

## 7. シミュレーションを利用して大数の法則を理解する

honocat

2025-12-18

### 大数の法則(Law of Large Numbers; LLN)

「公正な」コインを使って、大数の法則のシミュレーションを行う。

1. 表が出れば 1
2. 裏が出れば 0

コイン投げの回数が少ないと、表が出る比率は真の比率である 0.5 に近くなるとは限らない。しかし、大数の法則によると、コインを投げる回数を十分大きくすると、表の比率は 0.5 に近づくはずである。確かめてみよう。

```
coin <- c('表', '裏')
n_flips <- 500 # コインを投げる回数
ratio_1 <- rep(NA, n_flips) # 結果の記録
```

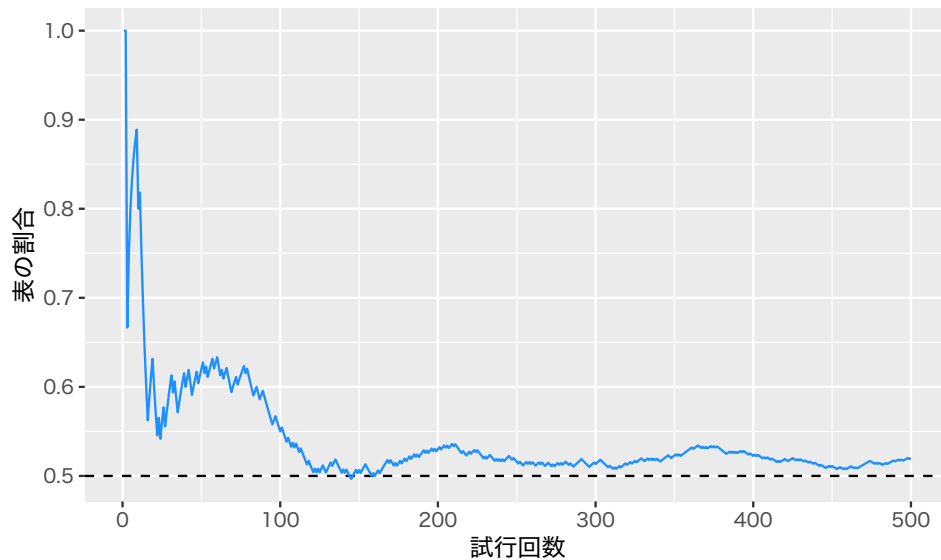
n\_flip 回コイン投げを行い、それぞれのコイン投げが終わった時点での表の比率を計算する。

```
coins_1 <- sample(coin, size = n_flips, replace = TRUE)
for (i in 1 : n_flips) {
  n_head <- sum(coins_1[1 : i] == '表') # i 回目までに何回表が出たか
  ratio_1[i] <- n_head / i              # i 回目までの表の比率を計算
}
```

結果を図示する。

```
df2 <- tibble(N = 1 : n_flips,
              ratio_1 = ratio_1)
p1 <- ggplot(df2, aes(x = N, y = ratio_1)) +
  geom_hline(yintercept = 0.5,
             linetype = 'dashed') +
```

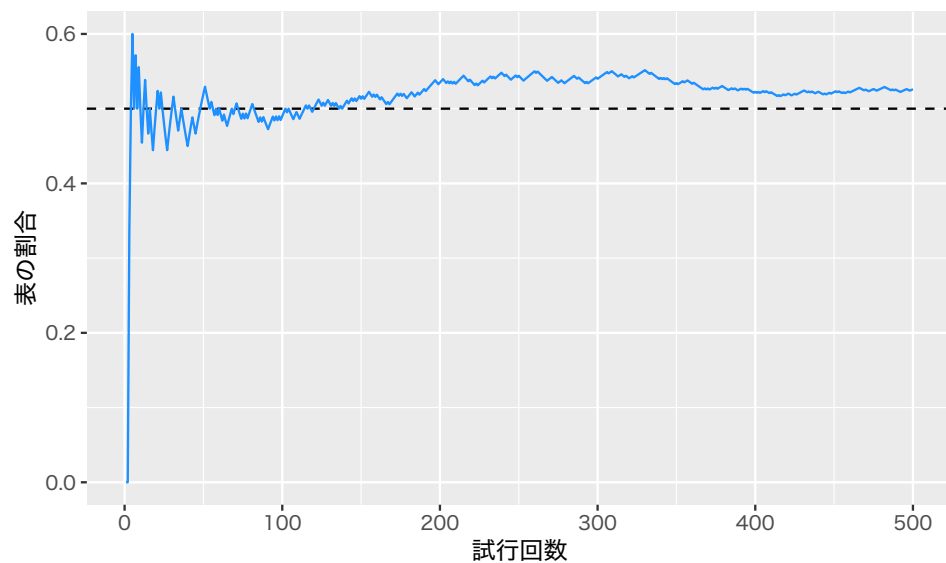
```
geom_line(color = 'dodgerblue') +
  labs(x = ' 試行回数',
        y = ' 表の割合')
plot(p1)
```



比率が少しずつ 0.5 に近づいていくことがわかる。

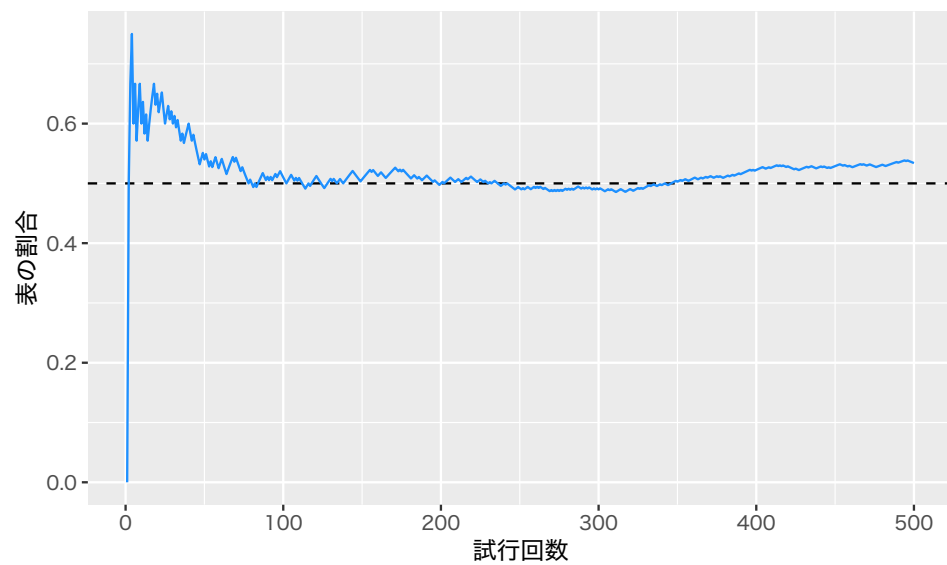
もう一度。

```
ratio_2 <- rep(NA, n_flips)
coins_2 <- sample(coin, size = n_flips, replace = TRUE)
for (i in 1 : n_flips) {
  n_head <- sum(coins_2[1 : i] == '表')
  ratio_2[i] <- n_head / i
}
df2$ratio_2 <- ratio_2
p2 <- ggplot(df2, aes(x = N, y = ratio_2)) +
  geom_hline(yintercept = 0.5,
             linetype = 'dashed') +
  geom_line(color = 'dodgerblue') +
  labs(x = ' 試行回数',
        y = ' 表の割合')
plot(p2)
```



もう一度。

```
ratio_3 <- rep(NA, n_flips)
coins_3 <- sample(coin, size = n_flips, replace = TRUE)
for (i in 1 : n_flips) {
  n_head <- sum(coins_3[1 : i] == '表')
  ratio_3[i] <- n_head / i
}
df2$ratio_3 <- ratio_3
p3 <- ggplot(df2, aes(x = N, y = ratio_3)) +
  geom_hline(yintercept = 0.5,
             linetype = 'dashed') +
  geom_line(color = 'dodgerblue') +
  labs(x = '試行回数',
       y = '表の割合')
plot(p3)
```



シミュレーションを実行するたびに、異なる軌跡を描きながら、比率が 0.5 に近づいていく様子が見て取れる。