STA314 Homework 4

student number: 1003942326

Yulin WANG

13/11/2019

Question 1

(i)

$$
\begin{aligned}
likelihood\ function:\ L(p) &= \prod_{i=1}^{n} p^{y_i}(1-p)^{1-y_i} \\
log-likelihood\ function:\ l(p) &= \sum_{i=1}^{n} log(p^{y_i}(1-p)^{1-y_i}) \\
&= \sum_{i=1}^{n} y_i \cdot log(p) + \sum_{i=1}^{n}(1-y_i) \cdot log(1-p) \\
&= log(p) \cdot \sum_{i=1}^{n} y_i + log(1-p) \cdot \sum_{i=1}^{n}(1-y_i)
\end{aligned}
$$

set $\frac{\partial l(p)}{\partial p} = 0$, then we get:

$$
\begin{gathered}
\frac{1}{p} \cdot \sum_{i=1}^{n} y_i - \frac{1}{1-p} \cdot \sum_{i=1}^{n}(1-y_i) = 0 \\
\Rightarrow\ (1-p) \cdot \sum_{i=1}^{n} y_i - p \cdot \sum_{i=1}^{n}(1-y_i) = 0 \\
(1-p) \cdot \sum_{i=1}^{n} y_i = p \cdot \sum_{i=1}^{n}(1-y_i) \\
\Rightarrow\ \sum_{i=1}^{n} y_i = np \quad \Rightarrow \quad \hat{p} = \frac{\sum_{i=1}^{n} y_i}{n}
\end{gathered}
$$

(ii)

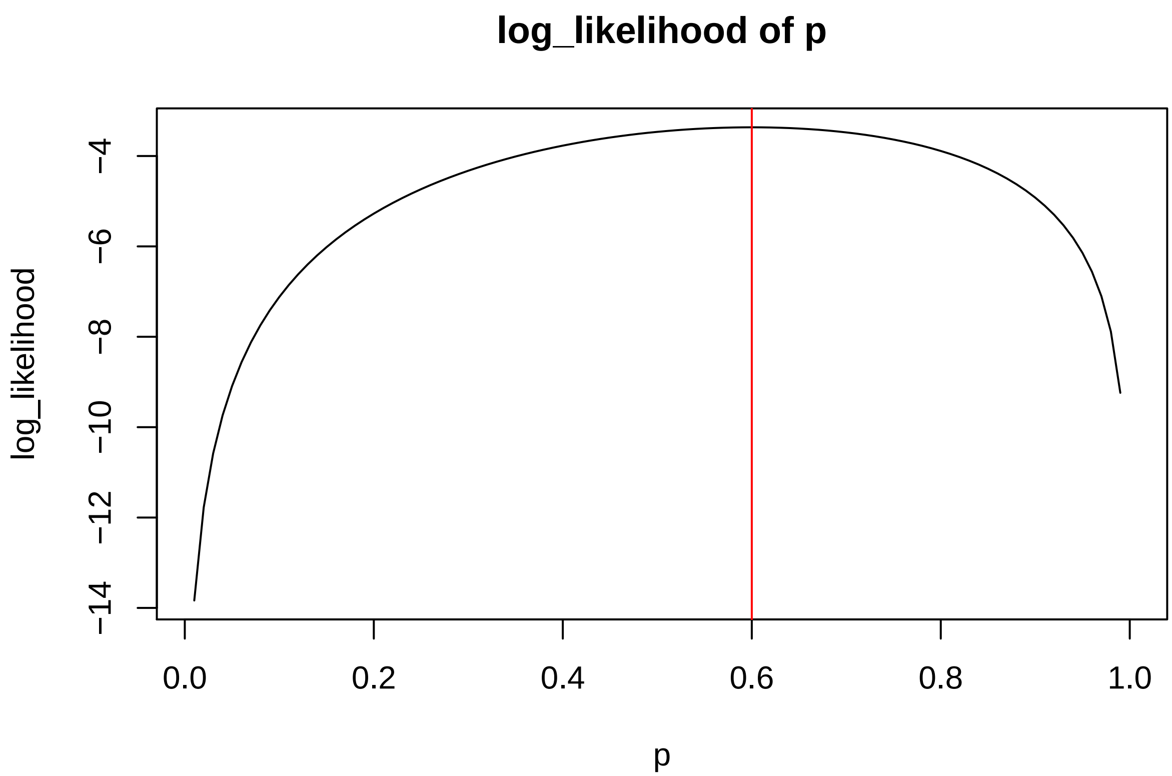
$$
\hat{p} = \frac{\sum_{i=1}^{n} y_i}{n} = \frac{\sum_{i=1}^{5} y_i}{5} = \frac{1+1+1+0+0}{5} = \frac{3}{5}
$$

(iii)

```r
p <- 1:100/100 #grid search sequence 0.01,0.02,...,0.99,1
log_likelihood <- 3*log(p) + (5-3)*log(1-p) #3 successes and 2 failures
plot(p, log_likelihood, type = "l", main = "log_likelihood of p")
p_hat<-p[which.max(log_likelihood)]  #value of p that maximizes the log-likelihood
p_hat
```

```
## [1] 0.6
```

```r
abline(v = p_hat, col= "red")
```



Question 2

(i)

$$likelihood\ function:\ L(\beta) = \prod_{i=1}^{n} p_i^{y_i}(1-p_i)^{1-y_i}$$

$$\begin{aligned}
log-likelihood\ function:\ l(\beta) &= \sum_{i=1}^{n} log(p_i^{y_i}(1-p_i)^{1-y_i}) \\
&= \sum_{i=1}^{n}\{y_i \cdot log(p_i) + (1-y_i)\cdot log(1-p_i)\} \\
&= \sum_{i=1}^{n}\{y_i \cdot log(\frac{p_i}{1-p_i}) + log(1-p_i)\} \\
&= \sum_{i=1}^{n}\{y_i \cdot (\beta_0+\beta_1 x_i) + log(1-\frac{e^{\beta_0+\beta_1 x_i}}{1+e^{\beta_0+\beta_1 x_i}})\} \\
&= \sum_{i=1}^{n}\{y_i \cdot (\beta_0+\beta_1 x_i) + log(\frac{1}{1+e^{\beta_0+\beta_1 x_i}})\} \\
&= \sum_{i=1}^{n}\{y_i \cdot (\beta_0+\beta_1 x_i) - log(1+e^{\beta_0+\beta_1 x_i})\}
\end{aligned}$$

(ii)

```r
# function ll to calculate the log-likelihood
ll <- function(beta, x, y){
  beta0 <- beta[1]
  beta1 <- beta[2]
  return (sum(y*(beta0 + beta1*x))-sum(log(1+exp(beta0 + beta1*x))))
}
```

(iii)

```r
library(ISLR)
data(Default)
model1 <- glm(default ~ balance, family = "binomial", data = Default)
summary(model1)
```

```
##
## Call:
## glm(formula = default ~ balance, family = "binomial", data = Default)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.2697  -0.1465  -0.0589  -0.0221   3.7589
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.065e+01  3.612e-01  -29.49   <2e-16 ***
## balance      5.499e-03  2.204e-04   24.95   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2920.6  on 9999  degrees of freedom
## Residual deviance: 1596.5  on 9998  degrees of freedom
## AIC: 1600.5
##
## Number of Fisher Scoring iterations: 8
```

(iv)

```r
default <- ifelse(Default$default == "Yes", 1, 0) #set "Yes"=1, "No"=0
coefs <- optim(c(0, 0), ll, x = Default$balance, y = default,
               control = list(fnscale = -1), hessian = TRUE)
coefs$par
```

```
## [1] -10.652058220   0.005499188
```

(v)

The maximum likelihood estimates are almost same obtained using optim() in part(iv) and using glm() in part(iii).

(vi)

```r
sqrt(diag(solve(coefs$hessian)))
```

```
## [1] 1.643313504 0.001021681
```

(vii)

The standard error estimates are different obtained using hessian in part(vi) and using glm() in part (iii).

Question 3

(i)

```
set.seed(100)
model2 <- glm(default ~ income + balance, family = "binomial", data = Default)
summary(model2)
```

```
##
## Call:
## glm(formula = default ~ income + balance, family = "binomial",
##     data = Default)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.4725  -0.1444  -0.0574  -0.0211   3.7245
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.154e+01  4.348e-01 -26.545  < 2e-16 ***
## income       2.081e-05  4.985e-06   4.174 2.99e-05 ***
## balance      5.647e-03  2.274e-04  24.836  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2920.6  on 9999  degrees of freedom
## Residual deviance: 1579.0  on 9997  degrees of freedom
## AIC: 1585
##
## Number of Fisher Scoring iterations: 8
```

(ii)

```
#function boot.fn
boot.fn <- function(data, index){
  return (coef(glm(default ~ income + balance, family = "binomial", data = data, subset = index)))
}
boot.fn(Default, 1:nrow(Default))
```

```
##   (Intercept)        income       balance
## -1.154047e+01  2.080898e-05  5.647103e-03
```

(iii)

```r
library(boot)
boot(Default, boot.fn, R=100)
```

```
##
## ORDINARY NONPARAMETRIC BOOTSTRAP
##
##
## Call:
## boot(data = Default, statistic = boot.fn, R = 100)
##
##
## Bootstrap Statistics :
##          original        bias     std. error
## t1* -1.154047e+01 -6.178943e-02 4.639390e-01
## t2*  2.080898e-05  6.431276e-07 4.530553e-06
## t3*  5.647103e-03  2.375342e-05 2.398827e-04
```

(iv)

Standard error estimates for income and balance are pretty close using glm summary function and bootstrap with R=100.

- income: 4.985e-06 using glm summary, 4.530553e-06 using bootstrap
- balance: 2.274e-04 using glm summary, 2.398827e-04 using bootstrap