Homework #5 Resampling Methods

1. **(Jackknife)** An acoustic studio needs to estimate the range of voice frequencies that an adult singer can produce. A sample of n = 10 recordings contains frequencies 102, 115, 127, 127, 162, 180, 184, 205, 239, 240.

   (a) Compute the jackknife estimator of the population highest frequency of a human voice.

```
freq <- c(102, 115, 127, 127, 162, 180, 184, 205, 239, 240)
theta.hat <- function(x){return(max(x))} # original estimator
jn <- jackknife( freq, theta.hat )
theta.hat(freq) - jn$jack.bias

## [1] 240.9
```

$$\hat{\theta} = max\{102, 115, 127, 127, 162, 180, 184, 205, 239, 240\} = 240$$
$$\hat{\theta}_{(-i)} = \underbrace{240, 240, \cdots, 240,}_{\times 9} 239 \qquad i = 1, 2, \cdots, 10$$
$$\hat{\theta}_{(\bullet)} = \frac{1}{n}\sum_{i=1}^{10} \hat{\theta}_{(-i)} = 239.9$$
$$\begin{aligned}\hat{\theta}_{JK} &= n\hat{\theta} - (n-1)\hat{\theta}_{(\bullet)} \\ &= 10 \times 240 - (10-1) \times 239.9 \\ &= 240.9\end{aligned}$$

   (b) Compute the jackknife estimator of the population lowest frequency of a human voice.

```
freq <- c(102, 115, 127, 127, 162, 180, 184, 205, 239, 240)
theta.hat <- function(x){return(min(x))} # original estimator
jn <- jackknife( freq, theta.hat )
theta.hat(freq) - jn$jack.bias

## [1] 90.3
```

$$\hat{\theta} = min\{102, 115, 127, 127, 162, 180, 184, 205, 239, 240\} = 102$$
$$\hat{\theta}_{(-i)} = 115, \underbrace{102, \cdots, 102}_{\times 9} \qquad i = 1, 2, \cdots, 10$$
$$\hat{\theta}_{(\bullet)} = \frac{1}{n}\sum_{i=1}^{10} \hat{\theta}_{(-i)} = 103.3$$
$$\begin{aligned}\hat{\theta}_{JK} &= n\hat{\theta} - (n-1)\hat{\theta}_{(\bullet)} \\ &= 10 \times 102 - (10-1) \times 103.3 \\ &= 90.3\end{aligned}$$

Natural range of human voice frequencies: (85, 255). The Jackknife estimation is very close to the fact.

(c) Generalize the results. Assume a sample $X_1, \cdots, X_n$ of size n, where X_1, X_2 are the smallest two observations, and X_{n-1}, X_n are the largest two. Derive equations for the jackknife estimators of the population minimum and maximum.

i. Population maximum

$$\hat{\theta} = max\{X_1, X_2, \cdots, X_{n-1}, X_n\} = X_n$$

$$\hat{\theta}_{(-i)} = \underbrace{X_n, \cdots, X_n}_{\times(n-1)}, X_{n-1} \qquad i = 1, 2, \cdots, n$$

$$\hat{\theta}_{(\bullet)} = \frac{1}{n}\sum_{i=1}^{n}\hat{\theta}_{(-i)} = \frac{(n-1)X_n + X_{n-1}}{n}$$

$$\begin{aligned}\hat{\theta}_{JK} &= n\hat{\theta} - (n-1)\hat{\theta}_{(\bullet)} \\ &= nX_n - (n-1)\frac{(n-1)X_n + X_{n-1}}{n} \\ &= \frac{(2n-1)X_n - (n-1)X_{n-1}}{n}\end{aligned}$$

$$\begin{aligned}\hat{Bias} &= \hat{\theta} - \hat{\theta}_{JK} \\ &= \frac{(X_{n-1} - X_n)(n-1)}{n}\end{aligned}$$

Appling this Jackknife estimator to the tank example, we obtain an estimation of 103.2, which is not so accurate as the estimation, 109.2, proposed by the statisticians.

ii. Population minimum

$$\hat{\theta} = min\{X_1, X_2, \cdots, X_{n-1}, X_n\} = X_1$$

$$\hat{\theta}_{(-i)} = X_2, \underbrace{X_1, \cdots, X_1}_{\times(n-1)} \qquad i = 1, 2, \cdots, n$$

$$\hat{\theta}_{(\bullet)} = \frac{1}{n}\sum_{i=1}^{n}\hat{\theta}_{(-i)} = \frac{(n-1)X_1 + X_2}{n}$$

$$\begin{aligned}\hat{\theta}_{JK} &= n\hat{\theta} - (n-1)\hat{\theta}_{(\bullet)} \\ &= nX_1 - (n-1)\frac{(n-1)X_1 + X_2}{n} \\ &= \frac{(2n-1)X_1 - (n-1)X_2}{n}\end{aligned}$$

$$\begin{aligned}\hat{Bias} &= \hat{\theta} - \hat{\theta}_{JK} \\ &= \frac{(X_2 - X_1)(n-1)}{n}\end{aligned}$$

2. **(Jackknife)** One needs to estimate θ, the frequency of days with 0 traffic accidents on a certain highway. The data are collected. During 40 days, there are 26 days with 0

accidents, 10 days with 1 accident, and 4 days with 2 accidents.
Statistician A estimates θ with a sample proportion $\hat{p} = 26/40 = 0.65$.
Statistician B argues that this method does not distinguish between the days with 1 accident and the days with 2 accidents, losing some valuable information. She suggests to model the number of accidents X by a Poisson distribution with parameter λ. Then we have $\theta = P\{X = 0\} = exp(-\lambda)$. She estimates λ with $\hat{\lambda} = \bar{X}$. Then $\hat{\theta} = exp(-\hat{\lambda})$. However, this estimator is biased.

(a) Use jackknife to estimate the bias of $\hat{\theta}$.

```
accident <- sample( c(rep(0,26),rep(1,10),rep(2,4)) )
theta.hat <- function(x){return( exp( - mean(x) ) )} # original estimator
jn <- jackknife( accident, theta.hat )
jn$jack.bias # estimated bias

## [1] 0.003683256
```

The estimated bias of $\hat{\theta}$ is 0.003683256.

(b) Compute the jackknife estimator $\hat{\theta_{JK}}$ based on θ.

```
theta.hat(accident) - jn$jack.bias # jackknife estimator of theta

## [1] 0.6376282
```

$\hat{\theta_{JK}} = 0.6339449$.

(c) Apply the jackknife method to estimate the bias of $\hat{p}$ and compute the jackknife estimator $\hat{p_{JK}}$. Explain the result.

```
p.hat <- function(x){return( mean(x == 0) )} # original estimator
jn <- jackknife( accident, p.hat )
jn$jack.bias # estimated bias

## [1] 0

p.hat(accident) - jn$jack.bias # jackknife estimator of p

## [1] 0.65
```

The estimated bias of $\hat{p}$ is 0. And the jackknife estimator $\hat{p_{JK}} = 0.65$. That means $\hat{p}$ is an unbiased estimator, so Jackknife cannot improve more.

(d) Now, let us estimate the standard deviation of $\hat{p}$. The standard formula is

$$s_{\hat{p}} = \hat{Std}(\hat{p}) = \sqrt{\frac{\hat{p}(1-\hat{p})}{n}}$$

Use jackknife to estimate the bias of $s_{\hat{p}}$. Propose an estimator with a smaller bias and calculate it from the given data. (Knowledge of Poisson distribution is not needed to solve this problem.)

```r
sp.hat <- function(x){ # original estimator
  n <- length(x)
  p <- mean(x == 0)
  sp <- sqrt(p*(1-p)/n)
  return(sp)
  }
jn <- jackknife( accident, sp.hat )
jn$jack.bias # estimated bias

## [1] 0.03638723

sp.hat(accident) - jn$jack.bias# JK estimator

## [1] 0.03902828
```

The estimated bias of $s_{\hat{p}} = 0.03638723$ under Jackknife procedure. An estimator with smaller bias:

$$\hat{Std}(\hat{p})_{JK} = 0.03902828$$

.

3. **(Cross-Validation. Pages 198-199, chap. 5, ≈#5)**
Logistic regression is used to predict the probability of default using `income`, `balance`, and `student` on the `Default` data set in R package `ISLR`. Using

   (a) the validation set approach,

   (b) leave-one-out cross-validation, and

   (c) K-fold cross-validation for $K = 100$ and $K = 1000$,

   estimate the test error of this logistic regression model and decide whether it will be improved when the dummy variable `student` is excluded from this prediction.

   (a) the validation set approach

```r
set.seed(666)
n <- nrow(Default)
train = sample( n, n/2 )
reg = glm( default ~ income + balance + student,
           family="binomial", data=Default[train,] )
y.hat = predict( reg, Default, type="response" )
pred <- ifelse(y.hat > 0.5, "Yes", "No")
a.1 <- mean( Default$default[-train] != pred[-train] )
# remove student
reg = glm( default ~ income + balance,
           family="binomial", data=Default[train,] )
y.hat = predict( reg, Default, type="response" )
```

```
pred <- ifelse(y.hat > 0.5, "Yes", "No")
a.2 <- mean( Default$default[-train] != pred[-train] )
# output
data.frame('test error rate' = c('include student' = a.1, 'remove student' = a.2))
```

```
##                 test.error.rate
## include student          0.0244
## remove student           0.0240
```

After removing the student variable, the test error of the validation set approach slight decreases from 0.0244 to 0.0240. The model does not significantly improve.

(b) leave-one-out cross-validation

```
pred.loocv <- rep(NA, n)
for (i in 1:n){
  reg <- glm( default ~ income + balance + student,
              family="binomial", data=Default[-i,] )
  y.hat <- predict( reg, Default, type="response" )[i]
  pred.loocv[i] <- ifelse(y.hat > 0.5, "Yes", "No")
  }
b.1 <- mean( Default$default != pred.loocv )
# remove student
for (i in 1:n){
  reg <- glm( default ~ income + balance,
              family="binomial", data=Default[-i,] )
  y.hat <- predict( reg, Default, type="response" )[i]
  pred.loocv[i] <- ifelse(y.hat > 0.5, "Yes", "No")
  }
b.2 <- mean( Default$default != pred.loocv )
# output
data.frame('test error rate' = c('include student' = b.1, 'remove student' = b.2)
```

```
##                 test.error.rate
## include student          0.0268
## remove student           0.0263
```

Under LOOCV procedure, removing the student variable would cause the test error to slight decrease from 0.0268 to 0.0263. The model does not improve much.

(c) K-fold cross-validation for $K = 100$ and $K = 1000$

```
# K = 100
k <- 100
pred.k.100 <- rep(NA, n)
k.group <- split(1:n, sample(1:n, k))
for (i in 1:k){
```

```
  reg = glm( default ~ income + balance + student,
             family="binomial", data=Default[-k.group[[i]],] )
  y.hat = predict( reg, Default, type="response" )[k.group[[i]]]
  pred.k.100[k.group[[i]]] <- ifelse(y.hat > 0.5, "Yes", "No")
  }
c1.1 = mean( Default$default != pred.k.100 )
# remove student
for (i in 1:k){
  reg = glm( default ~ income + balance,
             family="binomial", data=Default[-k.group[[i]],] )
  y.hat = predict( reg, Default, type="response" )[k.group[[i]]]
  pred.k.100[k.group[[i]]] <- ifelse(y.hat > 0.5, "Yes", "No")
  }
c1.2 = mean( Default$default != pred.k.100 )
# k =1000
k <- 1000
pred.k.1000 <- rep(NA, n)
k.group <- split(1:n, sample(1:n, k))
for (i in 1:k){
  reg = glm( default ~ income + balance + student,
             family="binomial", data=Default[-k.group[[i]],] )
  y.hat = predict( reg, Default, type="response" )[k.group[[i]]]
  pred.k.1000[k.group[[i]]] <- ifelse(y.hat > 0.5, "Yes", "No")
  }
c2.1 = mean( Default$default != pred.k.1000 )
# remove student
for (i in 1:k){
  reg = glm( default ~ income + balance,
             family="binomial", data=Default[-k.group[[i]],] )
  y.hat = predict( reg, Default, type="response" )[k.group[[i]]]
  pred.k.1000[k.group[[i]]] <- ifelse(y.hat > 0.5, "Yes", "No")
  }
c2.2 = mean( Default$default != pred.k.1000 )
# output
data.frame( 'K' = c(100,100,1000,1000),
            'student' = c('include','remove','include','remove'),
            'test error rate' = c(c1.1,c1.2,c2.1,c2.2))

##      K student test.error.rate
## 1  100 include          0.0269
## 2  100  remove          0.0263
## 3 1000 include          0.0268
## 4 1000  remove          0.0263
```

When we use K-fold method with $K = 100$, removing the student variable will help

decrease the test error rate from 0.0269 to 0.0263. With $K = 1000$, the test error rate will decrease from 0.0268 to 0.0263 if we remove the student variable. The model does not obviously improve.