STAT341 A4 Markdown

QUESTION 1: Horvitz-Thompson Estimation – SRSWOR [18 points]

(a) Using the following code, take a *simple random sample without replacement* of size $n = 50$ from the population. (This is not worth points)

i. Calculate the Horvitz-Thompson estimate of the population average happiness score.

```
happy <- read.csv("world_happiness.csv", header = TRUE)
srsSampIndex <- read.table("srsSampIndex.txt")$V1
srsSamp <- happy[srsSampIndex, ]

createvariateFnN <- function(popData, variate, N = 1) {
  function(u) {
    popData[u, variate]/N
  }
}

N = 156
n = 50

# functions from class
inclusionProb <- createInclusionProbFn(1:N, sampSize=n)
happyHTestimator <- createHTestimator(inclusionProb)

happyAvg <- createvariateFnN(happy, "Score", N=N)
happyAvg_HT_srswor <- happyHTestimator(srsSampIndex, happyAvg)
print(paste0("The HT Estimate is: ", happyAvg_HT_srswor))
```

```
## [1] "The HT Estimate is: 5.3863"
```

ii. Calculate the standard error for this estimate.

```
estVarHT <- function(y_u, pi_u, pi_uv) {
  ## y_u = an n element array containing the variate values for the sample
  ## pi_u = an n element array containing the (marginal) inclusion ## probabilities for the sample
  ## pi_uv = an nxn matrix containing the joint inclusion probabilities ## for the sample
  delta <- pi_uv - outer(pi_u, pi_u)
  estimateVar <- sum((delta/pi_uv) * outer(y_u/pi_u, y_u/pi_u))
  return(abs(estimateVar))
}

y_u <- srsSamp$Score/N
pi_u <- rep(n/N, n)
pi_uv <- matrix((n * (n - 1)) / (N * (N - 1)),
                nrow = n, ncol = n)
diag(pi_uv) <- pi_u
```

```r
se_HT_srswor <- sqrt(estVarHT(y_u, pi_u, pi_uv))
print(paste0("The std. error is: ", se_HT_srswor))
```

```
## [1] "The std. error is: 0.134142565516323"
```

iii. Calculate an approximate 95% confidence interval for the population average happiness score.

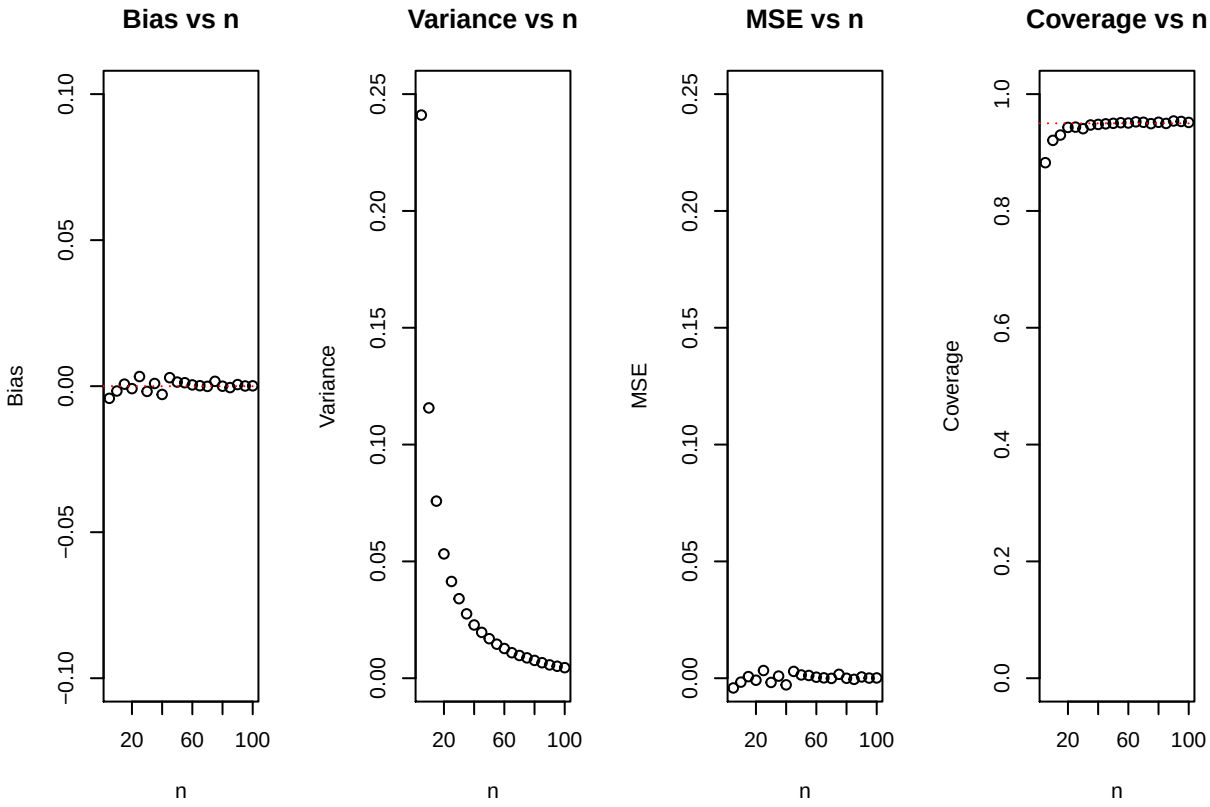
```r
happy_srswor <- happyAvg_HT_srswor + 2 * c(-1, 1) * se_HT_srswor
print(happy_srswor) # print confidence interval
```

```
## [1] 5.118015 5.654585
```

(b) In this question you will explore the dependence of the Horvitz-Thompson estimator's sampling distribution on sample size. Consider the sample sizes $n \in \{5, 10, 15, 20, 25, \ldots, 100\}$.

```r
happy_sizes <- seq(5, 100, by=5)
happy_biases <- c(NA);
happy_var <- c(NA);
happy_mse <- c(NA);
happy_cov <- c(NA);

est <- rep(0, 10000)
ci <- matrix(0, nrow = 10000, ncol = 2)
happyMean <- mean(happy$Score)

# for each sample size n, calculate the bias, var, MSE, and cov
for (val in happy_sizes) {
  pi_u = rep(val/N, val)
  pi_uv = matrix((val * (val - 1)) / (N * (N - 1)),
                 nrow = val, ncol = val)
  diag(pi_uv) = pi_u

  for (i in 1:10000) {
    samp = sample(happy$Score, size=val, replace=FALSE)
    y_u = samp/N
    est[i] = sum(y_u/pi_u)
    se = sqrt(estVarHT(y_u, pi_u, pi_uv))
    ci[i, ] = sum(y_u/pi_u) + 2 * c(-1, 1) * se
  }

  # bias, variance, MSE, and coverage
  bias_srswor = mean(est - happyMean)
  variance_srswor = var(est)
   MSE_srswor = mean((est - happyMean)^2)
  coverage = apply(X=ci, MARGIN = 1, FUN = function(u) {
    happyMean >= u[1] & happyMean <= u[2]
  })

  # add values to the columns
  happy_biases[val/5] = bias_srswor;
  happy_var[val/5] = variance_srswor;
  happy_mse[val/5] = MSE_srswor;
```

```
  happy_cov[val/5] = mean(coverage);
}

# plots of bias, variance, mse, coverage vs n
par(mfrow = c(1, 4))
plot(happy_sizes, happy_biases, main="Bias vs n",
     xlab="n", ylab="Bias", ylim=c(-0.1, 0.1))
abline(h=0, lty=3, col="red")
plot(happy_sizes, happy_var, main="Variance vs n",
     xlab="n", ylab="Variance", ylim=c(0, 0.25))
plot(happy_sizes, happy_biases, main="MSE vs n",
     xlab="n", ylab="MSE", ylim=c(0, 0.25))
plot(happy_sizes, happy_cov, main="Coverage vs n",
     xlab="n", ylab="Coverage", ylim=c(0, 1))
abline(h=0.95, lty=3, col="red")
```



Bias and n looks to have a high correlation and suggests bias will be near 0 for higher sample sizes.

Variance vs n looks to be following an decreasing exponential curve which suggests the variance will grow smaller as n increases.

MSE and n looks to have a high correlation and also suggests MSE is close to 0 for higher sample sizes.

For coverage and n, it looks that coverage increases as n increases. Also for every n, a high proportion of the 95% confidence intervals contain the true value of the happy population mean.

QUESTION 2: Horvitz-Thompson Estimation – Stratified Random Sampling

(a) Show that the (marginal) inclusion probability π_u for stratified random sampling is

$$\pi_u = \frac{n_h}{N_h} \qquad \text{if } u \in \mathcal{P}_h$$

$$\text{if u} \in \mathcal{P}_h \Longrightarrow \pi_u = \frac{1 \times \begin{pmatrix} N_h - 1 \\ n_h - 1 \end{pmatrix}}{\begin{pmatrix} N_h \\ n_h \end{pmatrix}}$$

$$\pi_u = \frac{(N_h - 1)!}{(n_h - 1)!(N_h - n_h)!} \div \frac{N_h!}{n_h!(N_h - n_h)!}$$

$$\pi_u = \frac{(N_h - 1)!}{(n_h - 1)!} \div \frac{N_h(N_h - 1)!}{n_h(n_h - 1)!}$$

$$\text{hence} \pi_u = \frac{n_h}{N_h}$$

(b) Show that the joint inclusion probability π_{uv} for stratified random sampling and $u \neq v$ is

$$\pi_{uv} = \begin{cases} \frac{n_h(n_h-1)}{N_h(N_h-1)} & \text{if } u, v \in \mathcal{P}_h \\ \\ \frac{n_h n_k}{N_h N_k} & \text{if } u \in \mathcal{P}_h, v \in \mathcal{P}_k \end{cases}$$

$$\text{if u, v} \in \mathcal{P}_h \Longrightarrow \pi_{uv} = \frac{\begin{pmatrix} N_h - 2 \\ n_h - 2 \end{pmatrix}}{\begin{pmatrix} N_h \\ n_h \end{pmatrix}}$$

$$\pi_{uv} = \frac{(N_h - 2)!}{(n_h - 2)!(N_h - n_h)!} \div \frac{N_h!}{n_h!(N_h - n_h)!}$$

$$\pi_{uv} = \frac{(N_h - 2)!}{(n_h - 2)!} \div \frac{N_h(N_h - 1)(N_h - 2)!}{n_h(n_h - 1)(n_h - 2)!}$$

$$\text{hence} \pi_{uv} = \frac{n_h(n_h - 1)}{N_h(N_h - 1)}$$

$$\text{if u} \in \mathcal{P}_h, \text{v} \in \mathcal{P}_k \Longrightarrow \pi_{uv} = \frac{\begin{pmatrix} N_h - 1 \\ n_h - 1 \end{pmatrix} \times \begin{pmatrix} N_k - 1 \\ n_k - 1 \end{pmatrix}}{\begin{pmatrix} N_h \\ n_h \end{pmatrix} \times \begin{pmatrix} N_k \\ n_k \end{pmatrix}}$$

$$\pi_{uv} = \frac{(N_h - 1)!(N_k - 1)!}{(n_h - 1)!(N_h - n_h)!(n_k - 1)!(N_k - n_k)!} \div \frac{N_h!N_k!}{n_h!n_k!(N_h - n_h)!(N_k - n_k)!}$$

$$\pi_{uv} = \frac{n_h n_k}{N_h N_k}$$

$$\text{hence} \pi_{uv} = \begin{cases} \frac{n_h(n_h-1)}{N_h(N_h-1)} & \text{if } u, v \in \mathcal{P}_h \\ \\ \frac{n_h n_k}{N_h N_k} & \text{if } u \in \mathcal{P}_h, v \in \mathcal{P}_k \end{cases}$$

(c) Add a new column to the `happy` dataframe called `stratLabel` which assigns a numeric label to each `Continent`. In particular, when `Continent = "Africa"` then `stratLabel = 1`; when `Continent = "Asia"` then `stratLabel = 2`; when `Continent = "Europe"` then `stratLabel = 3`; when `Continent = "North America"` then `stratLabel = 4`; when `Continent = "Oceania"` then `stratLabel = 5`; when `Continent = "South America"` then `stratLabel = 6`. Once you have done this, present the output from the command `table(happy$stratLabel)`.

```
stratLabel <- seq(1:156)
hapcont <- as.character(happy$Continent)

for (i in 1:156) {
  j = hapcont[i]
  if (j == "Africa")
    stratLabel[i] <- 1
  else if (j == "Asia")
    stratLabel[i] <- 2
  else if (j == "Europe")
    stratLabel[i] <- 3
  else if (j == "North America")
    stratLabel[i] <- 4
  else if (j == "Oceania")
    stratLabel[i] <- 5
  else
    stratLabel[i] <- 6
}

happy <- data.frame(happy, stratLabel)
table(happy$stratLabel)
```

```
##
##  1  2  3  4  5  6
## 44 40 47 13  2 10
```

(d)

i. Calculate the Horvitz-Thompson estimate of the population average happiness score. You will find the `getInclusionProbStrat` function defined on page 6 useful. This function takes the inputs `stratLabel` and `stratSampSize` as defined below, and outputs an N element array containing the inclusion probabilities for each unit in the population $\mathcal{P}$.

```
stratSampIndex <- read.table("stratSampIndex.txt")$V1
stratSamp <- happy[stratSampIndex, ]

N = 156
n = 50

# function from assignment
stratIncProb <- getInclusionProbStrat(stratLabel, c(14, 13, 15, 4, 1, 3))
```

```r
strat_y_u <- stratSamp$Score/N

stratAvg_HT_srswor <- sum(strat_y_u/stratIncProb[stratSampIndex])
print(paste0("The HT estimate is: ", stratAvg_HT_srswor))
```

```
## [1] "The HT estimate is: 5.41501466023293"
```

ii. Calculate the standard error for this estimate. Feel free to use the `estVarHT` function from Question 1. You will find the `getJointInclusionProbStrat` function defined on page 6 useful. This function takes the inputs `stratLabel` and `stratSampSize` as defined above, and outputs an $N \times N$ matrix containing the joint inclusion probabilities for each unit in the population $\mathcal{P}$.

```r
# function from assignment
stratJointIncProb <- getJointInclusionProbStrat(stratLabel, c(14, 13, 15, 4, 1, 3))
diag(stratJointIncProb) <- stratIncProb

strat_se_HT_srswor <- sqrt(estVarHT(strat_y_u, stratIncProb[stratSampIndex],
                                    stratJointIncProb[stratSampIndex,][,stratSampIndex]))
print(paste0("The std. error is: ", strat_se_HT_srswor))
```

```
## [1] "The std. error is: 0.128551998546129"
```

iii. Calculate an approximate 95% confidence interval for the population average happiness score.

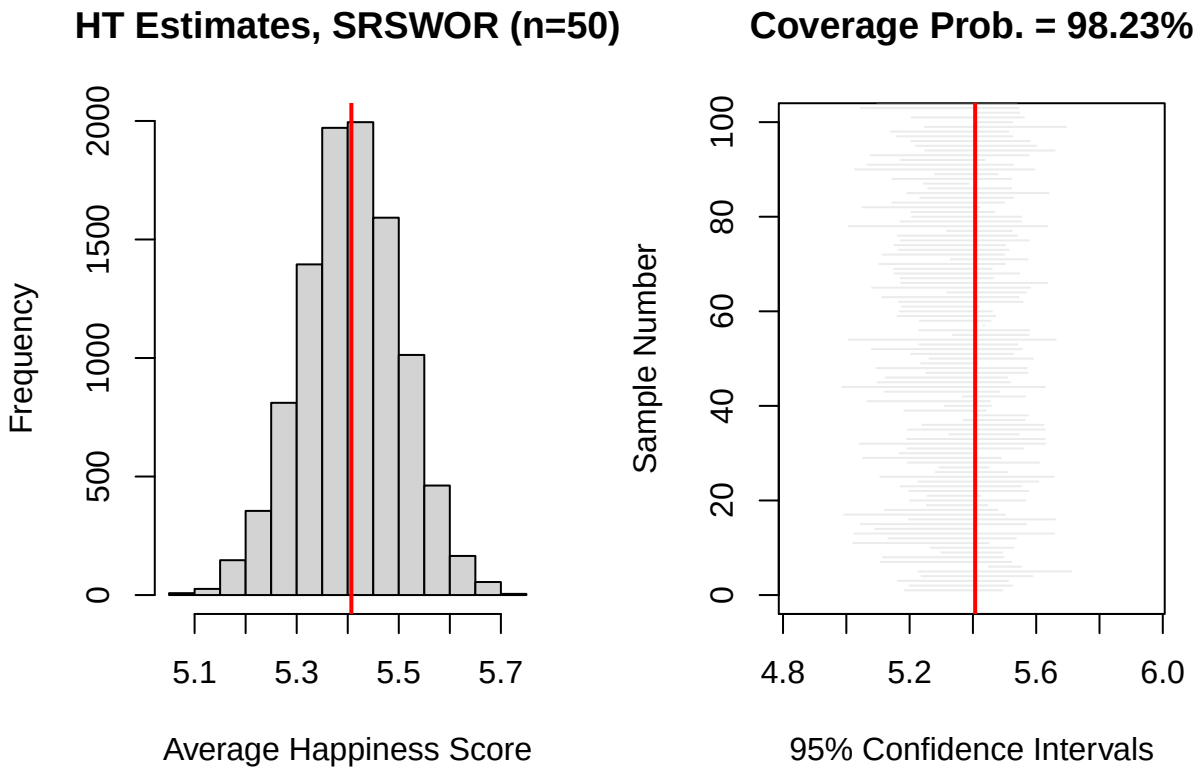
```r
strat_srswor <- stratAvg_HT_srswor + 2 * c(-1, 1) * strat_se_HT_srswor
print(strat_srswor) # print confidence interval
```

```
## [1] 5.157911 5.672119
```

(e) Take 10,000 stratified random samples of size $n = 50$ from the population, each with $\{n_1 = 14, n_2 = 13, n_3 = 15, n_4 = 4, n_5 = 1, n_6 = 3\}$. For each sample calculate the Horvitz-Thompson estimate, as well as an approximate 95% confidence interval, for the population average happiness score. Construct the following two plots (laid out in a 1×2 grid). See Tutorial 7 for an example.

```r
strat_est <- rep(0, 10000)
strat_ci <- matrix(0, nrow=10000, ncol=2)

for (i in 1:10000) {
  samp <- stratRS(stratLabel, c(14, 13, 15, 4, 1, 3))
  y_u <- happy$Score[samp]/N
  strat_est[i] <- sum(y_u/stratIncProb[samp])
  se <- sqrt(estVarHT(y_u, stratIncProb[samp],
                      stratJointIncProb[samp,][,samp]))
  strat_ci[i, ] <- sum(y_u/stratIncProb[samp]) + 2 * c(-1, 1) * se
}

par(mfrow = c(1,2))
hist(strat_est, col="lightgrey", main="HT Estimates, SRSWOR (n=50)", xlab="Average Happiness Score")
abline(v=happyMean, col="red", lwd=2)
strat_coverage <- apply(X=strat_ci, MARGIN=1, function(u) {
  happyMean >= u[1] & happyMean <= u[2]
})
```

```
plot(0, type="n", ylim=c(0,100), xlim=c(min(strat_ci[,1]),max(strat_ci[,2])),
     xlab="95% Confidence Intervals", ylab="Sample Number",
     main=paste0("Coverage Prob. = ", round(100*mean(strat_coverage),2), "%"))
for (i in 1:10000) {
  segments(x0=strat_ci[i, 1], y0=i, x1=ci[i,2], y1=i, col=adjustcolor("gray",alpha=0.3))
}
abline(v=happyMean, col="red", lwd=2)
```



(f) Using your calculations from part (e), estimate the sampling bias, sampling variance, sampling mean squared error (MSE), and coverage associated with this stratified random sample Horvitz-Thompson estimator. Compare these results to the $n = 50$ case from Question 1(b) i.

```
strat_bias_srswor = mean(strat_est - happyMean)
strat_variance_srswor = var(strat_est)
strat_MSE_srswor = mean((strat_est - happyMean)^2)
strat_cov = mean(strat_coverage)

# results for bias, variance, MSE, coverage prob.
cat("Q2e results:\n ","bias = ", strat_bias_srswor,
              " variance = ", strat_variance_srswor,
              " MSE = ", strat_MSE_srswor,
              " coverage prob. = ", strat_cov, "\n")


# results from question 1b (n=50)
cat("Q1b results:\n", "bias = ",happy_biases[10],
              " variance = ", happy_var[10],
              " MSE = ", happy_mse[10],
              " coverage prob. = ", happy_cov[10], "\n")
```

```
## Q2e results:
##   bias =  -5.415399e-05  variance =  0.0093467  MSE =  0.009345768  coverage prob. =  0.9823
```

```
## Q1b results:
##  bias =  0.001404368  variance =  0.01692398  MSE =  0.01692426  coverage prob. =  0.95
```

Strata Random Sampling is preferred

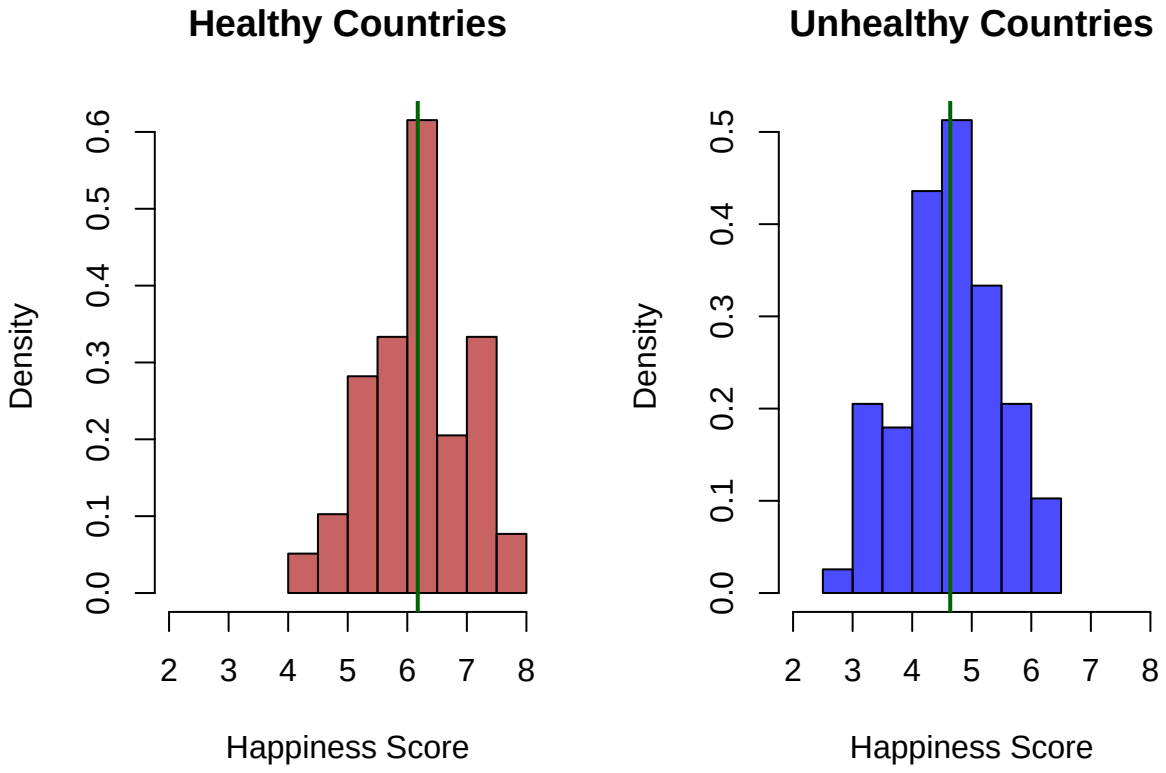
Compared to regular SRSWOR, Strata RS is preferred since it has:
A bias closer to 0
A lower variance
A lower MSE
A higher coverage probability for 95% confidence intervals

Strata Random Sampling, considers units from each continent while the regular method can be using samples with units from the same continent or discarding units from other continents which do not give accurate values for the average happiness score. So it was expected Strata Random Sampling is preferred for this case.

QUESTION 3: Permutation Test

(a) Construct two *density* histograms: one of the happiness scores for healthy countries, and the other of happiness scores for unhealthy countries, and plot them next to each in a 1×2 grid. Be sure to include informative titles and axis labels. To enhance comparability ensure that the bin widths and x-axes of the histograms are the same, and indicate (with a vertical line) the average happiness score in each sub-population.

```
pop <- list(pop1 = happy[order(happy$Health, decreasing=TRUE), ][1:78, ],
            pop2 = happy[order(happy$Health, decreasing=TRUE), ][79:156, ])

par(mfrow = c(1, 2))
hist(pop[[1]]$Score, col=adjustcolor("firebrick", 0.7), freq = FALSE,
     xlab = "Happiness Score", main = "Healthy Countries", xlim = c(2, 8))
abline(v = mean(pop[[1]]$Score), col = "darkgreen", lwd = 2)
hist(pop[[2]]$Score, col=adjustcolor("blue", 0.7), freq = FALSE,
     xlab = "Happiness Score", main = "Unhealthy Countries", xlim = c(2, 8))
abline(v = mean(pop[[2]]$Score), col = "darkgreen", lwd = 2)
```



(b) State the null hypothesis H_0 that is being tested when comparing these two sub-populations with a permutation test.

$H_0 : \mathcal{P}_1 and \mathcal{P}_2$ are both drawn from the same population of happiness scores

(c) In this question you will test the hypothesis in (b) using the discrepancy measure

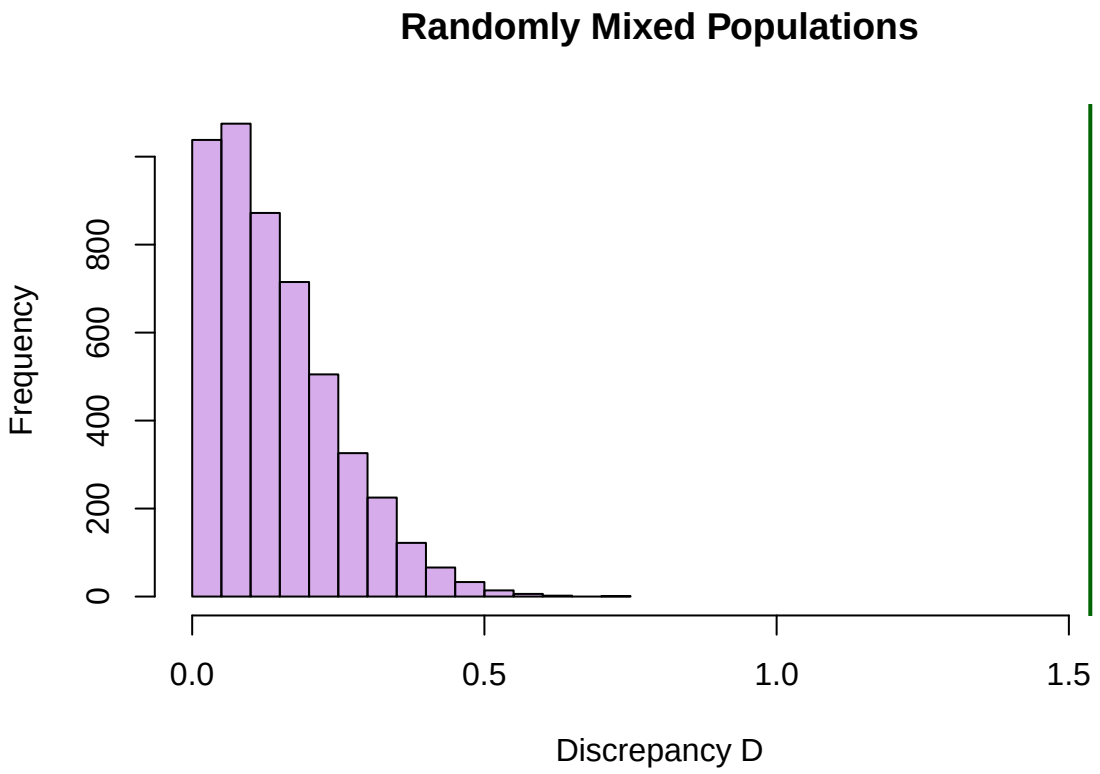
i. Calculate the observed discrepancy.

```
D <- function(pop) {
  abs(mean(pop[[1]]$Score) - mean(pop[[2]]$Score))
}
d_obs <- D(pop)
print(paste0("Obs. discrepancy: ", d_obs)) # observed discrepancy
```

```
## [1] "Obs. discrepancy: 1.53670512820513"
```

ii. Randomly mix the populations $M = 5,000$ times and construct a histogram of the 5,000 $D(\mathcal{P}_1^\star, \mathcal{P}_2^\star)$ values. Indicate, with a vertical line, the observed discrepancy calculated in i. Note that you may use the `mixRandomly` function from class.

```
diffPops <- sapply(1:5000, FUN = function(...) {
  D(mixRandomly(pop))
})

hist(diffPops, breaks = 20, main = "Randomly Mixed Populations",
     xlab = "Discrepancy D", col = adjustcolor("darkorchid", 0.4),
     xlim=c(0, 1.6))
abline(v = D(pop), col = "darkgreen", lwd = 2)
```



iii. Calculate the p-value associated with this test.

```
print(paste0("p-value: ", mean(diffPops >= D(pop))))
```

```
## [1] "p-value: 0"
```

iv. Based on the p-value calculated in iii. what do you conclude about the comparability of these two populations? In other words, summarize your findings and draw a conclusion about the null hypothesis from part (b). By referring to the histograms constructed in part (a), explain why this conclusion is, or is not, surprising.

Since the p-value is 0, there is very strong evidence against the null hypothesis that the healthy and unhealthy happiness scores are indistinguishable.

This is not suprising since all the values from the histogram are less than the observed discrepancy so it was expected this conclusion will occur

(d) In this question you will test the hypothesis in (b) using the discrepancy measure

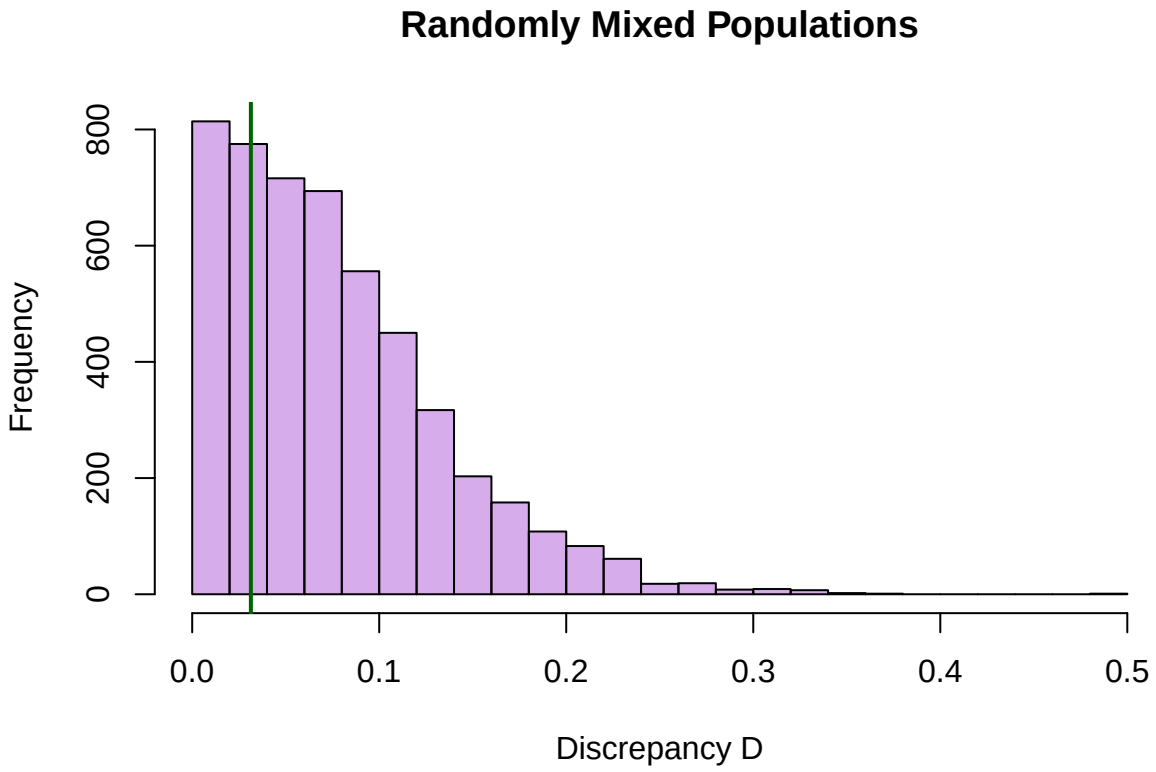
i. Calculate the observed discrepancy.

```
D1 <- function(pop) {
  abs(sd(pop[[1]]$Score) / sd(pop[[2]]$Score) - 1)
}
d1_obs <- D1(pop)
print(paste0("Obs. discrepancy: ", d1_obs))
```

```
## [1] "Obs. discrepancy: 0.031389688770509"
```

ii. Randomly mix the populations $M = 5,000$ times and construct a histogram of the 5,000 $D(\mathcal{P}_1^\star, \mathcal{P}_2^\star)$ values. Indicate, with a vertical line, the observed discrepancy calculated in i. Note that you may use the `mixRandomly` function from class.

```
diffPops1 <- sapply(1:5000, FUN = function(...) {
  D1(mixRandomly(pop))
})

hist(diffPops1, breaks = 20, main = "Randomly Mixed Populations",
     xlab = "Discrepancy D", col = adjustcolor("darkorchid", 0.4))
abline(v = D1(pop), col = "darkgreen", lwd = 2)
```



iii. Calculate the p-value associated with this test.

```
print(paste0("p-value: ", mean(diffPops1 >= D1(pop))))
```

```
## [1] "p-value: 0.7446"
```

iv. Based on the p-value calculated in iii. what do you conclude about the comparability of these two populations? In other words, summarize your findings and draw a conclusion about the null hypothesis from part (b). By referring to the histograms constructed in part (a), explain why this conclusion is, or is not, surprising.

Since the p-value is 0.743, there is no evidence against the null hypothesis that the healthy and unhealthy happiness scores are indistinguishable.

This is suprising since several values from the histogram are less than the observed discrepancy, I was expecting a p-value that will conclude that there is little evidence against the null hypothesis.