HW 6.

1) (1) $y_i = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + \varepsilon_i$

$i = 1, \dots n$

$\varepsilon_n \overset{iid}{\sim} N(0, \sigma^2)$.

pdf $= f(y_i) = \dfrac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2\sigma^2}[y_i - \hat{y}_i]^2}$

$(\hat{y}_i = \beta_0 + \beta_1 x_1 + \beta_2 x_{2i}$

(2) Joint distribution. $f(y_1, y_2, \dots y_n) = f(y_1) \cdot f(y_2) \dots f(y_n)$

$f(y_1, y_2, \dots y_n) = \prod_{i=1}^{n} \dfrac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2\sigma^2}[y_i - \hat{y}_i]^2}$

$\because$ sampled independently & randomly.

$= \left(\dfrac{1}{\sigma\sqrt{2\pi}}\right)^n \prod_{i=1}^{n} e^{-\frac{1}{2\sigma^2}(y_i - \hat{y}_i)^2}$

$= \left(\dfrac{1}{\sigma\sqrt{2\pi}}\right)^n \left[e^{-\frac{1}{2\sigma^2}(y_1-\hat{y}_1)^2} \cdot e^{-\frac{1}{2\sigma^2}(y_2-\hat{y}_2)^2} \cdot\times\cdot e^{-\frac{1}{2\sigma^2}(y_n-\hat{y}_n)^2} \right]$

$= \left(\dfrac{1}{\sigma\sqrt{2\pi}}\right)^n \cdot \exp\left[-\dfrac{1}{2\sigma^2} \sum_{i=1}^{n} [y_i - [\beta_0 + \beta_1 x_i + \beta_2 x_{2i}]^2 \right]$

(3) Likelihood function maximizing L is $\equiv$ max $\log(L)$.

$L(\beta_0, \beta_1, \beta_2, \sigma) = \left(\dfrac{1}{\sqrt{2\pi}\,\sigma}\right)^n \cdot \exp\left[\dfrac{-1}{2\sigma^2} \sum_{i=1}^{n} [y_i - [\beta_0 + \beta_1 x_{1i} + \beta_2 {}_i]]^2 \right]$

(4) Log (L) $= l(\beta_0, \beta_1, \beta_2, \sigma)$.

$\log(L(\beta_0, \beta_1, \beta_2, \sigma)) = \log\left[\left(\dfrac{1}{\sqrt{2\pi}}\right)^n \cdot \left(\dfrac{1}{\sigma}\right)^n \cdot \exp\left[\dfrac{-1}{2\sigma^2} \sum_{i=1}^{n} [y_i - [\dots]^2]\right]\right]$

$= -n\log(\sqrt{2\pi}) - n\log\sigma - \dfrac{1}{2\sigma^2} \sum_{i=1}^{n} [y_i - (\beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i})$

$-\ell(\beta_0, \beta_1, \beta_2, \sigma)$.

$$-\log(L) = n\log(\sqrt{2\pi}) + n\log\sigma + \frac{1}{2\sigma^2}\sum_{i=1}^{n}\left[y_i - (\beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i})\right]^2$$

(5) Let estimates be, $\hat{\beta}_0, \hat{\beta}_1, \hat{\beta}_2, \hat{\sigma}$ from the process we want to maximize the likelihood i.e the probability of observing the sampled data given the estimated parameters.

i.e $\arg\max L(\beta_0, \beta_1, \beta_2, \sigma)$

$\because$ log is a monotonic function (increasing), maximizing $L = \max \log(L)$ & same as minimizing negative of it

$\therefore$ $\arg\max L(\beta_0, \beta_1, \beta_2, \sigma) = \arg\max \ell(\beta_0, \beta_1, \beta_2, \sigma)$.
$= \arg\min -\ell(\beta_0, \beta_1, \beta_2, \sigma)$.

Using log we convert (Π) to (Σ) & thus one value changing to 0 does not straight forward. makes everything 0 and simplifies computation.

(6) Consider. $\arg\min \; n\log(\sqrt{2\pi}) + n\log\sigma + \frac{1}{2\sigma^2}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$.

$$\frac{d}{d\beta}(-\ell(\beta)) = \frac{d}{d\beta}\left(\frac{1}{2\sigma^2}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2\right) \quad \text{—} \; \text{I}$$

$\because$ terms without $\beta \Rightarrow 0$.

I is similar to the form of equation for Least Squares

$$LS = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 = \sum_{i=1}^{n}(y_i - (\beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i}))^2.$$

from I (only considering terms with $\beta_0, \beta_1, \beta_2$). & above derivations for $-\ell(\beta)$ & LS equation. we can say that the method of least squares is same as method of maximum likelihood. estimation.

$\therefore$ Both methods give the same estimates for $\hat{\beta}_0, \hat{\beta}_1, \hat{\beta}_2$

Q2) $Y \sim \text{Bernoulli}(p)$, x_{i1} = student, x_{i2} = balance, x_{i3} = income

$\therefore$ pmf : $f(y) = p^{y} \cdot (1-p)^{y}$. , $y \in \{0,1\}$.

$f(y_i) = p_i^{y_i}(1-p_i)^{y_i}$ for each customer y_i (or each instance)

$y_i = 0, 1$ & $i = 1, 2, \ldots n$.

for logistic regression,

$$p_i = P(Y_i = 1) = \frac{1}{1+e^{-x_i^T \beta}}.$$

$$\& \quad P(Y_i = 0) = 1 - p_i = 1 - \frac{1}{1+e^{-x_i^T \beta}}.$$

$$= \frac{e^{-x_i^T \beta}}{1+e^{-x_i^T \beta}}.$$

where $x_i^T \beta = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \beta_3 x_{i3}$ & $y = y_i$

(1) $\therefore$ pmf = $\underline{f(y_i)} = p_i^{y_i}(1-p_i)^{y_i}$

$$= \left(\frac{1}{1+e^{-x_i^T \beta}}\right)^{y_i} \left(\frac{e^{-x_i^T \beta}}{1+e^{-x_i^T \beta}}\right)^{1-y_i}$$

$i = 1, \ldots n$.

(2) $\underline{\text{Joint distribution}}$ $f(y_1, \ldots y_n)$

$= f(y_1) \cdot f(y_2) \cdot \ldots f(y_n)$. [$\because$ independent].

$$= \prod_{i=1}^{n} \left(\frac{1}{1+e^{-x_i^T \beta}}\right)^{y_i} \left(\frac{e^{-x_i^T \beta}}{1+e^{-x_i^T \beta}}\right)^{1-y_i}$$

(3) Likelihood fun $L(\beta_0, \beta_1, \beta_2, \sigma)$

$$= \underline{\underline{L(\beta)}} = \prod_{i=1}^{n} \left(\frac{1}{1+e^{-x_i^T\beta}}\right)^{y_i} \left(\frac{e^{-x_i^T\beta}}{1+e^{-x_i^T\beta}}\right)^{1-y_i}$$

(4) $l(\beta_0, \beta_1, \beta_2, \sigma) = \log(L(\beta_0, \beta_1, \beta_2, \sigma))$.

$$= \log\left[\prod_{i=1}^{n} \left(\frac{1}{1+e^{-x_i^T\beta}}\right)^{y_i} \left(\frac{e^{-x_i^T\beta}}{1+e^{-x_i^T\beta}}\right)^{1-y_i}\right]$$

$$\underline{\underline{l(\beta)}} = \sum_{i=1}^{n} \left(y_i \log\left(\frac{1}{1+e^{-x_i^T\beta}}\right)\right) + (1-y_i)\log\left(\frac{e^{-x_i^T\beta}}{1+e^{-x_i^T\beta}}\right)$$

$$\underline{\underline{-l(\beta_0, \beta_1, \beta_2, \sigma)}} = -\sum_{i=1}^{n} y_i \log\frac{1}{1+e^{-x_i^T\beta}} + (1-y_i)\log\frac{e^{-x_i^T\beta}}{1+e^{-x_i^T\beta}}.$$

(5) $(\hat{\beta}_0, \hat{\beta}_1, \hat{\beta}_2, \hat{\sigma}) = \operatorname{argmax} L(\beta_0, \beta_1, \beta_2, \cancel{\beta_3}, \sigma)$.

$= \operatorname{argmax} l(\beta_0, \beta_1, \beta_2, \sigma) = \operatorname{arg\,min} -l(\beta_0, \beta_1, \beta_2, \sigma)$

$\because$ maximizing a function is same as maximizing its log.

Since log is a monotonically increaning function maximizing log is equivivalent to minimizing -ve of it i.e negative log.

S520_HW6_Q1_Q2

Rushank Ghanshyam Sheta

2022-11-30

1(7) Get estimates using matrix multiplication

```
data.carprice <- read.csv('/Users/rushank/Downloads/q1_carprice.csv')
data.carprice
```

```
##    Car Age Miles Price
## 1    1   5    57    85
## 2    2   4    40   103
## 3    3   6    77    70
## 4    4   5    60    82
## 5    5   5    49    89
## 6    6   5    47    98
## 7    7   6    58    66
## 8    8   6    39    95
## 9    9   2     8   169
## 10  10   7    69    70
## 11  11   7    89    48
```

```
# get individual attribute values and store it in vectors
y <- data.carprice[,4]
x1 <- data.carprice[,2]
x2 <- data.carprice[,3]
```

```
# convert vectors to matrix
X.matrix <- cbind(1,x1,x2)
X.matrix
```

```
##         x1 x2
##  [1,] 1  5 57
##  [2,] 1  4 40
##  [3,] 1  6 77
##  [4,] 1  5 60
##  [5,] 1  5 49
##  [6,] 1  5 47
##  [7,] 1  6 58
##  [8,] 1  6 39
##  [9,] 1  2  8
## [10,] 1  7 69
## [11,] 1  7 89
```

```
# matrix multiplication
beta.est <- solve(t(X.matrix)%*%X.matrix)%*%t(X.matrix)%*%y
beta.est
```

```
##            [,1]
##    183.0352076
## x1  -9.5042704
## x2  -0.8214833
```

Beta estimates are Beta1 = -9.5042 and Beta2 = -0.8214, and intercept is 183.0352.

1(8) Interpretation

Here, Beta1 = -9.5042 and Beta2 = -0.8214, and intercept is 183.0352

Interpretation: Beta1 and Beta2 suggests that both the Age and Miles are negatively correlated to the target(price). It means that when Age or Miles Increases the price decreases. Also, Coefficient for Age is more than that of miles, that means impact of Age on price will be more than miles. And, one unit increase in Age will decrease the value of price by 9.5042 and 1 unit change in Miles will impact the price by -0.8214.

1(9) Prediction using custom input

```
cat('Beta0: ',beta.est[1])
```

```
## Beta0:  183.0352
```

```
cat('\nBeta1(Age): ',beta.est[2])
```

```
##
## Beta1(Age):  -9.50427
```

```
cat('\nBeta2(Miles): ',beta.est[3])
```

```
##
## Beta2(Miles):  -0.8214833
```

```
test = c(4,50)
print('Year, Mileage(in thousands): ',test)
```

```
## [1] "Year, Mileage(in thousands): "
```

```
cat('The prediction is: ',beta.est[1]+(beta.est[2]*test[1])+(beta.est[3]*test[2]))
```

```
## The prediction is:  103.944
```

1(10) Using lm function

```
lm_out = lm(y~x1+x2)
lm_out
```

```
##
## Call:
## lm(formula = y ~ x1 + x2)
##
## Coefficients:
## (Intercept)           x1           x2
##    183.0352      -9.5043      -0.8215
```

from lm function Beta0 is 183.0352, Beta1 is -9.5043 and Beta2 is -0.8215. Which are exactly the same as what we get from 1(7).

Question 2

```
#install.packages("ISLR")
library(ISLR)
summary(Default)
```

```
##  default    student       balance           income     
##  No :9667   No :7056   Min.   :   0.0   Min.   :  772  
##  Yes: 333   Yes:2944   1st Qu.: 481.7   1st Qu.:21340  
##                        Median : 823.6   Median :34553  
##                        Mean   : 835.4   Mean   :33517  
##                        3rd Qu.:1166.3   3rd Qu.:43808  
##                        Max.   :2654.3   Max.   :73554
```

2(6) Newton Raphson method to estimate Beta's

```r
# initalize parameters with random values
y <- as.numeric(Default$default)-1
x1 <- as.numeric(Default$student)-1
x2 <- Default$balance
x3 <- Default$income

X <- cbind(1,x1,x2,x3)
beta0 <- rep(0,4)
phat <- 1/(1+exp(-X%*%beta0))
beta1 <- beta0 + solve(t(X)%*%diag(c(phat*(1-phat)))%*%X)%*%t(X)%*%(y-phat)

i.count <- 1
print(c(i.count,beta1))
```

```
## [1]  1.000000e+00 -2.324718e+00 -4.132040e-02  5.307589e-04  7.966111e-07
```

```r
# loop until we get very small difference between previous and curent estimates
while (sum((beta1-beta0)^2) > 1e-6){
  beta0 <- beta1
  phat <- 1/(1+exp(-X%*%beta0))
  beta1 <- beta0 + solve(t(X)%*%diag(c(phat*(1-phat)))%*%X)%*%t(X)%*%(y-phat)
  i.count <- i.count+1
  print(c(i.count,beta1))
}
```

```
## [1]  2.000000e+00 -4.068918e+00 -1.154098e-01  1.440760e-03  2.095259e-06
## [1]  3.000000e+00 -6.142053e+00 -2.442090e-01  2.791335e-03  3.452405e-06
## [1]  4.000000e+00 -8.257790e+00 -4.110228e-01  4.139451e-03  3.659522e-06
## [1]  5.000000e+00 -9.949292e+00 -5.611771e-01  5.182132e-03  3.305623e-06
## [1]  6.000000e+00 -1.074128e+01 -6.347290e-01  5.660321e-03  3.072217e-06
## [1]  7.000000e+00 -1.086642e+01 -6.465264e-01  5.734951e-03  3.034252e-06
## [1]  8.000000e+00 -1.086904e+01 -6.467757e-01  5.736505e-03  3.033450e-06
## [1]  9.000000e+00 -1.086905e+01 -6.467758e-01  5.736505e-03  3.033450e-06
```

2(7) Interpretation

```r
cat('Beta0: ',beta1[1])
```

```
## Beta0:  -10.86905
```

```r
cat('\nBeta1(Student): ',beta1[2])
```

```
##
## Beta1(Student):  -0.6467758
```

```
cat('\nBeta2(Balance): ',beta1[3])
```

```
##
## Beta2(Balance):  0.005736505
```

```
cat('\nBeta2(Income): ',beta1[4])
```

```
##
## Beta2(Income):  3.03345e-06
```

The weights of out model are beta0, beta1, beta2, beta3. From the associated coefficients in the above cell, we can say that Student attribute has the highest and only(negative) correlation with the target attribute as it has the negative value. And the impact of Balance and Income is comparatively less on target than impact made by Student attribute.

2(8) Custom Input

```
sigmoid = function(x) {
   1 / (1 + exp(-x))
}

Student=0
Balance=900
Income=20000

py = -10.86905-(0.6467758*Student)+(Balance*0.005736505)+(Income*3.03345e-06)

cat('py: ', py)
```

```
## py:  -5.645526
```

```
cat('\nProbablity(sigmoid): ',sigmoid(py))
```

```
##
## Probablity(sigmoid):  0.003520847
```

Therefore the probability of a person who is not student with balance 900 and income of 20000 of defaulting is 0.003.

2(9) using glm command

```
glm(default~student+balance+income,family="binomial",data=Default)
```

```
##
## Call:  glm(formula = default ~ student + balance + income, family = "binomial",
##     data = Default)
##
## Coefficients:
## (Intercept)   studentYes      balance       income
##  -1.087e+01   -6.468e-01    5.737e-03    3.033e-06
##
## Degrees of Freedom: 9999 Total (i.e. Null);  9996 Residual
## Null Deviance:       2921
## Residual Deviance: 1572  AIC: 1580
```

Yes using the glm command also returns the same coefficients as that of newton Raphson Method.