Homework9

```r
# homework9
rate <- read.csv("/Users/raymond/Desktop/R/course/data/rate.csv", header =
TRUE)
rate

# creating data with dummy variables
(ratema <- as.data.frame(rate))
for (j in 2 : 12) {
    for (i in 1 : length(rate$Age)) {
        if (ratema[i, 1] == j) {
            ratema[i, j + 3] <- 1
        } else {
            ratema[i, j + 3] <- 0
        }
    }
}
ratema
```

	Age	PY	Death	sex	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15
1	1	1299868	55	m	0	0	0	0	0	0	0	0	0	0	0
2	2	1240595	49	m	1	0	0	0	0	0	0	0	0	0	0
3	3	1045453	38	m	0	1	0	0	0	0	0	0	0	0	0
4	4	795776	26	m	0	0	1	0	0	0	0	0	0	0	0
5	5	645991	19	m	0	0	0	1	0	0	0	0	0	0	0
6	6	599729	17	m	0	0	0	0	1	0	0	0	0	0	0
7	7	568109	22	m	0	0	0	0	0	1	0	0	0	0	0
8	8	506475	21	m	0	0	0	0	0	0	1	0	0	0	0
9	9	368751	18	m	0	0	0	0	0	0	0	1	0	0	0
10	10	252581	11	m	0	0	0	0	0	0	0	0	1	0	0
11	11	140053	10	m	0	0	0	0	0	0	0	0	0	1	0
12	12	81850	4	m	0	0	0	0	0	0	0	0	0	0	1
13	1	1300402	37	f	0	0	0	0	0	0	0	0	0	0	0
14	2	1217896	29	f	1	0	0	0	0	0	0	0	0	0	0
15	3	1045801	23	f	0	1	0	0	0	0	0	0	0	0	0
16	4	810260	12	f	0	0	1	0	0	0	0	0	0	0	0
17	5	665612	7	f	0	0	0	1	0	0	0	0	0	0	0
18	6	633646	12	f	0	0	0	0	1	0	0	0	0	0	0
19	7	650686	9	f	0	0	0	0	0	1	0	0	0	0	0
20	8	600455	19	f	0	0	0	0	0	0	1	0	0	0	0
21	9	474609	13	f	0	0	0	0	0	0	0	1	0	0	0
22	10	376781	14	f	0	0	0	0	0	0	0	0	1	0	0
23	11	255412	5	f	0	0	0	0	0	0	0	0	0	1	0
24	12	213603	3	f	0	0	0	0	0	0	0	0	0	0	1

```r
# use NR method to find MLE of the coefficeints of Poison regression
y <- rate$Death
x <- cbind(rep(1, length(y)),
           ratema$V5,
           ratema$V6,
           ratema$V7,
           ratema$V8,
           ratema$V9,
           ratema$V10,
           ratema$V11,
           ratema$V12,
           ratema$V13,
           ratema$V14,
           ratema$V15,
           ifelse(ratema$sex == "m", 1, 0))


ftn <- function(betacoef) {
    mu <- exp(x %*% betacoef + log(ratema$PY / 100000))
    gradient <- t(x) %*% (y - mu)
    hessian <- -t(x) %*% diag(c(mu), length(y)) %*% x
    loglike <- sum(-mu + y * log(mu) - log(factorial(y)))

    return(list(gradient, hessian, loglike))
}


highnew <- function(ftn, x0, tol, maxiter) {
    x <- x0
    fx <- ftn(x)
    iter <- 0

    while ((max(abs(fx[[1]])) > tol) && (iter < maxiter)) {
        x <- x - (solve(fx[[2]]) %*% fx[[1]])
        fx <- ftn(x)
        iter <- iter + 1
    }
```

```r
    if (max(abs(fx[[1]])) > tol) {
        cat("algorithm failed to converge\n")
        return(NULL)
    } else {
        cat("algorithm converges to\n")
        return(x)
    }
}

(beta <- highnew(ftn, x0 = c(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0), tol = 1e-
9, maxiter = 100))#nolint
write.csv(beta, file = "/Users/raymond/Desktop/R/course/beta.csv")
```

beta

	V1
1	0.944535338554194
2	-0.111587604998041
3	-0.193073956584897
4	-0.39994534890244
5	-0.57528753380895
6	-0.40114659844663
7	-0.311421036338812
8	0.0445282799251619
9	0.0730132968193308
10	0.171117692685545
11	0.152563707400858
12	-0.271157265961863
13	0.560645252097408

```r
#variance covariance matrix
(vcov <- solve(-ftn(beta)[[2]]))
vcov(model)
write.csv(vcov, file = "/Users/raymond/Desktop/R/course/vcov.csv")
```

vcov

	V1	V2	V3	V4	V5	V6	V7
1	0.0145095991081799	-0.0108446534059454	-0.010869462131085	-0.0108929432598965	-0.0109087666055999	-0.0109423347081131	-0.0110516743426813
2	-0.0108446534059454	0.0236902485303688	0.0108695659228976	0.0108694052217967	0.0108692969293421	0.0108690671946019	0.0108683188915801
3	-0.010869462131085	0.0108695659228976	0.0272630078432615	0.0108695645553216	0.0108695641072021	0.0108695631565484	0.0108695600600335
4	-0.0108929432598965	0.0108694052217967	0.0108695645553216	0.0371855048360708	0.0108698169874977	0.010870032577943	0.0108707348094884
5	-0.0109087666055999	0.0108692969293421	0.0108695641072021	0.0108698169874977	0.0493315258587456	0.0108703489092857	0.0108715264437785
6	-0.0109423347081131	0.0108690671946019	0.0108695631565484	0.0108700325779429	0.0108703489092857	0.0453537786044207	0.010873205839622
7	-0.0110516743426813	0.0108683188915801	0.0108695600600335	0.0108707348094883	0.0108715264437785	0.010873205839622	0.0431367405644931
8	-0.0110990020736223	0.0108679949881372	0.0108695587197051	0.0108710387708631	0.0108720361403388	0.0108741519887461	0.010881043831726
9	-0.011212934339205	0.0108672152538812	0.0108695554931262	0.0108717704984864	0.0108732631352258	0.0108764296579085	0.0108867438075642
10	-0.0114209388402335	0.0108657917042481	0.0108695496024075	0.0108731064030994	0.0108755032421695	0.0108805879657875	0.0108971501715913
11	-0.0117076984365923	0.0108638291673023	0.0108695414813327	0.0108749481106861	0.0108785915030475	0.0108863207008149	0.0109114966147367
12	-0.0122126438198355	0.0108603734016267	0.0108695271812036	0.010878191111984	0.010884029518348	0.010896415282207	0.0109367587866891
13	-0.00571876666888036	-3.91383270686056E-05	-1.61956330664126E-07	3.67286608514288E-05	6.15883255450508E-05	0.000114326336110233	0.000286107115223002

V8	V9	V10	V11	V12	V13
-0.0110990020736223	-0.011212934339205	-0.0114209388402335	-0.0117076984365923	-0.0122126438198355	-0.00571876666888036
0.0108679949881372	0.0108672152538812	0.0108657917042481	0.0108638291673023	0.0108603734016267	-3.91383270686061E-05
0.0108695587197051	0.0108695554931262	0.0108695496024075	0.0108695414813327	0.0108695271812036	-1.6195633066461E-07
0.0108710387708631	0.0108717704984864	0.0108731064030994	0.0108749481106861	0.010878191111984	3.67286608514283E-05
0.0108720361403388	0.0108732631352258	0.0108755032421695	0.0108785915030475	0.010884029518348	6.15883255450501E-05
0.0108741519887461	0.0108764296579085	0.0108805879657875	0.0108863207008149	0.010896415282207	0.000114326336110232
0.010881043831726	0.0108867438075641	0.0108971501715913	0.0109114966147367	0.0109367587866891	0.000286107115223001
0.0358840269703795	0.0108912082938569	0.0109043191329946	0.0109223940252227	0.0109542214984759	0.000360462535622784
0.0108912082938569	0.0431600201884866	0.0109215770069369	0.0109486274118858	0.0109962595653245	0.00053945868304137
0.0109043191329946	0.0109215770069369	0.0509530844595433	0.010996521329164	0.0110730078602374	0.000866249378718386
0.0109223940252227	0.0109486274118858	0.010996521329164	0.077729215607343	0.0111788147536539	0.00131677024496317
0.0109542214984759	0.0109962595653245	0.0110730078602374	0.0111788147536539	0.154222269430957	0.00211007737174673
0.000360462535622785	0.000539458683041371	0.000866249378718387	0.00131677024496317	0.00211007737174673	0.00898461201030506

```r
#loglike
(loglike <- ftn(beta)[[3]])
write.csv(loglike, file = "/Users/raymond/Desktop/R/course/loglike.csv")
```

loglike

	x
1	-59.3896572222313