3.0e+00
##  19    0.045944617903    0.001627064191    0.000010192634    9.3e-10    3.6e-17    3.0e+00
```

```
cat("MLE_beta = ",MLE_beta)
```

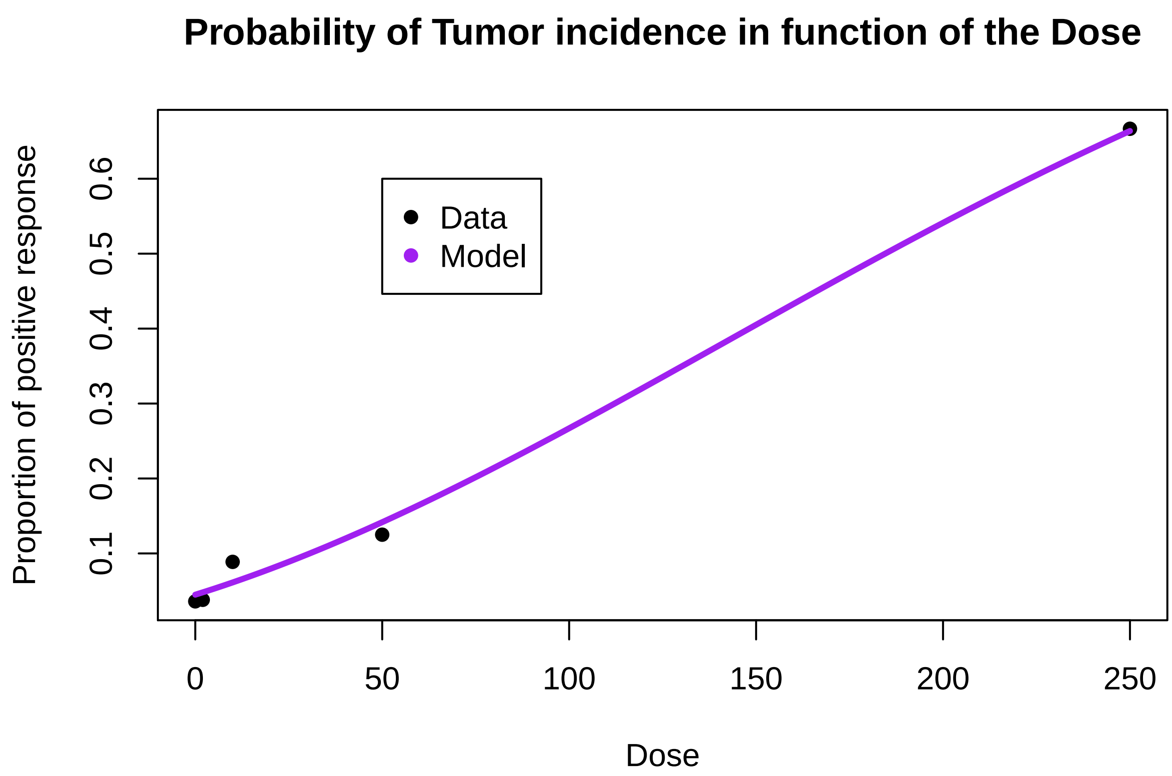
```
## MLE_beta =  0.04594462 0.001627064 1.019263e-05
```

Therefore, the MLE of $\boldsymbol{\beta}$ is $\hat{\boldsymbol{\beta}} = (4.59 \cdot 10^{-2}, 1.63 \cdot 10^{-3}, 1.02 \cdot 10^{-5})^T$.

(b)

```
plot(Dose, Tumor_incidence/Number_tested, lwd=2, ylab="Proportion of positive response", pch=16, main='

x = seq(0,250, length=100)
lines(x, 1 - exp(-MLE_beta[1] - MLE_beta[2]*x - MLE_beta[3]*x^2), col='purple', lwd=3)
legend(50, 0.6, c("Data", "Model") ,col=c("black", "purple"), pch = 16)
```



Problem J-2.5:

(a)

```
data = read.table('http://mathfaculty.fullerton.edu/mori/math534/examdata/blowBF.txt', header=TRUE)
attach(data)
```

```
## The following object is masked _by_ .GlobalEnv:
##
##     y
```

```
y = data$y

X = as.matrix(cbind(1, log(D), S))
y = as.matrix(y)

beta_0=c(0,0,0)
maxit = 30
tolgrad = 1e-9 ; tolerr = 1e-6
#converged solution used to compute the convergence ratio
beta_star = c(-9.562084874971, 3.197563503535, 4.508593181753)

MLE_beta = GN(y, X, beta_0, list_f_Jac_W_bern, Wt = 1, maxit, IRLS = TRUE, tolgrad, tolerr, beta_star)
```

```
##
## it        beta_0            beta_1            beta_2           ||grad||     MRE      CV ratio
##   1   0.000000000000   0.000000000000   0.000000000000   2.4e+02   8.1e+00   1.1e+00
##   2  -6.124979085543   2.035718241648   2.857736621916   4.1e+01   3.1e+00   7.8e-01
##   3  -8.673430233033   2.895866528488   4.076634312631   7.1e+00   1.0e+00   2.4e-01
##   4  -9.492692680136   3.173968058887   4.474456484213   4.5e-01   8.4e-02   1.9e-02
##   5  -9.561642171111   3.197412849302   4.508373266712   2.6e-03   5.4e-04   1.2e-04
##   6  -9.562084856863   3.197563497368   4.508593172694   9.6e-08   2.2e-08   9.0e-05
##   7  -9.562084874971   3.197563503535   4.508593181753   1.0e-12   2.5e-15   3.0e+00
```

```r
cat("MLE_beta = ",MLE_beta)
```

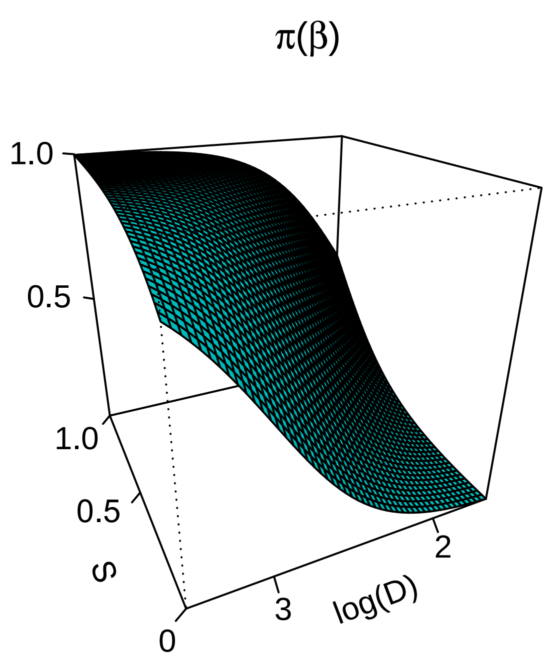
```
## MLE_beta =  -9.562085 3.197564 4.508593
```

Therefore, the MLE of $\boldsymbol{\beta}$ is $\hat{\boldsymbol{\beta}} = (-9.5621, 3.1976, 4.5086)^T$.

(b)

```r
n=50
x1 = seq(0,1, length=n) # = S
x2 = seq(min(X[,2]), max(X[,2]), length=n) #log (D)
x3 = outer(x1, x2, pi_fct_MLE) #pi(S, log(D))

persp(x1,x2,x3, theta=245, phi=25, r=2, shade=0.4, axes=TRUE, box=TRUE,
    ticktype="detailed" , nticks= 3, col="cyan", xlab="S", ylab="log(D)",
    zlab="", main=expression(paste(pi,'(', beta, ')')), expand = 1)
```



(c)

```
#prediction of the model for S = 0.3 and D = 10
pi_fct_MLE(0.3, log(10))
```

```
## [1] 0.3000949
```

Therefore, for a Severity of 0.3 and a tree Diameter of 10, our model predicts that there is a probbility of 0.30 that the tree would be blown down.