Midtest

Masroor Hossain(45765758)

4/16/2021

1 a)

$J(w) = \frac{1}{m}\sum_{i=1}^{m} L(\hat{y_i}, y_i)$

where,

$L(\hat{y_i}, y_i) = -(y_i log(\hat{y_i}) + (1 - y_i) * log(1 - \hat{y_i}))$

and,

$\hat{y_i} = \sigma(w^T x)$

$J(w) = \frac{1}{m}\sum_{i=1}^{m} -[y_i log(\hat{y_i}) + (1 - y_i) log(1 - \hat{y_i})]$

$J(w) = -\frac{1}{m}\sum_{i=1}^{m} [y_i log(\hat{y_i}) + (1 - y_i) log(1 - \hat{y_i})]$

$J(w) = -\frac{1}{m}\sum_{i=1}^{m} [y_i log(a^i) + (1 - y_i) log(1 - a^i))]$

$J(w) = -\frac{1}{m}\sum_{i=1}^{m} [y_i log(h_\theta(x_i)) + (1 - y_i) log(1 - h_\theta(x_i))]$

$J(w) = -\frac{1}{m}\sum_{i=1}^{m} [y_i log(\sigma(w^T x_i)) + (1 - y_i) log(1 - \sigma(w^T x_i)))]$

Two important properties of logistic regression can be derived from this. They are:

First property is:

$1 - \sigma(w^T x) = 1 - \frac{1}{1+e^{-w^T x}}$

$1 - \sigma(w^T x) = \frac{1+e^{-w^T x}-1}{1+e^{-w^T x}}$

$1 - \sigma(w^T x) = \frac{e^{-w^T x}}{1+e^{-w^T x}}$

Second property is:

$\sigma(w^T x) = \frac{1}{1+e^{-w^T x}}$

$\sigma(w^T x) = (1 + e^{-w^T x})^-1$

$\frac{\partial}{\partial w^T x}(\sigma(w^T x)) = \frac{\partial}{\partial w^T x}(1 + e^{-w^T x})^-1$

$\frac{\partial}{\partial w^T x}(\sigma(w^T x)) = (-1)(1 + e^{-w^T x})^-2(0 + e^{-w^T x}(-1))$

$\frac{\partial}{\partial w^T x}(\sigma(w^T x)) = \frac{e^{-w^T x}}{(1+e^{-w^T x})^2}$

This can be written as:

$\frac{\partial}{\partial w^T x}(\sigma(w^T x)) = \frac{1}{1+e^{-w^T x}} * \frac{e^{-w^T x}}{1+e^{-w^T x}}$

$\frac{\partial}{\partial w^T x}(\sigma(w^T x)) = \sigma(w^T x) * (1 - \sigma(w^T x))$

$J(w) = -\frac{1}{m}\sum_{i=1}^{m}[y_i log(\sigma(w^T x_i)) + (1 - y_i) log(1 - \sigma(w^T x_i)))]$

The gradient of cost function can be written as:

$\nabla J(w) = -\frac{1}{m}\sum_{i=1}^{m}\frac{\partial}{\partial w^T}[y_i log(\sigma(w^T x_i)) + (1 - y_i) log(1 - \sigma(w^T x_i)))]$

Now, I can apply the chain rule to find the gradient of the cost function

$\frac{\partial}{\partial w^T} log(\sigma(w^T x_i)) = \frac{1}{\sigma(w^T x_i)}\frac{\partial \sigma(w^T x_i)}{\partial w^T}$

$\frac{\partial}{\partial w^T} log(\sigma(w^T x_i)) = \frac{1}{\sigma(w^T x_i)}\frac{\partial \sigma(w^T x_i)}{\partial w^T x_i}\frac{\partial w^T x_i}{\partial w^T}$

$\frac{\partial}{\partial w^T} log(\sigma(w^T x_i)) = \frac{1}{\sigma(w^T x_i)}\frac{\partial \sigma(w^T x_i)}{\partial w^T x_i}\frac{\partial w^T x_i}{\partial w^T}$

$\frac{\partial}{\partial w^T} log(\sigma(w^T x_i)) = \frac{1}{\sigma(w^T x_i)}\sigma(w^T x_i)(1 - \sigma(w^T x_i))x_i$

$\frac{\partial}{\partial w^T} log(\sigma(w^T x_i)) = (1 - \sigma(w^T x_i))x_i$

$\frac{\partial}{\partial w^T} log(1 - \sigma(w^T x_i)) = \frac{1}{1-\sigma(w^T x_i)}\frac{\partial 1-\sigma(w^T x_i)}{\partial w^T}$

$\frac{\partial}{\partial w^T} log(1 - \sigma(w^T x_i)) = \frac{1}{1-\sigma(w^T x_i)} - \sigma(w^T x_i)(1 - \sigma(w^T x_i))x_i$

$\frac{\partial}{\partial w^T} log(1 - \sigma(w^T x_i)) = -\sigma(w^T x_i)x_i$

Now, the gradient of our cost function is

$\nabla(J(w)) = \frac{\partial J(w)}{\partial w^T} = -\frac{1}{m}\sum_{i=1}^{m}\frac{\partial}{\partial w^T}[y_i \frac{\partial}{\partial w^T} log(\sigma(w^T x_i) + (1 - y_i)\frac{\partial}{\partial w^T} log(1 - \sigma(w^T x_i))]$

$\nabla(J(w)) = \frac{\partial J(w)}{\partial w^T} = -\frac{1}{m}\sum_{i=1}^{m}(y_i(1 - \sigma(w^T x_i))x_i + (1 - y_i)(-\sigma(w^T x_i)x_i))$

$\nabla(J(w)) = \frac{\partial J(w)}{\partial w^T} = -\frac{1}{m}\sum_{i=1}^{m}(y_i(1 - \sigma(w^T x_i))x_i + (1 - y_i)(-\sigma(w^T x_i)x_i))$

$\nabla(J(w)) = \frac{\partial J(w)}{\partial w^T} = \frac{1}{m}\sum_{i=1}^{m}(-y_i(1 - \sigma(w^T x_i))x_i - (1 - y_i)(-\sigma(w^T x_i)x_i))$

$\nabla(J(w)) = \frac{\partial J(w)}{\partial w^T} = \frac{1}{m}\sum_{i=1}^{m}(-y_i x_i + x_i y_i \sigma(w^T x_i) + \sigma(w^T x_i)x_i - y_i \sigma(w^T x_i)x_i)$

$\nabla(J(w)) = \frac{\partial J(w)}{\partial w^T} = \frac{1}{m}\sum_{i=1}^{m}(x_i(\sigma(w^T x_i) - y_i)$

1 b)

The Hessian matrix of cost function is:

$\nabla^2(J(w)) = \frac{\partial J(w)}{\partial w^T \partial w}$

$\frac{\partial J(w)}{\partial w^T \partial w} = \frac{1}{m}\sum_{i=1}^{m}(x_i(\frac{\partial}{\partial w}(\sigma(w^T x_i)) - y_i)$

It is know,

$\partial log(\sigma(w^T x)) = \frac{\partial \sigma(w^T x)}{\sigma(w^T x)}$

$\partial\sigma(w^Tx) = \sigma(w^Tx)\partial log(\sigma(w^Tx))$

$\frac{\partial\sigma(w^Tx)}{\partial w} = \sigma(w^Tx)\frac{\partial log(\sigma(w^Tx))}{\partial w}$

$\frac{\partial\sigma(w^Tx)}{\partial w} = \sigma(w^Tx)\frac{1}{\sigma(w^Tx_i)}\frac{\partial\sigma(w^Tx_i)}{\partial w^T}$

$\frac{\partial\sigma(w^Tx)}{\partial w} = \sigma(w^Tx)\frac{1}{\sigma(w^Tx_i)}\sigma(w^Tx_i)(1-\sigma(w^Tx))x^T$

$\frac{\partial\sigma(w^Tx)}{\partial w} = \sigma(w^Tx)(1-\sigma(w^Tx))x^T$

$\frac{\partial J(w)}{\partial w^T\partial w} = \frac{1}{m}\sum_{i=1}^{m}(x_i\sigma(w^Tx)(1-\sigma(w^Tx))x^T)$

So,

$\nabla^2(J(w)) = \frac{\partial J(w)}{\partial w^T\partial w} = \frac{1}{m}\sum_{i=1}^{m}(x_i\sigma(w^Tx)(1-\sigma(w^Tx))x^T)$

Here, as both xi and xi^T are concatenation of column vectors for m number of samples, it can be written,

$\sum_{i=1}^{m}(x_ix^T) = XX^T$

The scalar matrix is ,

$D = \sigma(w^Tx)(1-\sigma(w^Tx))$

Thus, the Hessian matrix can be written as :

$\overrightarrow{H(w)} = \nabla^2(J(w)) = \sum_{i=1}^{m}XX^TD$

1 c)

$\overrightarrow{H(w)} = \nabla^2(J(w)) = \sum_{i=1}^{m}XX^TD$

By the square root of this D matrix, it can be found,

$\nabla^2(J(w)) = \sum_{i=1}^{m}XX^TD(\frac{1}{2})D(\frac{1}{2})$

So this becomes,

$\nabla^2(J(w)) = \sum_{i=1}^{m}(XD(\frac{1}{2}))^T(XD(\frac{1}{2}))$

Here, D cannot be negative as it is based on sigmoid function, and also, X^tX is a squared term which automatically makes it positive. Thus, Hessian matrix is positive semidefinite and J(w) is convex.

2 a) This problem is concerned with predicting whether the email is spam or not spam. If the email is spam, then the output is 1, and if the email is not spam, then, it is 0. So, the response variable is a binary type. This is a classification problem. The logistic regression model for this problem can be defined by applying the sigmoid function to the linear predictor of this problem.

$Yi \sim Bernoulli(\sigma(w^Tx))$

$Yi \sim Bernoulli(\sigma(z))$

So,

$\sigma(z) = \frac{1}{1+e^{-z}}$

where,

$-z = -\beta_0 - \beta_1 * x_1 - \beta_2 * x_2 - \beta_3 * x_3 - \beta_4 * x_4 - \beta_5 * x_5 - \beta_6 * x_6 - \beta_7 * x_7 - \beta_8 * x_8 - \beta_9 * x_9 - \beta_{10} * x_{10} - \beta_{11} * x_{11} - \beta_{12} * x_{12} - \beta_{13} * x_{13} - \beta_{14} * x_{14} - \beta_{15} * x_{15} - \beta_{16} * x_{16} - \beta_{17} * x_{17} - \beta_{18} * x_{18} - \beta_{19} * x_{19} - \beta_{20} *$

$x_{20} - \beta_{21} * x_{21} - \beta_{22} * x_{22} - \beta_{23} * x_{23} - \beta_{24} * x_{24} - \beta_{25} * x_{25} - \beta_{26} * x_{26} - \beta_{27} * x_{27} - \beta_{28} * x_{28} - \beta_{29} * x_{29} - \beta_{30} * x_{30} - \beta_{31} * x_{31} - \beta_{32} * x_{32} - \beta_{33} * x_{33} - \beta_{34} * x_{34} - \beta_{35} * x_{35} - \beta_{36} * x_{36} - \beta_{37} * x_{37} - \beta_{38} * x_{38} - \beta_{39} * x_{39} - \beta_{40} * x_{40} - \beta_{41} * x_{41} - \beta_{42} * x_{42} - \beta_{43} * x_{43} - \beta_{44} * x_{44} - \beta_{45} * x_{45} - \beta_{46} * x_{46} - \beta_{47} * x_{47} - \beta_{48} * x_{48} - \beta_{49} * x_{49} - \beta_{50} * x_{50} - \beta_{51} * x_{51} - \beta_{52} * x_{52} - \beta_{53} * x_{53} - \beta_{54} * x_{54} - \beta_{55} * x_{55} - \beta_{56} * x_{56} - \beta_{57} * x_{57}$

Here,

$x_1 = make\ term$

$x_2 = address\ term$

$x_3 = all\ term$

$x_4 = num3d\ term$

$x_5 = our\ term$

$x_6 = over\ term$

$x_7 = remove\ term$

$x_8 = internet\ term$

$x_9 = order\ term$

$x_{10} = mail\ term$

$x_{11} = receive\ term$

$x_{12} = will\ term$

$x_{13} = people\ term$

$x_{14} = report\ term$

$x_{15} = addresses\ term$

$x_{16} = free\ term$

$x_{17} = business\ term$

$x_{18} = email\ term$

$x_{19} = you\ term$

$x_{20} = credit\ term$

$x_{21} = your\ term$

$x_{22} = font\ term$

$x_{23} = num000\ term$

$x_{24} = money\ term$

$x_{25} = hp\ term$

$x_{26} = hpl\ term$

$x_{27} = george\ term$

$x_{28} = num650\ term$

$x_{29} = lab\ term$

$x_{30} = labs\ term$

$x_{31} = telnet\ term$

$x_{32} = num857\ term$

$x_{33} = data\ term$

$x_{34} = num415\ term$

$x_{35} = num85\ term$

$x_{36} = technology\ term$

$x_{37} = num1999\ term$

$x_{38} = parts\ term$

$x_{39} = pm\ term$

$x_{40} = direct\ term$

$x_{41} = cs\ term$

$x_{42} = meeting\ term$

$x_{43} = original\ term$

$x_{44} = project\ term$

$x_{45} = re\ term$

$x_{46} = edu\ term$

$x_{47} = table\ term$

$x_{48} = conference\ term$

$x_{49} = charSemicolon\ term$

$x_{50} = charRoundbracket\ term$

$x_{51} = charSquarebracket\ term$

$x_{52} = charExclamation\ term$

$x_{53} = charDollar\ term$

$x_{54} = charHash\ term$

$x_{55} = capitalAve\ term$

$x_{56} = capitalLong\ term$

$x_{57} = capitalTotal\ term$

2 b)

```
a=load("C:/Users/Dell/Downloads/SPAM.Rdata")
head(a)
```

```
## [1] "train_data" "test_data"
```

It can be seen that there are two dataframes known as training and testing. Below is the summarized version of two dataframes.

```
head(test_data)
```

```
##      make address  all num3d  our over remove internet order mail receive will
## 1570 0.09    0.09 1.14     0 0.38 0.00      0     0.09  0.00 0.19    0.38 0.19
## 2338 0.00    0.00 0.00     0 0.00 0.00      0     0.00  0.00 0.00    0.00 1.11
```

```
## 3278 0.00    0.00 0.33     0 0.00 0.49      0     1.32  0.16 5.12    0.00 0.00
## 2776 0.00    0.00 0.00     0 0.00 0.00      0     0.00  0.00 0.84    0.00 0.00
## 426  0.33    0.00 0.66     0 0.22 0.00      0     0.00  0.44 0.11    0.00 0.33
## 3417 0.00    0.00 0.00     0 0.00 0.49      0     0.49  0.00 0.00    0.00 0.00
##      people report addresses free business email  you credit your font num000
## 1570      0   0.00         0 0.66     0.00     0 1.52      0 1.42    0      0
## 2338      0   0.00         0 0.00     0.00     0 0.00      0 0.00    0      0
## 3278      0   0.66         0 0.00     0.33     0 0.33      0 0.00    0      0
## 2776      0   0.00         0 0.00     0.84     0 1.68      0 0.00    0      0
## 426       0   0.00         0 0.55     0.00     0 1.76      0 1.10    0      0
## 3417      0   0.00         0 0.00     0.00     0 0.49      0 0.00    0      0
##      money   hp  hpl george num650 lab labs telnet num857 data num415 num85
## 1570  0.00 0.00 0.00   0.00      0   0    0      0      0    0      0  0.00
## 2338  0.00 1.11 1.11   0.00      0   0    0      0      0    0      0  0.00
## 3278  0.00 0.00 0.00   0.00      0   0    0      0      0    0      0  0.16
## 2776  0.00 0.00 0.00   0.00      0   0    0      0      0    0      0  0.00
## 426   0.22 0.00 0.00   0.00      0   0    0      0      0    0      0  0.00
## 3417  0.00 0.00 0.00   0.49      0   0    0      0      0    0      0  0.00
##      technology num1999 parts   pm direct cs meeting original project   re  edu
## 1570          0    0.00  0.00 0.00      0  0    0.00     0.38       0 0.00 0.00
## 2338          0    0.00  0.00 0.00      0  0    0.00     0.00       0 0.00 0.00
## 3278          0    0.00  0.00 0.00      0  0    0.16     0.00       0 0.00 0.33
## 2776          0    0.84  0.00 0.84      0  0    0.00     0.84       0 0.84 0.84
## 426           0    0.00  0.11 0.00      0  0    0.00     0.11       0 0.00 0.00
## 3417          0    0.49  0.00 0.00      0  0    0.00     0.00       0 0.00 0.00
##      table conference charSemicolon charRoundbracket charSquarebracket
## 1570     0          0         0.044            0.059             0.000
## 2338     0          0         0.000            0.183             0.000
## 3278     0          0         0.000            0.070             0.023
## 2776     0          0         0.000            0.000             0.137
## 426      0          0         0.000            0.173             0.000
## 3417     0          0         0.000            0.228             0.000
##      charExclamation charDollar charHash capitalAve capitalLong capitalTotal
## 1570           0.591      0.000    0.000      3.280          31          771
## 2338           0.000      0.000    0.000      1.800           4           36
## 3278           0.000      0.000    0.023      1.552          10          149
## 2776           0.413      0.000    0.137      3.052          13          116
## 426            0.367      0.193    0.077      2.559          75          389
## 3417           0.000      0.000    0.000      1.962           5          106
##         type
## 1570    spam
## 2338 nonspam
## 3278 nonspam
## 2776 nonspam
## 426     spam
## 3417 nonspam
```

```
head(train_data)
```

```
##      make address  all num3d  our over remove internet order mail receive will
## 273  0.25    0.25 0.00     0 0.75 0.00    0.0     0.00  0.25 0.75    0.00 1.51
## 3542 0.00    0.00 0.24     0 0.00 0.00    0.0     0.12  0.12 0.00    0.00 0.60
## 2859 0.00    0.00 0.00     0 0.00 0.00    0.0     0.00  0.00 0.00    0.00 0.00
## 4361 0.00    1.57 1.18     0 0.00 0.00    0.0     0.00  0.00 2.36    0.00 0.78
```

```
## 1076 0.00    0.55 0.55     0 1.10 0.55    2.2     0.00  0.00 0.55    0.00 0.55
## 420  0.51    0.43 0.29     0 0.14 0.03    0.0     0.18  0.54 0.62    0.29 0.65
##      people report addresses free business email  you credit your font num000
## 273    0.00   1.26      0.00 0.00     0.50  0.00 3.29      0 1.01 0.00   0.00
## 3542   0.12   0.12      0.00 0.00     0.72  0.00 0.00      0 0.00 0.00   0.00
## 2859   0.00   0.00      0.00 0.00     0.00  0.00 0.00      0 0.00 0.00   0.00
## 4361   0.00   0.00      0.00 0.00     0.00  0.00 0.39      0 0.00 6.29   0.00
## 1076   0.00   0.00      0.00 0.00     0.00  0.55 3.31      0 1.10 0.00   0.00
## 420    0.65   1.20      0.03 0.21     0.43  0.03 3.03      0 1.35 0.00   0.51
##      money   hp hpl george num650 lab labs telnet num857 data num415 num85
## 273   0.00 0.00   0   0.00      0   0    0      0      0 0.25      0     0
## 3542  0.00 1.81   0   0.00      0   0    0      0      0 0.00      0     0
## 2859  0.00 0.00   0   1.17      0   0    0      0      0 0.00      0     0
## 4361  0.00 0.00   0   0.00      0   0    0      0      0 0.00      0     0
## 1076  0.00 0.00   0   0.00      0   0    0      0      0 0.00      0     0
## 420   0.54 0.00   0   0.00      0   0    0      0      0 0.00      0     0
##      technology num1999 parts pm direct cs meeting original project   re edu
## 273        0.00    0.00     0  0      0  0       0        0       0 0.00   0
## 3542       0.12    0.12     0  0      0  0       0        0       0 0.00   0
## 2859       0.00    0.00     0  0      0  0       0        0       0 0.00   0
## 4361       0.00    0.00     0  0      0  0       0        0       0 0.00   0
## 1076       0.00    0.00     0  0      0  0       0        0       0 0.55   0
## 420        0.00    0.00     0  0      0  0       0        0       0 0.03   0
##      table conference charSemicolon charRoundbracket charSquarebracket
## 273      0          0         0.000            0.082             0.000
## 3542     0          0         0.105            0.060             0.000
## 2859     0          0         0.000            0.186             0.186
## 4361     0          0         1.151            0.203             0.000
## 1076     0          0         0.000            0.165             0.000
## 420      0          0         0.012            0.078             0.000
##      charExclamation charDollar charHash capitalAve capitalLong capitalTotal
## 273            0.041      0.124    0.124      3.181          32          210
## 3542           0.000      0.000    0.000      1.827          23          466
## 2859           0.000      0.000    0.000      3.862          28          112
## 4361           0.271      0.000    0.067      5.689          30          330
## 1076           0.496      0.000    0.082     16.826         148          387
## 420            0.443      0.510    0.133      6.590         739         2333
##         type
## 273     spam
## 3542 nonspam
## 2859 nonspam
## 4361 nonspam
## 1076    spam
## 420     spam
```

There are 57 attributes available in both the training and test data sets to predict the outcome.

```r
nrow(train_data['email'])
```

```
## [1] 700
```

```r
nrow(test_data['email'])
```

```
## [1] 300
```

There are 700 emails in training data, and 300 emails in testing data.

The outcome variable of both the dataframes is type. The type is a categorical variable which shows whether the email is spam or not spam. If the email is spam, the probability is 1, and if the email is not spam, then the probability is 0. It can be seen that the response variable type is a binary variable of 1 and 0 for different number of trials. Thus, it can be said that the outcome follows a binomial distribution.

2 c) A logistic model has been fitted to the training set using the glm() function in R.The default link function for the binomial family in R is the logit-link.

$Logit[h_{\theta(x)}] = logit[p(y = 1|x; \theta)] = \theta^T x$

```r
Trainlogistic <- glm(type~make+address+all+num3d+our+over+remove+internet+order+mail+receive+will+peopl
summary(Trainlogistic)
```

```
##
## Call:
## glm(formula = type ~ make + address + all + num3d + our + over +
##     remove + internet + order + mail + receive + will + people +
##     report + addresses + free + business + email + you + credit +
##     your + font + num000 + money + hp + hpl + george + num650 +
##     lab + labs + telnet + num857 + data + num415 + num85 + technology +
##     num1999 + parts + pm + direct + cs + meeting + original +
##     project + re + edu + table + conference + charSemicolon +
##     charRoundbracket + charSquarebracket + charExclamation +
##     charDollar + charHash + capitalAve + capitalLong + capitalTotal,
##     family = binomial, data = train_data)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.35985  -0.00556   0.00000   0.00075   2.82708
##
## Coefficients:
##                     Estimate Std. Error z value Pr(>|z|)
## (Intercept)       -3.416e+00  7.129e-01  -4.792 1.65e-06 ***
## make              -2.969e+00  1.679e+00  -1.768 0.077044 .
## address           -2.545e-01  4.405e-01  -0.578 0.563346
## all               -1.972e-01  7.599e-01  -0.260 0.795206
## num3d              3.281e+00  5.080e+01   0.065 0.948502
## our                1.855e-01  2.999e-01   0.619 0.536197
## over               2.699e+00  1.846e+00   1.462 0.143695
## remove             3.905e+00  4.124e+00   0.947 0.343682
## internet          -3.129e-01  1.076e+00  -0.291 0.771266
## order              8.627e-01  1.462e+00   0.590 0.555091
## mail               9.456e-02  2.037e-01   0.464 0.642550
## receive            9.407e-01  1.705e+00   0.552 0.581137
## will              -7.801e-02  5.405e-01  -0.144 0.885248
## people             2.006e-01  8.472e-01   0.237 0.812809
## report            -1.135e+00  1.463e+00  -0.776 0.437810
## addresses          4.185e+00  5.766e+00   0.726 0.467954
```

```
## free               3.211e+00  9.927e-01   3.234 0.001221 **
## business           6.653e+00  2.749e+00   2.420 0.015510 *
## email              6.469e-01  6.567e-01   0.985 0.324566
## you                9.309e-02  1.578e-01   0.590 0.555350
## credit             6.543e+00  7.031e+00   0.931 0.352053
## your               1.306e-01  3.436e-01   0.380 0.703949
## font              -3.933e-01  6.409e-01  -0.614 0.539385
## num000             7.973e+00  3.592e+00   2.219 0.026460 *
## money              3.597e-01  3.168e-01   1.135 0.256257
## hp                -6.202e+00  2.384e+00  -2.601 0.009297 **
## hpl                1.460e-01  7.304e-01   0.200 0.841589
## george            -2.363e+01  1.362e+01  -1.735 0.082823 .
## num650             1.325e+00  7.557e-01   1.754 0.079491 .
## lab                7.629e-01  1.541e+00   0.495 0.620672
## labs              -9.961e+01  1.072e+04  -0.009 0.992588
## telnet            -6.048e+01  1.187e+04  -0.005 0.995934
## num857            -5.397e+01  8.843e+03  -0.006 0.995130
## data              -1.136e+00  3.466e+00  -0.328 0.743094
## num415            -7.809e+01  2.215e+02  -0.353 0.724408
## num85             -7.757e+00  1.570e+02  -0.049 0.960584
## technology         2.664e+00  1.308e+00   2.037 0.041693 *
## num1999            1.900e-02  8.010e-01   0.024 0.981076
## parts              9.121e+00  7.990e+00   1.142 0.253637
## pm                -2.775e+00  1.333e+00  -2.082 0.037333 *
## direct             1.002e+01  2.578e+01   0.389 0.697476
## cs                -1.023e+02  3.141e+04  -0.003 0.997403
## meeting           -1.960e+01  8.453e+01  -0.232 0.816635
## original          -5.107e-01  3.325e+00  -0.154 0.877918
## project           -1.667e+01  1.136e+01  -1.467 0.142327
## re                -1.041e+00  6.936e-01  -1.502 0.133214
## edu               -1.967e+00  9.274e-01  -2.121 0.033889 *
## table             -1.065e+01  3.527e+01  -0.302 0.762789
## conference        -1.459e+01  1.358e+02  -0.107 0.914442
## charSemicolon      8.967e-01  3.245e+00   0.276 0.782309
## charRoundbracket  -4.523e+00  2.058e+00  -2.198 0.027979 *
## charSquarebracket  1.993e-01  2.004e+00   0.099 0.920805
## charExclamation    1.411e+00  6.980e-01   2.021 0.043292 *
## charDollar         1.392e+01  5.126e+00   2.716 0.006598 **
## charHash          -6.506e-01  6.578e+00  -0.099 0.921216
## capitalAve         8.697e-01  2.492e-01   3.490 0.000484 ***
## capitalLong       -1.729e-02  8.272e-03  -2.091 0.036571 *
## capitalTotal       9.543e-04  1.505e-03   0.634 0.526116
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 912.46  on 699  degrees of freedom
## Residual deviance: 128.21  on 642  degrees of freedom
## AIC: 244.21
##
## Number of Fisher Scoring iterations: 23
```

2 d)

It is not possible to get labels just by fitting the model and using the model parameters for estimating y. For this reason, the estimated probabilities for event per observation has been calculated, and the probabilities are classified by the below function:

$f(x) = 0, \text{ if } p(x) < 0.5$

$f(x) = 1, \text{ if } p(x) > 0.5$

```
predlog <- ifelse(predict(Trainlogistic, newdata = train_data, type = "response") >0.5, 1, 0)
```

2 d)

```
   # alternative installation of the %>%
library(magrittr) # needs to be run every time you start R and want to use %>%
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
#install.packages('kableExtra')
library(kableExtra)
```

```
##
## Attaching package: 'kableExtra'

## The following object is masked from 'package:dplyr':
##
##     group_rows
```

```
tabmat <- as.matrix(table(predlog, train_data$type))
colnames(tabmat) <- c("Label 0", "Label 1")
rownames(tabmat) <- c("Prediction 0", "Prediction 1")
kable(tabmat, caption = "Confusion matrix for the classifier on the training set")%>%
kable_styling(latex_options = c("striped", "hold_position"))
```

Table 1: Confusion matrix for the classifier on the training set

	Label 0	Label 1
Prediction 0	438	14
Prediction 1	12	236

We know,

$Accuracy = 1 - Misclassificationrate$

$Misclassificationrate = \frac{FN+FP}{TN+FN+TP+FP}$

$Misclassificationrate = \frac{12+14}{438+12+236+14}$

$Misclassificationrate = \frac{26}{700}$

$Misclassificationrate = 0.037$

$Accuracy = 1 - 0.037 = 0.963$

The accuracy of the model for train set is 0.963.

2 e)

```
predlogtest <- ifelse(predict(Trainlogistic, newdata = test_data, type = "response") >0.5, 1, 0)
```

```
tabmat <- as.matrix(table(predlogtest, test_data$type))
colnames(tabmat) <- c("Label 0", "Label 1")
rownames(tabmat) <- c("Prediction 0", "Prediction 1")
kable(tabmat, caption = "Confusion matrix for the classifier on the test set")%>%
kable_styling(latex_options = c("striped", "hold_position"))
```

Table 2: Confusion matrix for the classifier on the test set

	Label 0	Label 1
Prediction 0	143	16
Prediction 1	13	128

We know,

$Accuracy = 1 - Misclassificationrate$

$Misclassificationrate = \frac{FN+FP}{TN+FN+TP+FP}$

$Misclassificationrate = \frac{13+16}{143+13+128+16}$

$Misclassificationrate = \frac{29}{300}$

$Misclassificationrate = 0.09667$

$Accuracy = 1 - 0.09667 = 0.90333$

The accuracy of the test set is 0.9033

2 f) The accuracy for the test set is lower than the traning set because the test set has less number of observations than the train set. Test accuracy has to be reported for assessing the performance of my classifier, as this will give better estimate for the classification error probability. On the other hand, training accuracy should not be reported for assessing the performance of my classifier because the model is already trained on the training dataset and evaluating its performance on the same set will give optimistically biased result.

2 g)

Although Lasso regression has same good mean square error as Ridge regression, Lasso regression should be chosen over Ridge regression because it can perform a variable selection in the linear regression through a mechanism called lasso. The lasso uses a penalty called L1 norm of the coefficent vector, which causes the estimates of some coefficents to be exactly zero; but Ridge regression cannot set coefficients to zero. Thus, Lasso regression offers better interpretation than Ridge regression.

2 h)

```r
X <- as.matrix(train_data[,1:57])
head(X)
```

```
##      make address  all num3d  our over remove internet order mail receive will
## 273  0.25    0.25 0.00     0 0.75 0.00    0.0     0.00  0.25 0.75    0.00 1.51
## 3542 0.00    0.00 0.24     0 0.00 0.00    0.0     0.12  0.12 0.00    0.00 0.60
## 2859 0.00    0.00 0.00     0 0.00 0.00    0.0     0.00  0.00 0.00    0.00 0.00
## 4361 0.00    1.57 1.18     0 0.00 0.00    0.0     0.00  0.00 2.36    0.00 0.78
## 1076 0.00    0.55 0.55     0 1.10 0.55    2.2     0.00  0.00 0.55    0.00 0.55
## 420  0.51    0.43 0.29     0 0.14 0.03    0.0     0.18  0.54 0.62    0.29 0.65
##      people report addresses free business email  you credit your font num000
## 273    0.00   1.26      0.00 0.00     0.50  0.00 3.29      0 1.01 0.00   0.00
## 3542   0.12   0.12      0.00 0.00     0.72  0.00 0.00      0 0.00 0.00   0.00
## 2859   0.00   0.00      0.00 0.00     0.00  0.00 0.00      0 0.00 0.00   0.00
## 4361   0.00   0.00      0.00 0.00     0.00  0.00 0.39      0 0.00 6.29   0.00
## 1076   0.00   0.00      0.00 0.00     0.00  0.55 3.31      0 1.10 0.00   0.00
## 420    0.65   1.20      0.03 0.21     0.43  0.03 3.03      0 1.35 0.00   0.51
##      money   hp hpl george num650 lab labs telnet num857 data num415 num85
## 273   0.00 0.00   0   0.00      0   0    0      0      0 0.25      0     0
## 3542  0.00 1.81   0   0.00      0   0    0      0      0 0.00      0     0
## 2859  0.00 0.00   0   1.17      0   0    0      0      0 0.00      0     0
## 4361  0.00 0.00   0   0.00      0   0    0      0      0 0.00      0     0
## 1076  0.00 0.00   0   0.00      0   0    0      0      0 0.00      0     0
## 420   0.54 0.00   0   0.00      0   0    0      0      0 0.00      0     0
##      technology num1999 parts pm direct cs meeting original project   re edu
## 273        0.00    0.00     0  0      0  0       0        0       0 0.00   0
## 3542       0.12    0.12     0  0      0  0       0        0       0 0.00   0
## 2859       0.00    0.00     0  0      0  0       0        0       0 0.00   0
## 4361       0.00    0.00     0  0      0  0       0        0       0 0.00   0
## 1076       0.00    0.00     0  0      0  0       0        0       0 0.55   0
## 420        0.00    0.00     0  0      0  0       0        0       0 0.03   0
##      table conference charSemicolon charRoundbracket charSquarebracket
## 273      0          0         0.000            0.082             0.000
## 3542     0          0         0.105            0.060             0.000
## 2859     0          0         0.000            0.186             0.186
## 4361     0          0         1.151            0.203             0.000
## 1076     0          0         0.000            0.165             0.000
## 420      0          0         0.012            0.078             0.000
##      charExclamation charDollar charHash capitalAve capitalLong capitalTotal
## 273            0.041      0.124    0.124      3.181          32          210
## 3542           0.000      0.000    0.000      1.827          23          466
## 2859           0.000      0.000    0.000      3.862          28          112
## 4361           0.271      0.000    0.067      5.689          30          330
## 1076           0.496      0.000    0.082     16.826         148          387
## 420            0.443      0.510    0.133      6.590         739         2333
```

```r
y <- train_data[,58]
head(y)
```

```
## [1] spam    nonspam nonspam nonspam spam    spam
## Levels: nonspam spam
```

2 h)

```r
#install.packages("glmnet")
library(glmnet)
```

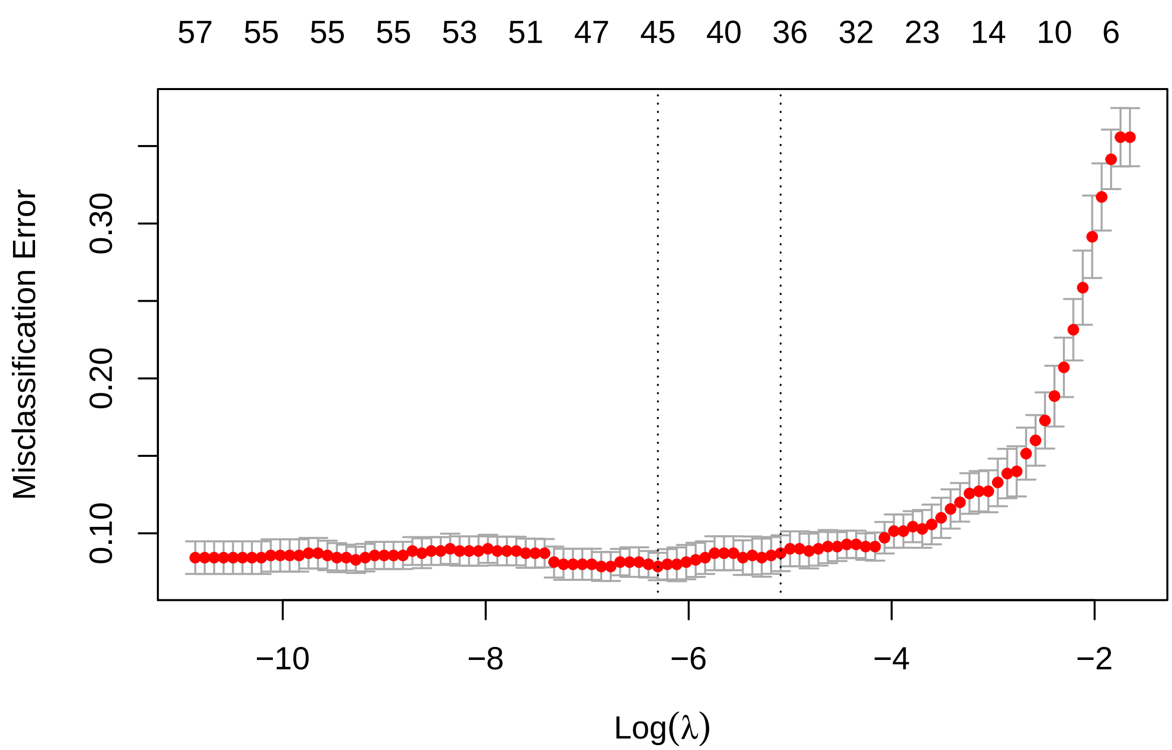
```
## Loading required package: Matrix

## Loaded glmnet 4.1-1
```

```r
lasso.cv = cv.glmnet(X, y, family = "binomial", type.measure = "class",alpha=1)
lasso.cv
```

```
##
## Call:  cv.glmnet(x = X, y = y, type.measure = "class", family = "binomial",      alpha = 1)
##
## Measure: Misclassification Error
##
##       Lambda Index Measure       SE Nonzero
## min 0.001830    51 0.07857 0.008845      45
## 1se 0.006134    38 0.08714 0.011557      37
```

```r
plot(lasso.cv)
```



2 i)

```r
lasso <- glmnet(X,y,lambda=lasso.cv$lambda.1se,alpha=1,family="binomial")
```

```r
colnames(X)[lasso$beta[,1]!=0]
```

```
##  [1] "address"          "num3d"            "our"              "over"
##  [5] "remove"           "internet"         "mail"             "will"
##  [9] "people"           "report"           "addresses"        "free"
## [13] "business"         "email"            "your"             "font"
## [17] "num000"           "money"            "hp"               "hpl"
## [21] "george"           "labs"             "data"             "num1999"
## [25] "pm"               "cs"               "meeting"          "project"
## [29] "re"               "edu"              "table"            "conference"
## [33] "charRoundbracket" "charExclamation"  "charDollar"       "capitalLong"
## [37] "capitalTotal"
```

There are still 47 attributes in the model.

2 j)

```r
X1 <- as.matrix(test_data[,1:57])
head(X1)
```

```
##      make address  all num3d  our over remove internet order mail receive will
## 1570 0.09    0.09 1.14     0 0.38 0.00      0     0.09  0.00 0.19    0.38 0.19
## 2338 0.00    0.00 0.00     0 0.00 0.00      0     0.00  0.00 0.00    0.00 1.11
## 3278 0.00    0.00 0.33     0 0.00 0.49      0     1.32  0.16 5.12    0.00 0.00
## 2776 0.00    0.00 0.00     0 0.00 0.00      0     0.00  0.00 0.84    0.00 0.00
## 426  0.33    0.00 0.66     0 0.22 0.00      0     0.00  0.44 0.11    0.00 0.33
## 3417 0.00    0.00 0.00     0 0.00 0.49      0     0.49  0.00 0.00    0.00 0.00
##      people report addresses free business email  you credit your font num000
## 1570      0   0.00         0 0.66     0.00     0 1.52      0 1.42    0      0
## 2338      0   0.00         0 0.00     0.00     0 0.00      0 0.00    0      0
## 3278      0   0.66         0 0.00     0.33     0 0.33      0 0.00    0      0
## 2776      0   0.00         0 0.00     0.84     0 1.68      0 0.00    0      0
## 426       0   0.00         0 0.55     0.00     0 1.76      0 1.10    0      0
## 3417      0   0.00         0 0.00     0.00     0 0.49      0 0.00    0      0
##      money   hp  hpl george num650 lab labs telnet num857 data num415 num85
## 1570  0.00 0.00 0.00   0.00      0   0    0      0      0    0      0  0.00
## 2338  0.00 1.11 1.11   0.00      0   0    0      0      0    0      0  0.00
## 3278  0.00 0.00 0.00   0.00      0   0    0      0      0    0      0  0.16
## 2776  0.00 0.00 0.00   0.00      0   0    0      0      0    0      0  0.00
## 426   0.22 0.00 0.00   0.00      0   0    0      0      0    0      0  0.00
## 3417  0.00 0.00 0.00   0.49      0   0    0      0      0    0      0  0.00
##      technology num1999 parts   pm direct cs meeting original project   re  edu
## 1570          0    0.00  0.00 0.00      0  0    0.00     0.38       0 0.00 0.00
## 2338          0    0.00  0.00 0.00      0  0    0.00     0.00       0 0.00 0.00
## 3278          0    0.00  0.00 0.00      0  0    0.16     0.00       0 0.00 0.33
## 2776          0    0.84  0.00 0.84      0  0    0.00     0.84       0 0.84 0.84
## 426           0    0.00  0.11 0.00      0  0    0.00     0.11       0 0.00 0.00
## 3417          0    0.49  0.00 0.00      0  0    0.00     0.00       0 0.00 0.00
##      table conference charSemicolon charRoundbracket charSquarebracket
## 1570     0          0         0.044            0.059             0.000
```

```
## 2338     0          0         0.000            0.183             0.000
## 3278     0          0         0.000            0.070             0.023
## 2776     0          0         0.000            0.000             0.137
## 426      0          0         0.000            0.173             0.000
## 3417     0          0         0.000            0.228             0.000
##       charExclamation charDollar charHash capitalAve capitalLong capitalTotal
## 1570            0.591      0.000    0.000      3.280          31          771
## 2338            0.000      0.000    0.000      1.800           4           36
## 3278            0.000      0.000    0.023      1.552          10          149
## 2776            0.413      0.000    0.137      3.052          13          116
## 426             0.367      0.193    0.077      2.559          75          389
## 3417            0.000      0.000    0.000      1.962           5          106
```

```
g <- predict(lasso, newdata = test_data, newx=X1, type = "response")
```

```
predlogtest1 <- ifelse(predict(lasso, newdata = test_data, newx=X1, type = "response") >0.5, 1, 0)
```

```
#install.packages("magrittr") # package installations are only needed the first time you use it
#install.packages("dplyr")    # alternative installation of the %>%
library(magrittr) # needs to be run every time you start R and want to use %>%
library(dplyr)
```

```
#install.packages('kableExtra')
library(kableExtra)
```

```
tabmat <- as.matrix(table(predlogtest1, test_data$type))
colnames(tabmat) <- c("Label 0", "Label 1")
rownames(tabmat) <- c("Prediction 0", "Prediction 1")
kable(tabmat, caption = "Confusion matrix for the classifier on the test set")%>%
kable_styling(latex_options = c("striped", "hold_position"))
```

Table 3: Confusion matrix for the classifier on the test set

	Label 0	Label 1
Prediction 0	145	19
Prediction 1	11	125

We know,

$Accuracy = 1 - Misclassificationrate$

$Misclassificationrate = \frac{FN+FP}{TN+FN+TP+FP}$

$Misclassificationrate = \frac{13+14}{143+13+130+14}$

$Misclassificationrate = \frac{27}{300}$

$Misclassificationrate = 0.09$

$Accuracy = 1 - 0.09 = 0.91$

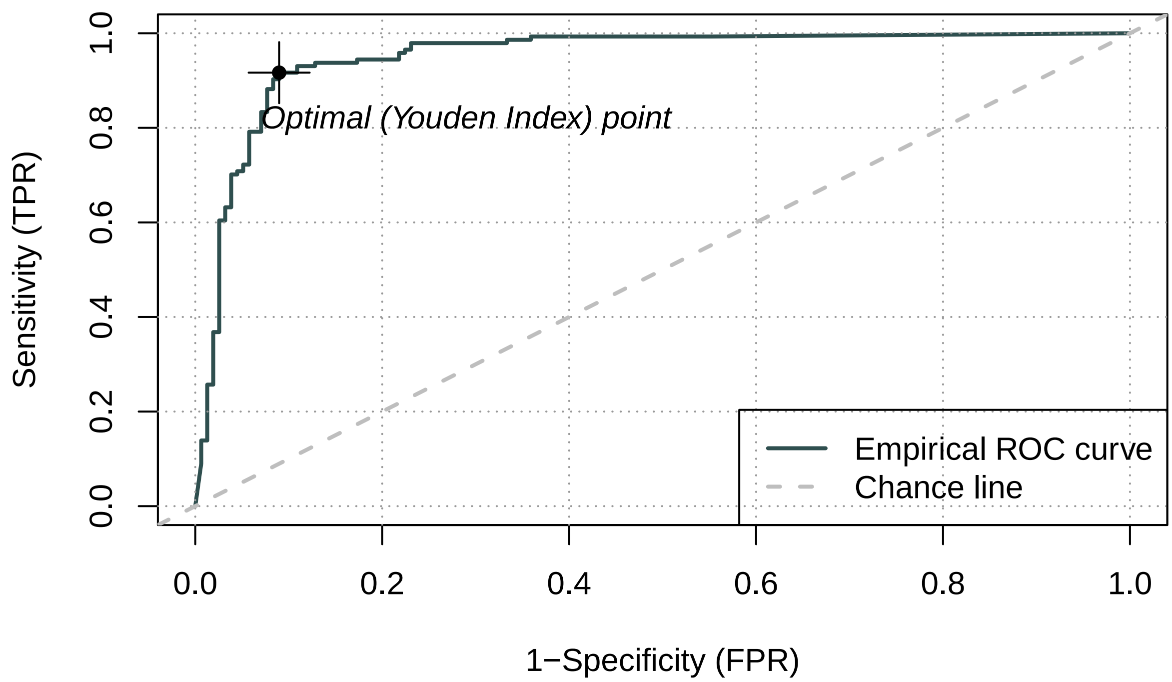
The accuracy on the test set is 0.91.

2 k)

Drawing the ROC curve for the logistic model

```
y_test.pred <- predict(Trainlogistic, newdata = test_data, type = "response")
```

```
#install.packages("ROCit")
library(ROCit)
roc.model1 <- rocit(score=y_test.pred,class=test_data$type)
plot(roc.model1)
```



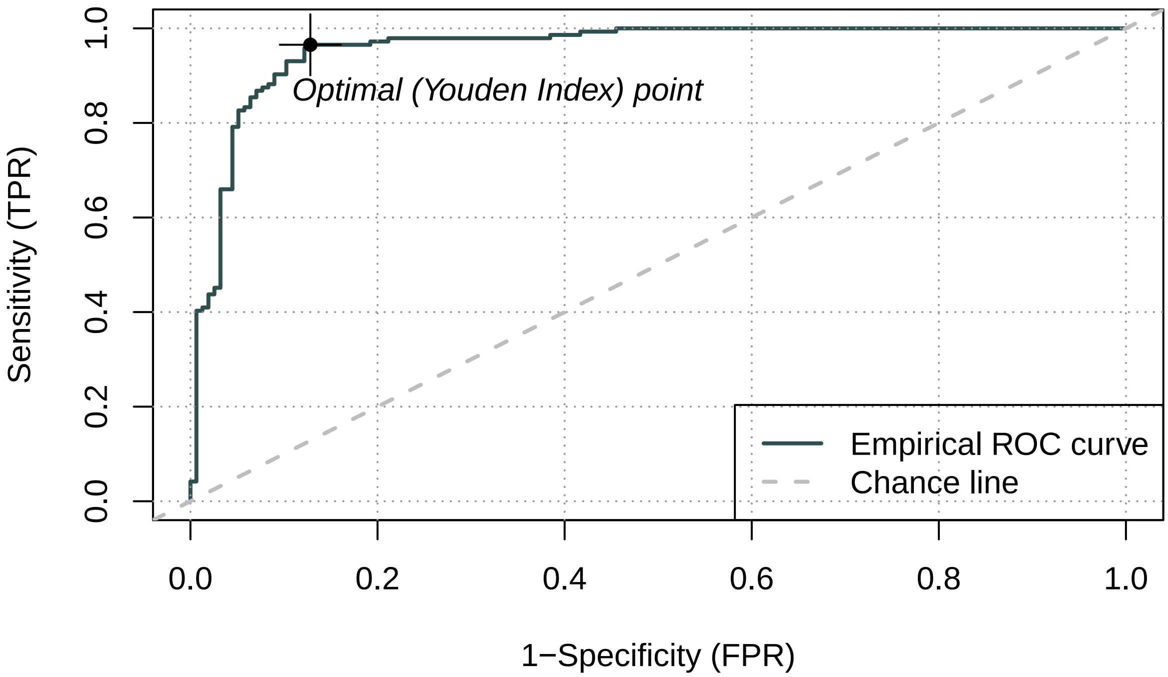
Drawing the ROC curve for the penalized model

```
g <- predict(lasso, newdata = test_data, newx=X1, type = "response")
```

```
head(g[,1])
```

```
##      1570      2338      3278      2776       426      3417
## 0.5672785 0.0324291 0.5319915 0.1630774 0.6970251 0.2125926
```

```
library(ROCit)
roc.model2 <- rocit(class=test_data$type,score=g[,1])
plot(roc.model2)
```



2 l)

```
roc.model1$AUC
```

```
## [1] 0.9498531
```

```
roc.model2$AUC
```

```
## [1] 0.9579772
```

The AUC of logistic model is 0.9498 and the AUC of lasso model is 0.9571314. The AUC of lasso model is higer than logistic model. Higher AUC means the model is better in terms of predictive ability. Thus, I would prefer lasso model rather than logistic model.