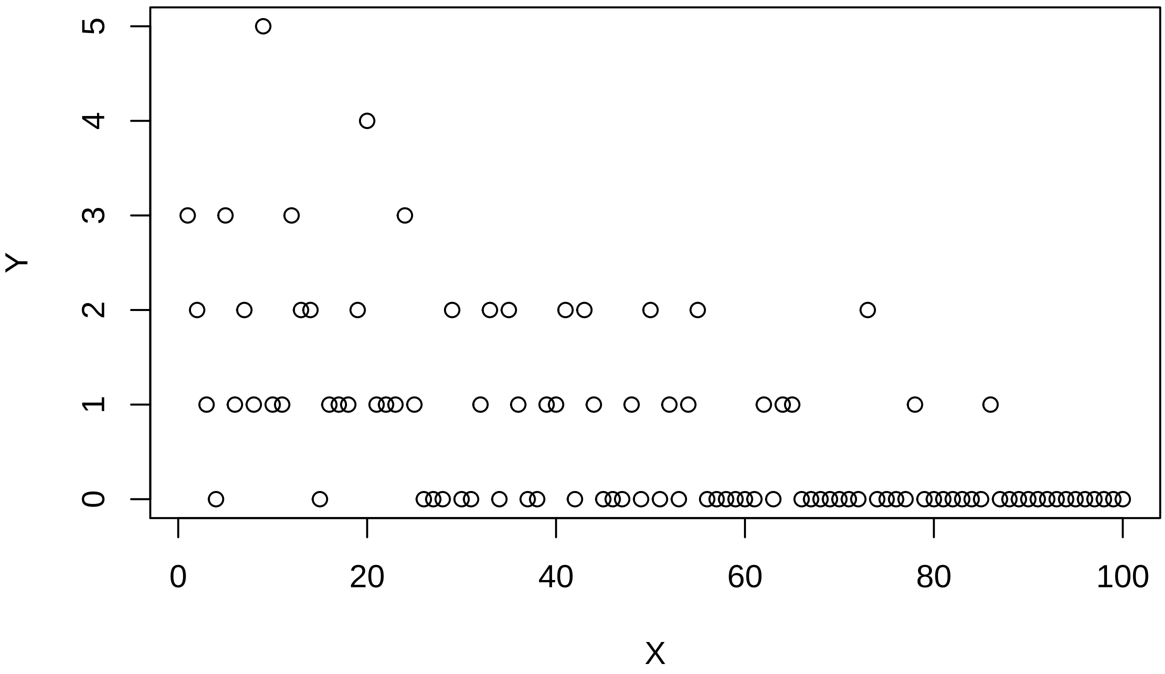
HW576

```
##4
#a
X <- 1:100
lambdagen <- exp(1-0.04*X)
Y <-as.vector(rpois(X, lambda = lambdagen))
pdata <- as.data.frame(cbind(X, Y))
plot(X, Y)
```



```
#b
X <- cbind(1, X)
beta <- matrix(rep(0, ncol(X)), nrow = ncol(X), ncol = 1) #initialize beta == 0
i <- 1 #counting iteration
maxi = nrow(X)
ones <- c(rep(1,100))
while(i <= maxi){
  eta <- X %*% beta
  mu <- exp(eta)
  V <- as.vector(mu)
  w <- diag(V)
  z <- eta + diag(ones/V)%*%(Y - mu)
  blast <- beta
  beta <- solve(t(X)%*%w%*%X)%*%t(X)%*%w%*%z
  if(abs(beta[2,] - blast[2,]) <= 1e-4)
    break
  i <- i + 1
}

fit <- glm(Y~X[,2], family = "poisson")
summary(fit)
```

```
##
## Call:
## glm(formula = Y ~ X[, 2], family = "poisson")
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.0859  -0.8007  -0.5187   0.3488   2.2604
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.907530   0.182826   4.964 6.91e-07 ***
## X[, 2]      -0.032557   0.005137  -6.338 2.33e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 136.900  on 99  degrees of freedom
## Residual deviance:  86.467  on 98  degrees of freedom
## AIC: 193.16
##
## Number of Fisher Scoring iterations: 5
```

```
#c
fisherin <- solve(t(X)%*%w%*%X)
fisherin
```

```
##                               X
##    0.0334254039 -7.180185e-04
## X -0.0007180185  2.638848e-05
```

b

The interation times is 5, which is also the result from the glm routine in R. The beta from the iteration procedure is the following: [,1] 0.95538073 X -0.03838775 which is also very close to the result from the glm routine in R

c

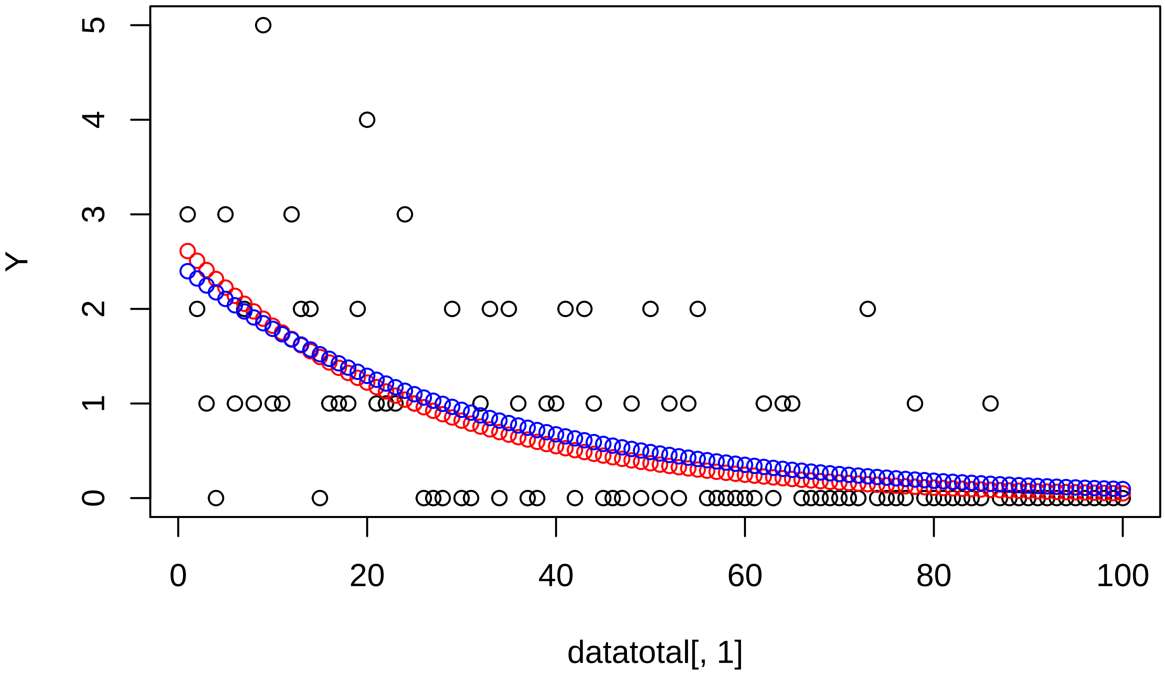
the fisher information matrix is the following: fisherin X 0.0354906454 -8.254298e-04 X -0.0008254298 3.388128e-05

```
varb0b1 <- 0.0354906454 + 3.388128e-05 - 2*8.254298e-04
upper.list <- unlist(lapply(X, function(X) exp(0.95538073 - 0.03838775*X + qt(0.975, 98)*sqrt(varb0b1)))
lower.list <- unlist(lapply(X, function(X) exp(0.95538073 - 0.03838775*X - qt(0.975, 98)*sqrt(varb0b1)))
CI.matrix <- cbind(upper.list, lambdagen ,lower.list)
CI.matrix[101:200,] #I don't know why my matrix also includes the 100 intervals of x0
```

```
##        upper.list  lambdagen lower.list
##   [1,] 3.60467008 2.61169647 1.73629878
##   [2,] 3.46891720 2.50929039 1.67090928
##   [3,] 3.33827682 2.41089971 1.60798237
##   [4,] 3.21255638 2.31636698 1.54742531
##    [5,] 3.09157061 2.22554093 1.48914884
##    [6,] 2.97514120 2.13827622 1.43306708
##    [7,] 2.86309655 2.05443321 1.37909738
##    [8,] 2.75527154 1.97387773 1.32716018
##    [9,] 2.65150725 1.89648088 1.27717896
##   [10,] 2.55165074 1.82211880 1.22908005
##   [11,] 2.45555487 1.75067250 1.18279255
##   [12,] 2.36307799 1.68202765 1.13824825
##   [13,] 2.27408381 1.61607440 1.09538151
##   [14,] 2.18844118 1.55270722 1.05412913
##   [15,] 2.10602387 1.49182470 1.01443034
##   [16,] 2.02671043 1.43332941 0.97622661
##   [17,] 1.95038395 1.37712776 0.93946164
##   [18,] 1.87693195 1.32312981 0.90408125
##   [19,] 1.80624617 1.27124915 0.87003330
##   [20,] 1.73822243 1.22140276 0.83726761
##   [21,] 1.67276049 1.17351087 0.80573587
##   [22,] 1.60976386 1.12749685 0.77539164
##   [23,] 1.54913971 1.08328707 0.74619017
##   [24,] 1.49079868 1.04081077 0.71808844
##   [25,] 1.43465478 1.00000000 0.69104503
##   [26,] 1.38062528 0.96078944 0.66502008
##   [27,] 1.32863055 0.92311635 0.63997524
##   [28,] 1.27859395 0.88692044 0.61587359
##   [29,] 1.23044175 0.85214379 0.59267962
##   [30,] 1.18410297 0.81873075 0.57035914
##   [31,] 1.13950932 0.78662786 0.54887926
##   [32,] 1.09659508 0.75578374 0.52820831
##   [33,] 1.05529701 0.72614904 0.50831584
##   [34,] 1.01555423 0.69767633 0.48917252
##   [35,] 0.97730817 0.67032005 0.47075015
##   [36,] 0.94050247 0.64403642 0.45302157
##   [37,] 0.90508289 0.61878339 0.43596065
##   [38,] 0.87099721 0.59452055 0.41954225
##   [39,] 0.83819522 0.57120906 0.40374217
##   [40,] 0.80662855 0.54881164 0.38853713
##   [41,] 0.77625070 0.52729242 0.37390471
##   [42,] 0.74701688 0.50661699 0.35982336
##   [43,] 0.71888401 0.48675226 0.34627231
##   [44,] 0.69181064 0.46766643 0.33323160
##   [45,] 0.66575686 0.44932896 0.32068200
##   [46,] 0.64068427 0.43171052 0.30860503
##   [47,] 0.61655592 0.41478291 0.29698288
##   [48,] 0.59333626 0.39851904 0.28579842
##   [49,] 0.57099105 0.38289289 0.27503517
##   [50,] 0.54948736 0.36787944 0.26467727
##   [51,] 0.52879352 0.35345468 0.25470945
##   [52,] 0.50887901 0.33959553 0.24511702
##   [53,] 0.48971448 0.32627979 0.23588585
##   [54,] 0.47127170 0.31348618 0.22700232
##   [55,] 0.45352347 0.30119421 0.21845335
##   [56,] 0.43644365 0.28938422 0.21022634
##   [57,] 0.42000706 0.27803730 0.20230915
##   [58,] 0.40418948 0.26713530 0.19469013
##  [59,] 0.38896759 0.25666078 0.18735805
##  [60,] 0.37431896 0.24659696 0.18030209
##  [61,] 0.36022201 0.23692776 0.17351187
##  [62,] 0.34665595 0.22763769 0.16697736
##  [63,] 0.33360079 0.21871189 0.16068895
##  [64,] 0.32103729 0.21013607 0.15463736
##  [65,] 0.30894693 0.20189652 0.14881367
##  [66,] 0.29731191 0.19398004 0.14320931
##  [67,] 0.28611506 0.18637398 0.13781600
##  [68,] 0.27533988 0.17906615 0.13262581
##  [69,] 0.26497051 0.17204486 0.12763109
##  [70,] 0.25499164 0.16529889 0.12282447
##  [71,] 0.24538859 0.15881743 0.11819886
##  [72,] 0.23614719 0.15259011 0.11374746
##  [73,] 0.22725382 0.14660696 0.10946370
##  [74,] 0.21869537 0.14085842 0.10534127
##  [75,] 0.21045925 0.13533528 0.10137409
##  [76,] 0.20253329 0.13002871 0.09755631
##  [77,] 0.19490583 0.12493021 0.09388231
##  [78,] 0.18756562 0.12003163 0.09034668
##  [79,] 0.18050185 0.11532512 0.08694420
##  [80,] 0.17370410 0.11080316 0.08366985
##  [81,] 0.16716236 0.10645850 0.08051882
##  [82,] 0.16086697 0.10228421 0.07748646
##  [83,] 0.15480868 0.09827359 0.07456830
##  [84,] 0.14897854 0.09442022 0.07176004
##  [85,] 0.14336797 0.09071795 0.06905753
##  [86,] 0.13796869 0.08716085 0.06645681
##  [87,] 0.13277275 0.08374323 0.06395403
##  [88,] 0.12777249 0.08045961 0.06154550
##  [89,] 0.12296054 0.07730474 0.05922768
##  [90,] 0.11832982 0.07427358 0.05699715
##  [91,] 0.11387348 0.07136127 0.05485062
##  [92,] 0.10958498 0.06856315 0.05278493
##  [93,] 0.10545797 0.06587475 0.05079704
##  [94,] 0.10148640 0.06329177 0.04888400
##  [95,] 0.09766439 0.06081006 0.04704302
##  [96,] 0.09398632 0.05842567 0.04527137
##  [97,] 0.09044677 0.05613476 0.04356643
##  [98,] 0.08704052 0.05393369 0.04192571
##  [99,] 0.08376255 0.05181892 0.04034678
## [100,] 0.08060803 0.04978707 0.03882731
```

From the matrix above, we saw the 95% intervals includes the true value of lambda

```
#d
datatotal <- cbind(X[,2], Y, lambdagen, mu)
plot(datatotal[,1], Y)
points(datatotal[,1], lambdagen, col="red")
points(datatotal[,1], mu, col="blue")
```



The estimated values and the true means are very similar.