Exercise 2

EMAIL: n.c.deschipper

Question 1 part 1 1.1 population standard error of the mean difference, you can estimate this with SE of student t and SE of welch

1.2 $\sqrt{\frac{208}{208}}$

1.3 Input: S: integer deining the number of independent data sets to generate n: the sample size of the data sets μ, and σ^2 vlaues fo the population parameters ofthe data generating distribution Output: S estimates o the statistics of interest, namely bias, variance, MSE and RE

1. initialize out students_bias, students_var, students_mse, welch_bias, welch_var, welch_mse and relative_efficiency all as vector of length S
2. For s in 1:S: A. Generate data with sample size n1 from N(μ_1 , σ^2_1) and sample size n2 from N(μ_2 , σ^2_2) B. Obtain sample variance, bias and mse for both
3. $intializebias\_student, variance\_student, mse\_student, bias\_welch, variance\_welch, mse\_welchallasvectoroflentgh$

```r
set.seed(200)
n1 <- c(10, 100, 200)
n2 <- 200
var2 <- c(1, 2 , 10)
var1 <- 1
mean1 <- 0
mean2 <- 1
S <- 10000
conditions <- expand.grid(n1, var2)

MonteCarlo_SE <- function(n1, var2) {
  data_1 <- rnorm(n1, mean1, sqrt(var1))
  data_2 <- rnorm(n2, mean2, sqrt(var2))
  SE_student <- rep(NA, S)
  SE_welch <- rep(NA, S)
  for (i in 1:S) {
    data_1 <- rnorm(n1, mean1, sqrt(var1))
    data_2 <- rnorm(n2, mean2, sqrt(var2))
    pooled_sd <- sqrt(((n1-1)*var(data_1)+(n2-1)*var(data_2))/(n1+n2-2))
    SE_student[i] <- pooled_sd*sqrt(((1/n1)+(1/n2)))
    SE_welch[i] <- sqrt((var(data_1)/n1)+(var(data_2)/n2))
  }
  bias_Student <- sqrt((var1/n1)+(var2/n2)) - mean(SE_student)
  bias_Welch <- sqrt((var1/n1)+(var2/n2)) - mean(SE_welch)
  Out_variance_student <- var(SE_student)
  Out_variance_welch <- var(SE_welch)
  Out_bias_student <- bias_Student
  Out_bias_welch <- bias_Welch
  Out_MSE_student <- (bias_Student)^2 + Out_variance_student
  Out_MSE_welch <- (bias_Welch)^2 + Out_variance_welch
  RE <- (sum(SE_welch)/sum(SE_student))
  return(c(Out_variance_student, Out_variance_welch, Out_bias_student, Out_bias_welch, Out_MSE_student,
}
```

```r
results <- matrix(NA, 7, 9)
for (i in 1:nrow(conditions)) {
  results[, i] <- MonteCarlo_SE(n1 = conditions[i, 1], var2 = conditions[i,2])
}

out_results <- cbind(conditions, t(round(results, digits = 3)))
colnames(out_results) <- c("n1", "var2", "Variance_S", "variance_W", "Bias_S", "Bias_W", "MSE_S", "MSE_W
out_results
```

```
##    n1 var2 Variance_S variance_W Bias_S Bias_W MSE_S MSE_W
## 1  10    1      0.000      0.005  0.000  0.009 0.000 0.005
## 2 100    1      0.000      0.000  0.000  0.000 0.000 0.000
## 3 200    1      0.000      0.000  0.000  0.000 0.000 0.000
## 4  10    2      0.000      0.005 -0.121  0.008 0.015 0.005
## 5 100    2      0.000      0.000 -0.017  0.000 0.000 0.000
## 6 200    2      0.000      0.000  0.000  0.000 0.000 0.000
## 7  10   10      0.003      0.003 -0.616  0.005 0.382 0.004
## 8 100   10      0.000      0.000 -0.079  0.000 0.006 0.000
## 9 200   10      0.000      0.000  0.000  0.000 0.000 0.000
##   Relative Efficiency
## 1               0.974
## 2               1.000
## 3               1.000
## 4               0.716
## 5               0.895
## 6               1.000
## 7               0.381
## 8               0.755
## 9               1.000
```

1.2 Use t-test r-function to obtain p value Null hypothesis: The means of the two t-tests are the same Alternative Hypothesis: The means of the two t-tests are not the same

2.1 Null hypothesis: The means of the two t-tests are the same Alternative Hypothesis: The means of the two t-tests are not the same

2.2

```r
n1 <- c(10, 100, 200)
n2 <- 200
var1 <- 2
var2 <- c(1, 2, 10)
mean1 <- 0
mean2 <- 1
conditions <- expand.grid(n1, var2)
```