

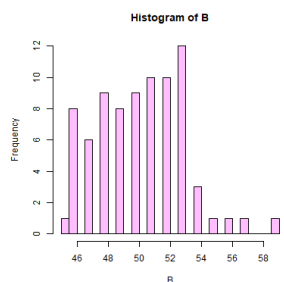
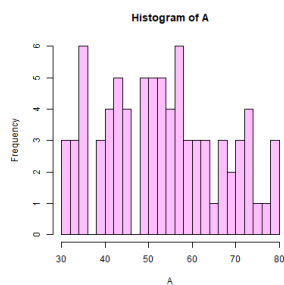
統計分析法 第7週レポート

202212022 田島瑞起

2023/12/05

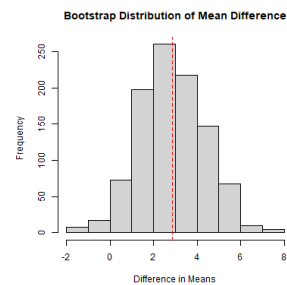
1 設問 1-1

A,B についてのヒストグラムを下記の図で示す。



2 設問 1-2

今回得られた A 群 B 群の平均値差の実測値は,2.87 である。A 群 B 群をブートストラップ処理し,二群間の平均値差の理想分布を図示し,実測値を添付した結果,下記の図となった。



3 設問 1-3

今回の帰無仮説は「A,B 群に平均値の差がない」を設定し,有意水準を両側検定の為 0.10 に設定する。ブートストラップ処理を実行し,両側検定の p 値を求めると $p=0.976$ と表示された。これは有意水準を大きく上回る値であるため,帰無仮説を棄却することは出来ない。よって A,B 群の平均値に差がないと言うことが出来る。

4 設問 1-4

t-test を用いて今回のデータセットについて検証したところ下記の結果が表示された。ブートストラップ処理とは結果が大きく異なる。この際考えられる要因としては,t 検定では正規分布を仮定した検証方法であるため,データの非対称性や外れ値に敏感に反応してしまう可能性がある。これ

に対して復元抽出して理想分布を作成するブートストラップ処理では、非対称性や外れ値に強いロバストな手法であるため、t-検定よりも大きな p 値を示したと考えられる。

図 1 s2212022-1.c

```
1      Welch Two Sample t-test
2
3      data:  A and B
4      t = 1.8629, df = 86.06, p-value
        = 0.06589
5      alternative hypothesis: true
        difference in means is not
        equal to 0
6      95 percent confidence interval:
7      -0.1923156  5.9231156
8      sample estimates:
9      mean of x mean of y
10     53.1654   50.3000
```

5 設問 1 ソースコード

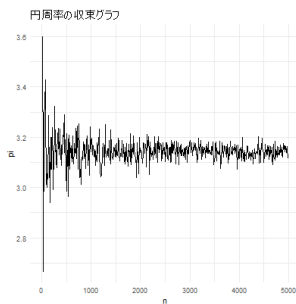
図 2 s2212022-1.c

```
1      #課題1-1
2      setwd('Z:/stats_work')
3      ReadData <- read.table('data_p.txt')
4      A <- ReadData$V1
5      B <- ReadData$V2
6      A <- A[2:length(A)]
7      B <- B[2:length(B)]
8      A <- as.numeric(A)
9      B <- as.numeric(B)
10     png("7-1-1.png", width = 400, height
        = 400)
11     hist(A,breaks=20,col='#ff00ff40')
12     dev.off()
13     png("7-1-2.png", width = 400, height
        = 400)
14     hist(B,breaks=20,col='#ff00ff40')
15     dev.off()
16
17     #課題1-2,1-3
18
19     # ライブラリの読み込み
20     library(boot)
21
22     # データの読み込み
23     setwd('Z:/stats_work')
24     ReadData <- read.table('data_p.txt')
```

```
25     A <- as.numeric(ReadData$V1[-1])
26     B <- as.numeric(ReadData$V2[-1])
27
28     # ブートストラップサンプリング関数
29     bootstrap_function <- function(data,
        indices) {
30         sample_A <- data$A[indices]
31         sample_B <- data$B[indices]
32         diff_mean <- mean(sample_A) - mean
            (sample_B)
33         return(diff_mean)
34     }
35
36     # オリジナルデータ
37     obs_diff_mean <- mean(A) - mean(B)
38
39     # ブートストラップ法
40     set.seed(22) # 再現性のためにシード
        を設定
41     bootstrap_results <- boot(data.frame
        (A, B), statistic =
        bootstrap_function, R = 1000)
42
43     # ブートストラップ結果からp値の計算
44     p_value <- 2 * min(mean(
        bootstrap_results$t >=
        obs_diff_mean), mean(
        bootstrap_results$t <=
        obs_diff_mean))
45
46     # ブートストラップ結果のヒストグラム
        (理論分布)
47     png("7-1-3.png", width = 400, height
        = 400)
48     hist(bootstrap_results$t, main = "
        Bootstrap Distribution of Mean
        Difference", xlab = "Difference in
        Means")
49     abline(v = obs_diff_mean, col = "red",
        lty = 2) # オリジナルデータの
        平均差
50     dev.off()
51
52     # p値の表示
53     cat("Observed Difference in Means:",
        obs_diff_mean, "\n")
54     cat("Bootstrap p-value:", p_value, "
        \n")
55
56     #課題1-4
57     t.test(A, B)
```

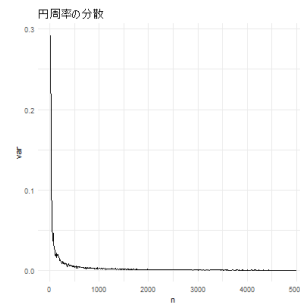
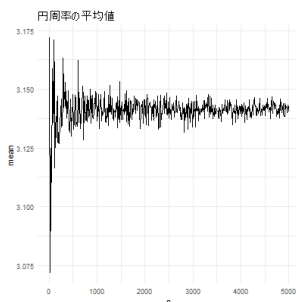
6 設問 2-1

乱数の個数に応じた円周率の近似結果をモンテカルロシミュレーションで実行し、個数に応じた近似値をプロットした結果下記の図の通りとなった。



7 設問 2-2

乱数の個数に応じた円周率の近似結果をモンテカルロシミュレーションで乱数の個数ごとに 100 回円周率を近似し、その 100 回で得られた円周率の平均値と分散を、乱数の個数に応じてプロットした結果下記の図の通りとなった。



8 設問 2-3

設問 2-1 の結果より k の値が大きくなれば正確な円周率の値に収束する。設問 2-2 の結果より k の値が大きくなれば円周率の平均値は正確な円周率の値に収束し、分散は 0 に収束する。

9 設問 1 ソースコード

図 3 s2212022-1.c

```
1 #課題2-1
2
3 estimate_pi <- function(num_samples)
4 {
5   # ランダムに点をサンプリング
6   x <- runif(num_samples, -1, 1)
7   y <- runif(num_samples, -1, 1)
8   # 円内に入る点の数を数える
9   inside_circle <- x^2 + y^2 <= 1
10  num_inside <- sum(inside_circle)
11  # 円周率の推定値を計算
12  pi_estimate <- 4 * (num_inside /
13    num_samples)
14
15  return(pi_estimate)
16
17 # サンプル数を指定して円周率を求める
18 n_array <- seq(10,5000,10)
19 pi_array <- numeric(length(n_array))
20 for( k in n_array){
21   i <- k / 10
22   pi_array[i] <- estimate_pi(k)
23 }
```

```

24 install.packages("ggplot2")
25 # ライブラリの読み込み
26 library(ggplot2)
27 # グラフの作成
28 df <- data.frame(n = n_array, pi =
    pi_array)
29 # 折れ線グラフの描画
30 png("7-2-1.png",width=400,height
    =400)
31 ggplot(df, aes(x = n, y = pi)) +
32   geom_line() +
33   labs(title = "円周率の収束グラフ",
34         x = "n",
35         y = "pi") +
36   theme_minimal()
37 dev.off()
38
39 #課題2-2
40 pi_mean_array <- numeric(length(
    n_array))
41 pi_var_array <- numeric(length(
    n_array))
42 r <- 100
43
44 for( k in n_array){
45   pi_array_tmp <- numeric(100)
46   i <- k / 10
47   for( j in 1:r){
48     pi_array_tmp[j] <-
        estimate_pi(k)
49   }
50   pi_mean_array[i] <- mean(
        pi_array_tmp)
51   pi_var_array[i] <- var(
        pi_array_tmp)
52 }
53
54 # グラフの作成
55 df <- data.frame(n = n_array, mean =
    pi_mean_array)
56 # 折れ線グラフの描画
57 png("7-2-2.png",width=400,height
    =400)
58 ggplot(df, aes(x = n, y = mean)) +
59   geom_line() +
60   labs(title = "円周率の平均値",
61         x = "n",
62         y = "mean") +
63   theme_minimal()
64 dev.off()
65
66 # グラフの作成
67 df <- data.frame(n = n_array, var =
    pi_var_array)

```

```

68 # 折れ線グラフの描画
69 png("7-2-3.png",width=400,height
    =400)
70 ggplot(df, aes(x = n, y = var)) +
71   geom_line() +
72   labs(title = "円周率の分散",
73         x = "n",
74         y = "var") +
75   theme_minimal()
76 dev.off()

```