L3:Simulation and probability (working)

Learning objectives

- Understand and Compute Expectations and Variances: Explain the concepts of expectations and variances of random variables, and compute the expectation and variance of a linear combination of random variables.

Lessons

1 - objectives 1

- Rolling dice, if r.v. X is the sum of two rolled dice.
- Then what is $E(X)$ and $Var(X)$? (theoretical) - this can be explained through slides
- What do we achieve?
- How do we do it?
  - How to roll the dice?
  - How to collect the data?
  - What do we show?
- `Sample()`
  - `sample(x, size, replace = FALSE, prob = NULL)`
  - sample takes a sample of the specified size from the elements of x using either with or without replacement.

```
# for loops
# how to make a function
# usage of replicate() function
# usage of sample function
```

How to rolle a dice ?

```
oneRoll <- sample(1:6, 1, TRUE)
oneRoll
```

```
## [1] 4
```

Sum of 2 dice?

```
firstRoll <- sample(1:6, 1, TRUE)
secondRoll <- sample(1:6, 1, TRUE)
sum(firstRoll + secondRoll)
```

```
## [1] 8
```

```
# or
sum_of_two <- sum(sample(1:6, 2, TRUE))
sum_of_two
```

```
## [1] 9
```

Theoretical sol (by definition)

```
Number_ways <- c(1,2,3,4,5,6,5,4,3,2,1)
prob <- (1/36) * Number_ways # theoretical prob
x <- 2:12
mu = sum(x * prob)
mu
```

```
## [1] 7
```

```
sigma_2 = sum((x-mu)^2*prob)
sigma_2
```

```
## [1] 5.833333
```

2 - Simulation

We can build a function to show if simulated values are close to theoretical values.

```
n = 1000 #sample size
sim_data = rep(0,n)
for (i in 1:n){
    sum_of_two <- sum(sample(1:6, 2, TRUE))
    sim_data[i] = sum_of_two
}

head(sim_data)
```

```
## [1]  6 10  8  8 10 12
```

We can make this using `replicate()` function. (This way is faster than for loop)

```
n = 1000 #sample size
sim_data <- replicate(n,{
    sum_of_two <- sum(sample(1:6, 2, TRUE))
    sum_of_two
})

head(sim_data)
```

```
## [1] 11  8 10  7  8  7
```

```

sim_table<-table(sim_data)
sim_table
```

What is simulated $E(X)$ and $Var(X)$?

```
## sim_data
##   2   3   4   5   6   7   8   9  10  11  12
##  37  55  70 114 140 172 161  89  79  51  32
```

```
x <- 2:12 # possible values
prob_est = sim_table/n
prob_est
```

```
## sim_data
##     2     3     4     5     6     7     8     9    10    11    12
## 0.037 0.055 0.070 0.114 0.140 0.172 0.161 0.089 0.079 0.051 0.032
```

```
mu_hat = sum(prob_est * x)
mu_hat
```

```
## [1] 6.957
```

```
sigma_2_hat = sum( (x-mu_hat)^2*prob_est )
sigma_2_hat
```

```
## [1] 5.873151
```

Let's make a function to do a simulation.

```
sim_fn <- function(n=1000){
sim_data <- replicate(n,{
    sum_of_two <- sum(sample(1:6, 2, TRUE))
    sum_of_two
})

sim_table <- table(factor(sim_data, levels = x))  # Ensure all values appear (important to aviod error)

x <- 2:12 # possible values
prob_est = sim_table/n
mu_hat = sum(prob_est * x)
sigma_2_hat = sum( (x-mu_hat)^2*prob_est )

return(c(mu_hat, sigma_2_hat))
}

sim_fn(1000)
```

```
## [1] 6.927000 5.463671
```

```
sim_fn(100)
```

```
## [1] 6.9900 5.0499
```

```
sim_fn(1000)
```

```
## [1] 6.952000 5.439696
```

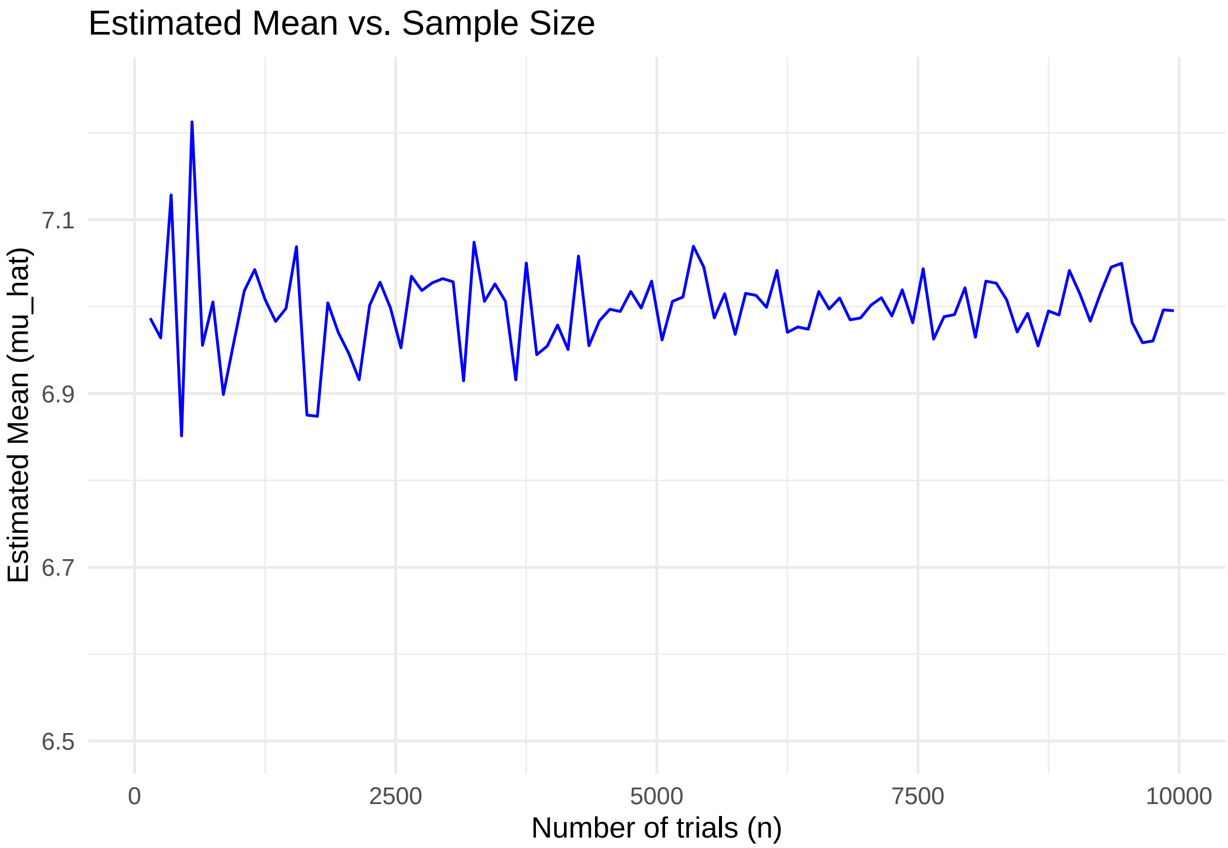
```
sim_fn(10000)
```

```
## [1] 7.011700 5.793563
```

```
n_values = seq(50, 10000, 100)
sim_mu = rep(0, length(n_values))
sim_sigma_2 = rep(0, length(n_values))

for (i in 1:length(n_values)){
    sim_results <- sim_fn(n_values[i])
    sim_mu[i] = sim_results[1]
    sim_sigma_2[i] = sim_results[2]
}

sim_data <- data.frame(n_values, sim_mu)

# mu hat plot
ggplot(sim_data, aes(x = n_values, y = sim_mu)) +
```

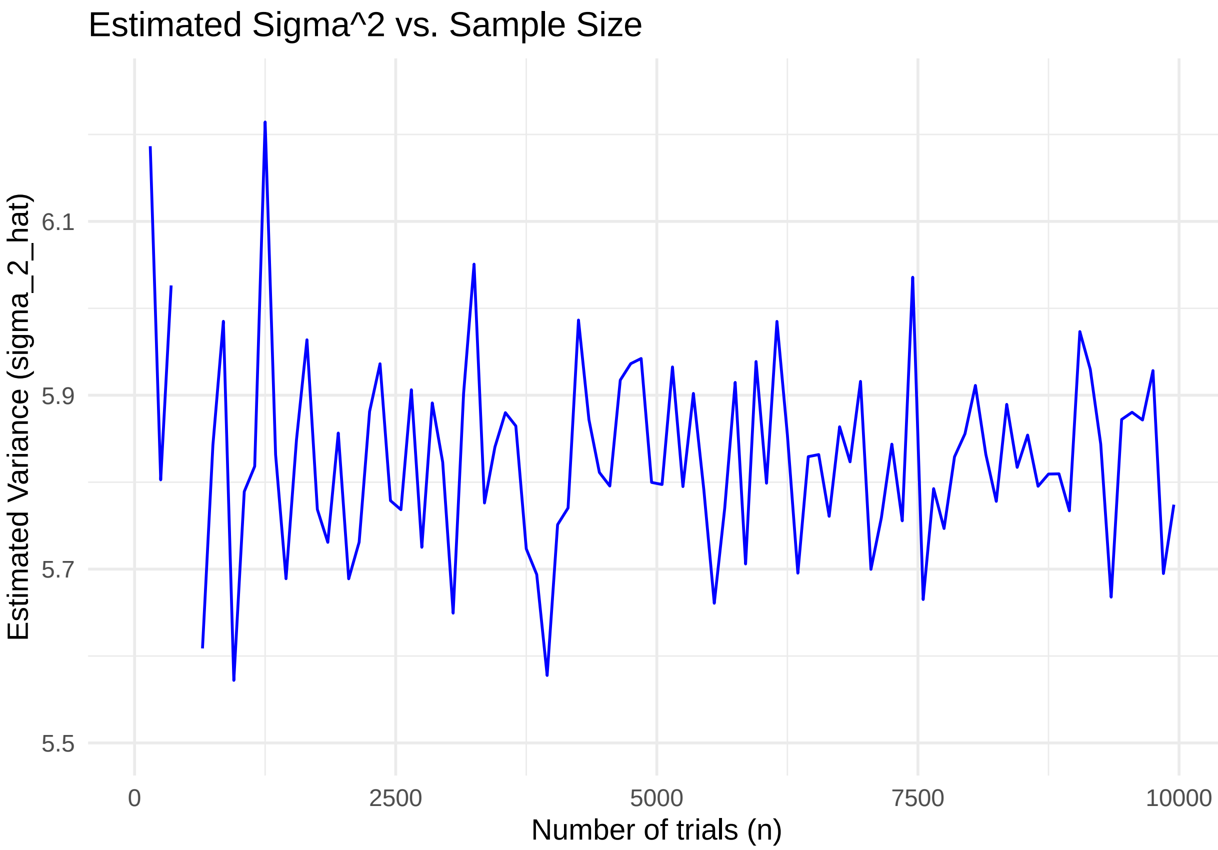
```r
  ylim(6.5,7.25) +
  geom_line(color = "blue") +
  labs(title = "Estimated Mean vs. Sample Size",
       x = "Number of trials (n)",
       y = "Estimated Mean (mu_hat)") +
  theme_minimal()
```

```
## Warning: Removed 1 row containing missing values (`geom_line()`).
```




```r
sim_data <- data.frame(n_values, sim_sigma_2)
# sigma^2 hat plot
ggplot(sim_data, aes(x = n_values, y = sim_sigma_2)) +
  ylim(5.5,6.25) +
  geom_line(color = "blue") +
  labs(title = "Estimated Sigma^2 vs. Sample Size",
       x = "Number of trials (n)",
       y = "Estimated Variance (sigma_2_hat)") +
  theme_minimal()
```

```
## Warning: Removed 1 row containing missing values (`geom_line()`).
```



3 - Activity (Linear combination of r.v.)

Let the random variable X be the sum of three rolled dice. However, this time, the first die has triple its value, the second die has double its value, and the last die remains unchanged. Can we simulate and verify whether the estimated $E(X)$ and $Var(X)$ are close to their true values?

1. What are possible values of X?
2. What are $E(X)$ and $Var(X)$?

3. How do you obtain one trial in R? and simulated trials with $n = 1000$?

4. Let Y_i for $i = 1, 2, 3$ be a roll of one die. Then can we say that $3E(Y_1) + 2E(Y_2) + E(Y_3) = E(3Y_1 + 2Y_2 + Y_3)$? How about $Var(3Y_1 + 2Y_2 + Y_3)$?

```
sim_fn <- function(n=1000){
sim_data <- replicate(n,{
    die1 <- 3*sample(1:6, 1, TRUE)
    die2 <- 2*sample(1:6, 1, TRUE)
    die3 <- sample(1:6, 1, TRUE)
    sum_of_three <- die1 + die2 + die3
})

x <- 6:36 # possible values
sim_table <- table(factor(sim_data, levels = x))  # Ensure all values appear (important to aviod error)


prob_est = sim_table/n
mu_hat = sum(prob_est * x)
```

```r
sigma_2_hat = sum( (x-mu_hat)^2*prob_est )

return(c(mu_hat, sigma_2_hat))
}

sim_fn(1000)
```

```
## [1] 20.93400 39.63364
```