Mini Project #1

STA302H Summer 2020

Ruo Ning Qiu

University of Toronto

May 22, 2020

Outline

Outline

Introduction

In this project, we are running several simulations to examine the properties of least square regression, its inference tests, and how it models the true population relation.

Our goal is the following:

To observe and understand

1. Whether the construction of confidence interval produces consistent results with its definition
2. How sample size affects the estimators (mean, variance...)
3. How error variance affects the estimators

Set up simulation

First define the population parameters with my student number seed.

```
set.seed(1004079631)
beta0 <- rnorm(n = 1, mean = 0, sd = 1)  # The population beta_0
beta1 <- runif(n = 1, min = 1, max = 3)  # The population beta_1
sig2 <- rchisq(n = 1, df = 25)  # The error variance sigma^2

nsample <- 5  # Sample size
n.sim <- 1000  # The number of simulations
sigX <- 0.2  # The variances of X
```

To run 1000(n.sim) simulations, we need to define some vectors for b_0, b_1, s^2 to assign what each simulation produces by a for loop.

```
X <- rnorm(n = nsample, mean = 0, sd = sqrt(sigX))  #Simulate the predictor variable

b0 <- vector()  # saves the sample estimates of beta_0
b1 <- vector()  # saves the sample estimates of beta_1
sig2hat <- vector()  # saves the sample estimates of error variance sigma^2

for(i in 1:n.sim){  # Assign the estimators for $n.sim$ samples individually
  Y <- beta0 + beta1*X + rnorm(n = nsample, mean = 0, sd = sqrt(sig2))
  model <- lm(Y ~ X)
  b0[i] <- coef(model)[1]
  b1[i] <- coef(model)[2]
  sig2hat[i] <- summary(model)$sigma^2
}
```

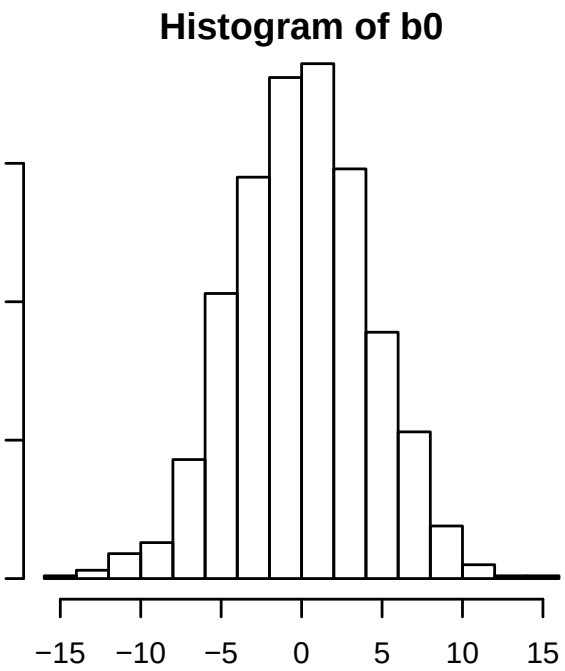
The mean of the estimates from the 1000 different simulations

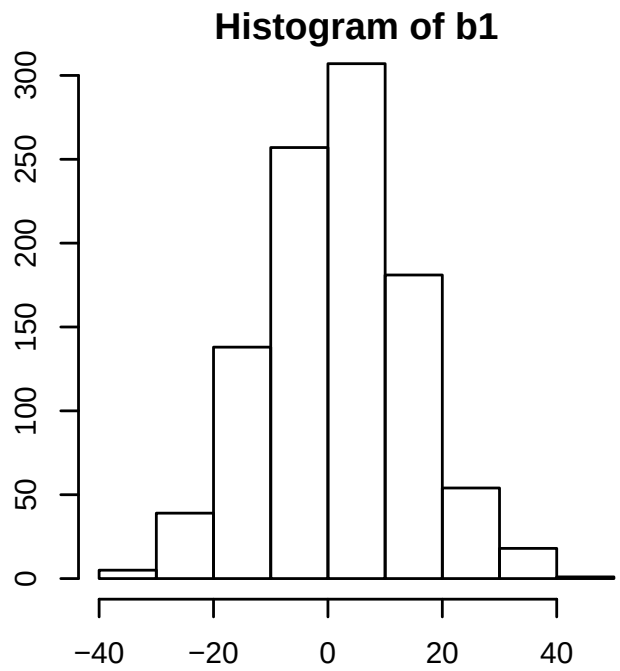
Let's have a look at the mean of estimators b_0, b_1, s^2 from the 1000 different simulations.

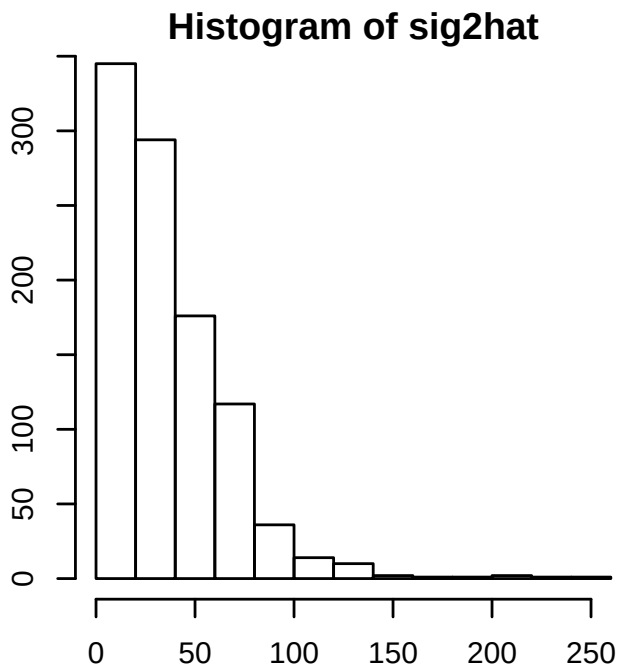
```
## Q2 ##
# The mean of esitmators b0, b1, and error variance s^2 for sample size = nsample with 100 simulations
mean(b0); mean(b1); mean(sig2hat)

## [1] -0.024439
## [1] 1.788608
## [1] 36.43437

# The true population parameters beta_0, beta_1, and error variance sigma^2
beta0; beta1; sig2

## [1] 0.009603152
## [1] 1.716823
## [1] 35.60787
```

The histograms of distributions for the 1000 sample estimators

```
## Q3 ##
# Construct hisotgrams of estiamtors for b0, b1, and error variance s^2 with $n.sim$ samples
par(mar=c(1,1,1,1)); par(mfrow = c(3,3))  # set up the margin of the plots
hist(b0); hist(b1); hist(sig2hat)
```







The mean of the variance of the regression parameter estimators for each simulation

```
## Q4 ##
# Calculate the true variance of b0 and b1
sumx <- sum(X); xbar <- sumx/nsample
sumx2 <- sum(X^2); SXX <- sumx2 - nsample*(xbar^2)
var_beta_0 <- sig2*(1/nsample + xbar^2/SXX)
var_beta_1 <- sig2/SXX

var_beta_0; var_beta_1  # true variance of b0 and b1

## [1] 17.44075
## [1] 159.5347

# Save sample variance of b0, b1 for each simulation
var_b0 <- vector()
var_b1 <- vector()
for(i in 1:n.sim){  # Assign the var estimators of b0, b1 for $n.sim$ samples individually
  var_b0[i] <- sig2hat[i]*(1/nsample + xbar^2/SXX)
  var_b1[i] <- sig2hat[i]/SXX
}

mean(var_b0); mean(var_b1)  # the mean of the sample variances of b0 and b1

## [1] 17.84557
## [1] 163.2377
```

95% Z Confidence Intervals for $\hat{\beta}_0$ and $\hat{\beta}_1$

```
## Q5 ##
# 95% Z-test CI, call it CI_z
ll_b0_z <- vector(); ul_b0_z <- vector()  # Save lower level, upper level of CI_z for b0
ll_b1_z <- vector(); ul_b1_z <- vector()  # Save lower level, upper level of CI_z for b1
counts_b0_z <- 0; counts_b1_z <- 0  # Save how many CI_z the true reg parameters do fall into
z_95 <- qnorm(.025, lower.tail=FALSE)  # z score for 95% CI

for(i in 1:n.sim){  # calculate 95% CI (Z-test) for b0, b1
  ll_b0_z[i] <- b0[i] - z_95*sqrt(var_beta_0); ul_b0_z[i] <- b0[i] + z_95*sqrt(var_beta_0)
  ll_b1_z[i] <- b1[i] - z_95*sqrt(var_beta_1); ul_b1_z[i] <- b1[i] + z_95*sqrt(var_beta_1)

  # Check if CI_z contains the true value of the parameters
  if ((ll_b0_z[i] <= beta0) && (beta0 <= ul_b0_z[i])) {
    counts_b0_z <- counts_b0_z + 1
  }
  if ((ll_b1_z[i] <= beta1) && (beta1 <= ul_b1_z[i])) {
    counts_b1_z <- counts_b1_z + 1
  }
}

counts_b0_z/n.sim; counts_b1_z/n.sim # Precentatge of the true value of regression parameters fall into CI_z

## [1] 0.951
## [1] 0.945
```

95% t Confidence Intervals for $\hat{\beta}_0$ and $\hat{\beta}_1$

```
# 95% T-test CI, call it CI_t
# Save lower level, uppler level of CI_t for b0, b1
ll_b0_t <- vector(); ul_b0_t <- vector()
ll_b1_t <- vector(); ul_b1_t <- vector()
counts_b0_t <- 0; counts_b1_t <- 0  # Save how many CI_t the true reg parameters do fall into
t_95 <- qt(c(.025, .975), df=nsample-2)[2] # t score for 95% CI with sample size = nsample = 5, so df = nsample

for(i in 1:n.sim){  #calculate 95% CI (T-test) for b0, b1
  # b0[i] +- t_95 * var_b0[i] is the CI_t for b0[i], as var_b0[i] is the sample var calculated before for b0
  ll_b0_t[i] <- b0[i] - t_95 * sqrt(var_b0[i]); ul_b0_t[i] <- b0[i] + t_95 * sqrt(var_b0[i])
  # b1[i] +- t_95 * var_b1[i] is the CI_t for b1[i], as var_b1[i] is the sample var calculated before for b1
  ll_b1_t[i] <- b1[i] - t_95 * sqrt(var_b1[i]); ul_b1_t[i] <- b1[i] + t_95 * sqrt(var_b1[i])

  # Check if CI_t contains the true value of the parameters
  if ((ll_b0_t[i] <= beta0) && (beta0 <= ul_b0_t[i])) {
    counts_b0_t <- counts_b0_t + 1
  }
  if ((ll_b1_t[i] <= beta1) && (beta1 <= ul_b1_t[i])) {
    counts_b1_t <- counts_b1_t + 1
  }
}

counts_b0_t/n.sim; counts_b1_t/n.sim # Precentatge of the true value of regression parameters fall into CI_t

## [1] 0.956
## [1] 0.947
```

Set up 1000 simulations with sample size = 25, 50, 100

We start simulating with the sample size = 25.

```
## Q6 ##
# Start with sample size 25
set.seed(1004079631)
X_25 <- rnorm(n = 25, mean = 0, sd = sqrt(sigX))  #Simulate the predictor variable with new sample size
b0_25 <- vector()  # saves the sample estimates of beta_0 with sample size 25
b1_25 <- vector()  # saves the sample estimates of beta_1 with sample size 25
sig2hat_25 <- vector()  # saves the sample estimates of error variance sigma^2 with sample size 25

for(i in 1:n.sim){  # Assign the estimators for $n.sim$ samples individually
  Y_25 <- beta0 + beta1*X_25 + rnorm(n = 25, mean = 0, sd = sqrt(sig2))
  model_25 <- lm(Y_25 ~ X_25)
  b0_25[i] <- coef(model_25)[1]
  b1_25[i] <- coef(model_25)[2]
  sig2hat_25[i] <- summary(model_25)$sigma^2
}
```

And with the sample size = 50.

```
# Increase sample size to 50
set.seed(1004079631)
X_50 <- rnorm(n = 50, mean = 0, sd = sqrt(sigX))  #Simulate the predictor variable with new sample size
b0_50 <- vector()  # saves the sample estimates of beta_0 with sample size 50
b1_50 <- vector()  # saves the sample estimates of beta_1 with sample size 50
sig2hat_50 <- vector()  # saves the sample estimates of error variance sigma^2 with sample size 50

for(i in 1:n.sim){  # Assign the estimators for $n.sim$ samples individually with sample size 50
  Y_50 <- beta0 + beta1*X_50 + rnorm(n = 50, mean = 0, sd = sqrt(sig2))
  model_50 <- lm(Y_50 ~ X_50)
  b0_50[i] <- coef(model_50)[1]
```

Set up 1000 simulations with sample size = 25, 50, 100

Finally with the sample size = 100.

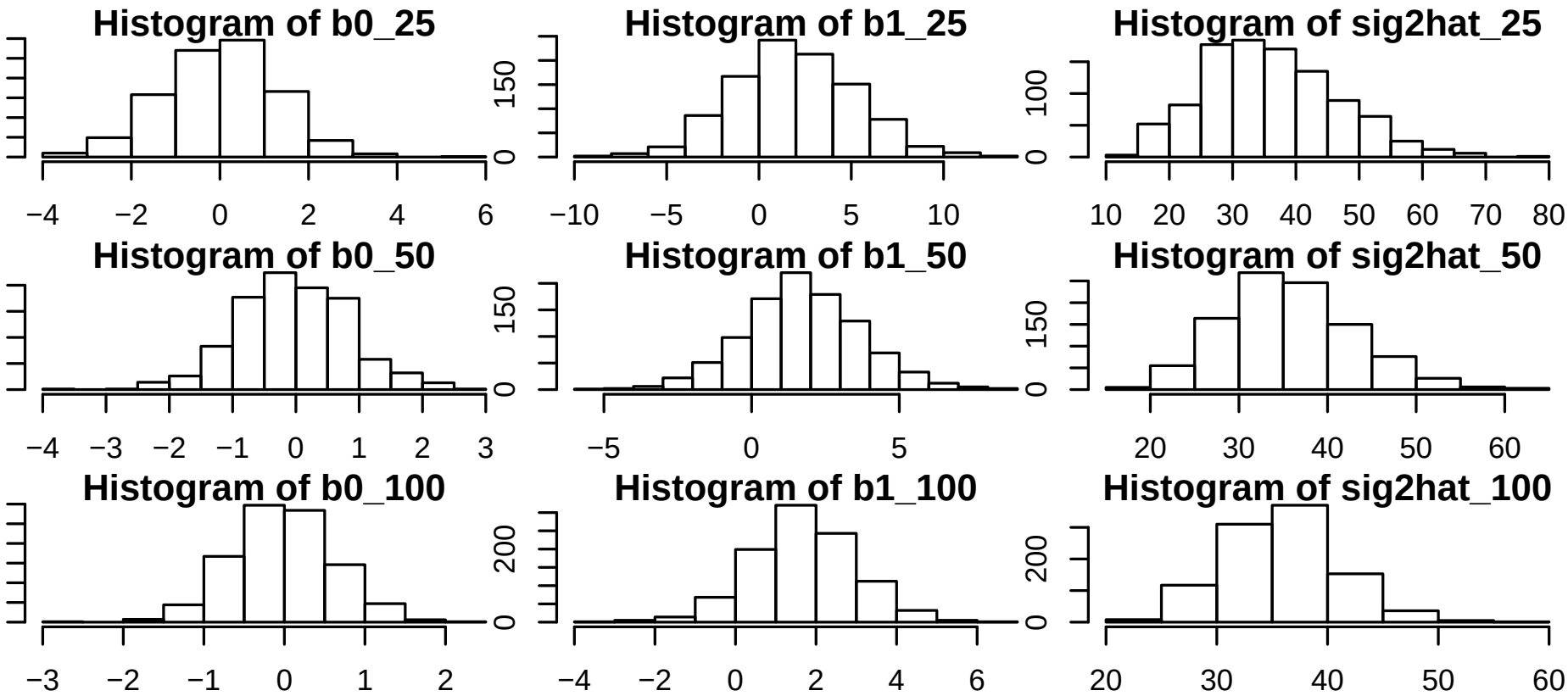
```
# Increase sample size to 100
set.seed(1004079631)
X_100 <- rnorm(n = 100, mean = 0, sd = sqrt(sigX))  #Simulate the predictor variable with new sample size
b0_100 <- vector()  # saves the sample estimates of beta_0 with sample size 100
b1_100 <- vector()  # saves the sample estimates of beta_1 with sample size 100
sig2hat_100 <- vector()  # saves the sample estimates of error variance sigma^2 with sample size 100

for(i in 1:n.sim){  # Assign the estimators for $n.sim$ samples individually with sample size 100
  Y_100 <- beta0 + beta1*X_100 + rnorm(n = 100, mean = 0, sd = sqrt(sig2))
  model_100 <- lm(Y_100 ~ X_100)
  b0_100[i] <- coef(model_100)[1]
  b1_100[i] <- coef(model_100)[2]
  sig2hat_100[i] <- summary(model_100)$sigma^2
}
```

Compare the mean of sample estimators with different sample size $= 25, 50, 100$

```
# The mean of esitmators b0, b1, and error variance s^2 for sample size = 25 with 100 simulations
mean(b0_25); mean(b1_25); mean(sig2hat_25)

## [1] 0.00168783
## [1] 1.904338
## [1] 35.96231

# The mean of esitmators b0, b1, and error variance s^2 for sample size = 50 with 100 simulations
mean(b0_50); mean(b1_50); mean(sig2hat_50)

## [1] -0.03020844
## [1] 1.719939
## [1] 35.68655

# The mean of esitmators b0, b1, and error variance s^2 for sample size = 100 with 100 simulations
mean(b0_100); mean(b1_100); mean(sig2hat_100)

## [1] -0.02098984
## [1] 1.686238
## [1] 35.86237

# The true population parameters beta_0, beta_1, and error variance sigma^2
beta0; beta1; sig2

## [1] 0.009603152
## [1] 1.716823
## [1] 35.60787
```

Compare the histograms of distributions of the sample estimators with different sample size $= 25, 50, 100$

```
# Construct hisotgrams of sample estiamtors to b0, b1, and error variance sig2 with increasing sample size
par(mar=c(1,1,1,1)); par(mfrow = c(4,3), mai = c(0.3, 0.1, 0.1, 0.1))  # Side by side comparsion
hist(b0_25); hist(b1_25); hist(sig2hat_25)
hist(b0_50); hist(b1_50); hist(sig2hat_50)
hist(b0_100); hist(b1_100); hist(sig2hat_100)
```



Compare the variance of the regression parameter estimators with different sample size $= 25, 50, 100$

We start with sample size $= 25$ as usual.

```
# Calculate the true variance of b0 and b1 for sample size 25
sumx_25 <- sum(X_25); xbar_25 <- sumx_25/25
sumx2_25 <- sum(X_25^2); SXX_25 <- sumx2_25 - 25*xbar_25^2
var_beta_0_25 <- sig2*(1/25 + xbar_25^2/SXX_25)
var_beta_1_25 <- sig2/SXX_25

var_beta_0_25; var_beta_1_25  # true variance of b0 and b1 for sample size 25

## [1] 1.641971
## [1] 11.5401

# Calculate sample variance of b0, b1 for each simulation for sample size 25
var_b0_25 <- vector()
var_b1_25 <- vector()
for(i in 1:n.sim){  # Assign the var estimators of b0, b1 for $n.sim$ samples individually
  var_b0_25[i] <- sig2hat_25[i]*(1/25 + xbar_25^2/SXX_25)
  var_b1_25[i] <- sig2hat_25[i]/SXX_25
}

mean(var_b0_25); mean(var_b1_25)  # the mean of sample variances of  b0 and b1

## [1] 1.658315
## [1] 11.65497
```

Compare the variance of the regression parameter estimators with different sample size $= 25, 50, 100$

Then with sample size $= 50$.

```
# Calculate the true variance of b0 and b1 for sample size 50
sumx_50 <- sum(X_50); xbar_50 <- sumx_50/50
sumx2_50 <- sum(X_50^2); SXX_50 <- sumx2_50 - 50*xbar_50^2
var_beta_0_50 <- sig2*(1/50 + xbar_50^2/SXX_50)
var_beta_1_50 <- sig2/SXX_50

var_beta_0_50; var_beta_1_50  # true variance of b0 and b1 for sample size 50

## [1] 0.7731356
## [1] 3.922478

# Calculate sample variance of b0, b1 for each simulation for sample size 50
var_b0_50 <- vector()
var_b1_50 <- vector()
for(i in 1:n.sim){  # Assign the var estimators of b0, b1 for $n.sim$ samples individually
  var_b0_50[i] <- sig2hat_50[i]*(1/50 + xbar_50^2/SXX_50)
  var_b1_50[i] <- sig2hat_50[i]/SXX_50
}

mean(var_b0_50); mean(var_b1_50)  # the mean of sample variances of b0 and b1

## [1] 0.7748439
## [1] 3.931146
```

Compare the variance of the regression parameter estimators with different sample size $= 25, 50, 100$

Finally with sample size $= 100$.

```
# Calculate the true variance of b0 and b1 for sample size 100
sumx_100 <- sum(X_100); xbar_100 <- sumx_100/100
sumx2_100 <- sum(X_100^2); SXX_100 <- sumx2_100 - 100*xbar_100^2
var_beta_0_100 <- sig2*(1/100 + xbar_100^2/SXX_100)
var_beta_1_100 <- sig2/SXX_100

var_beta_0_100; var_beta_1_100  # true variance of b0 and b1 for sample size 100

## [1] 0.360666
## [1] 1.678317

# Calculate sample variance of b0, b1 for each simulation for sample size 100
var_b0_100 <- vector()
var_b1_100 <- vector()
for(i in 1:n.sim){  # Assign the var estimators of b0, b1 for $n.sim$ samples individually
  var_b0_100[i] <- sig2hat_100[i]*(1/100 + xbar_100^2/SXX_100)
  var_b1_100[i] <- sig2hat_100[i]/SXX_100
}

mean(var_b0_100); mean(var_b1_100)  # the mean of sample variances of b0 and b1 for sample size 100

## [1] 0.3632439
## [1] 1.690313
```

Set up with a small error variance vs large error variance

```
## Q7 ##
set.seed(1004079631)
sig2_small <- 0.012
X_small <- rnorm(n = 100, mean = 0, sd = sqrt(sigX))  #Simulate the predictor variable with sample size = 100

b0_small <- vector()  # saves the sample estimates of beta_0 with small error var
b1_small <- vector()  # saves the sample estimates of beta_1 with small error var
sig2hat_small <- vector()  # saves the sample estimates of new error variance sig2_small


for(i in 1:n.sim){  # Assign the estimators for $n.sim$ samples individually with new error variance sig2_small
  Y_small <- beta0 + beta1*X_small + rnorm(n = 100, mean = 0, sd = sqrt(sig2_small))
  model_small <- lm(Y_small ~ X_small)
  b0_small[i] <- coef(model_small)[1]
  b1_small[i] <- coef(model_small)[2]
  sig2hat_small[i] <- summary(model_small)$sigma^2
}

# Increase error var to large
set.seed(1004079631)
sig2_large <- 1100.289
X_large <- rnorm(n = 100, mean = 0, sd = sqrt(sigX))  #Simulate the predictor variable with sample size = 100

b0_large <- vector()  # saves the sample estimates of beta_0 with large error var
b1_large <- vector()  # saves the sample estimates of beta_1 with large error var
sig2hat_large <- vector()  # saves the sample estimates of new error variance sig2_large

for(i in 1:n.sim){  # Assign the estimators with new error variance sig2_large
  Y_large <- beta0 + beta1*X_large + rnorm(n = 100, mean = 0, sd = sqrt(sig2_large))
  model_large <- lm(Y_large ~ X_large)
  b0_large[i] <- coef(model_large)[1]
```

Compare the mean of the sample esitamtes with different error variance

```
# The mean of esitmators b0, b1 for small error var
mean(b0_small); mean(b1_small)
```

```
## [1] 0.009041536
## [1] 1.716261
```

```
# The mean of esitmators b0, b1 for large error var
mean(b0_large); mean(b1_large)
```

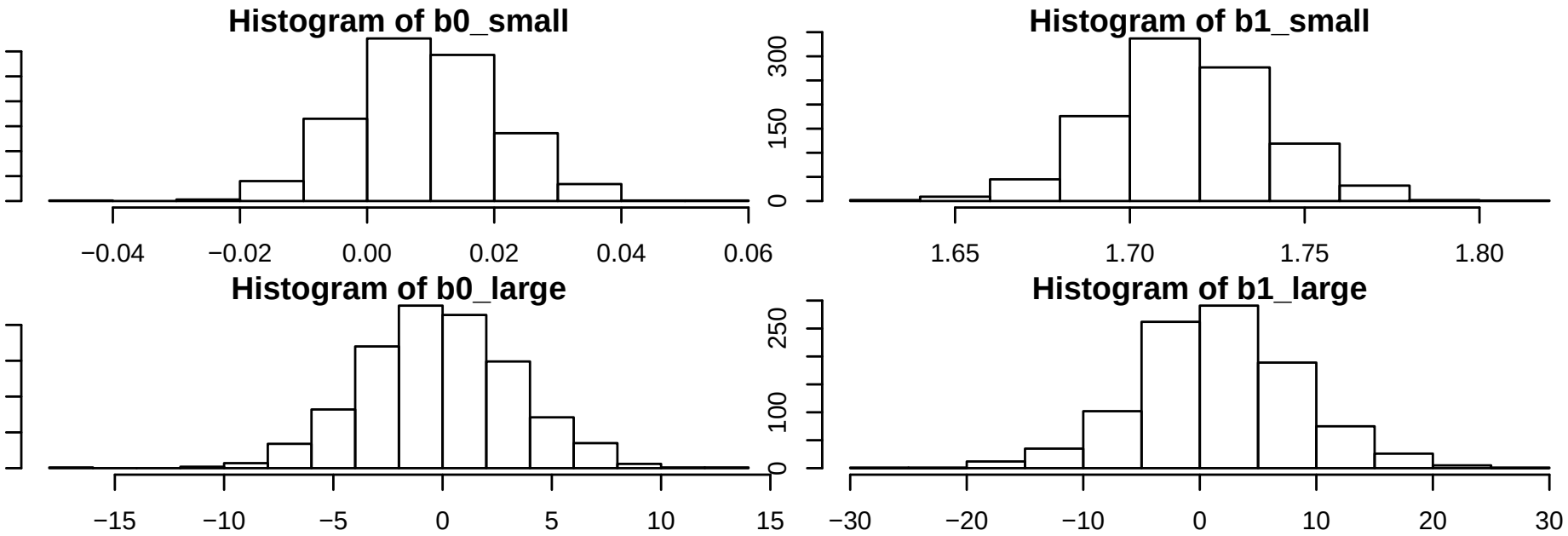
```
## [1] -0.1604569
## [1] 1.546808
```

```
# The true population parameters beta_0, beta_1
beta0; beta1
```

```
## [1] 0.009603152
## [1] 1.716823
```

Compare the histograms of distributions of sample estimates with different error variance

```r
# Construct hisotgrams of estiamtors for b0 & b1, small error var vs large error var
par(mar=c(1,1,1,1)); par(mfrow = c(3,2), mai = c(0.3, 0.1, 0.1, 0.1))  # set up the marigins
hist(b0_small); hist(b1_small)  # Side by Side comparsion
hist(b0_large); hist(b1_large)
```



Compare the variance of regression parameter estimators with different error variance

```
# Calculate the true variance of b0 and b1 for small error var
sumx_small <- sum(X_small); xbar_small <- sumx_small/100
sumx2_small <- sum(X_small^2); SXX_small <- sumx2_small - 100*xbar_small^2
var_beta_0_small <- sig2*(1/100 + xbar_small^2/SXX_small)
var_beta_1_small <- sig2/SXX_small

var_beta_0_small; var_beta_1_small  # true variance of b0 and b1 for small error var

## [1] 0.360666
## [1] 1.678317

# Calculate sample variance of b0, b1 for for small error var
var_b0_small <- vector()
var_b1_small <- vector()
for(i in 1:n.sim){  # Assign the var estimators of b0, b1 for $n.sim$ samples individually
  var_b0_small[i] <- sig2hat_small[i]*(1/100 + xbar_small^2/SXX_small)
  var_b1_small[i] <- sig2hat_small[i]/SXX_small
}

mean(var_b0_small); mean(var_b1_small)  # the mean of sample variances of b0 and b1 for small error var

## [1] 0.0001224147
## [1] 0.0005696424
```

Compare the variance of regression parameter estimators with different error variance

```
# Calculate the true variance of b0 and b1 for large error var
sumx_large <- sum(X_large); xbar_large <- sumx_large/100
sumx2_large <- sum(X_large^2); SXX_large <- sumx2_large - 100*xbar_large^2
var_beta_0_large <- sig2*(1/100 + xbar_large^2/SXX_large)
var_beta_1_large <- sig2/SXX_large

var_beta_0_large; var_beta_1_large  # true variance of b0 and b1 for large error var

## [1] 0.360666
## [1] 1.678317

# Calculate sample variance of b0, b1 for for large error var
var_b0_large <- vector()
var_b1_large <- vector()
for(i in 1:n.sim){  # Assign the var estimators of b0, b1 for $n.sim$ samples individually
  var_b0_large[i] <- sig2hat_large[i]*(1/100 + xbar_large^2/SXX_large)
  var_b1_large[i] <- sig2hat_large[i]/SXX_large
}

mean(var_b0_large); mean(var_b1_large)  # the mean of sample variances of b0 and b1 for large error var

## [1] 11.2243
## [1] 52.23094
```

Key Takeaways

From this project, we learn that...

1. If we conduct a $(1-\alpha)\%$ confidence interval on $\hat{\beta}_0, \hat{\beta}_1$ for a large number of times, then for $(1-\alpha)\%$ of them, the true population parameter β_0, β_1 will be captured by the CI.
2. As the sample size increases, the mean of parameter estimators are getting closer to the true population parameters; the variance of estimators also decreases.
3. As the true error variance increases, the variance of estimators also increases. The mean of parameter estimators are less close to the population parameters under the influence of larger error variance (error variance dominates).

Thank You!

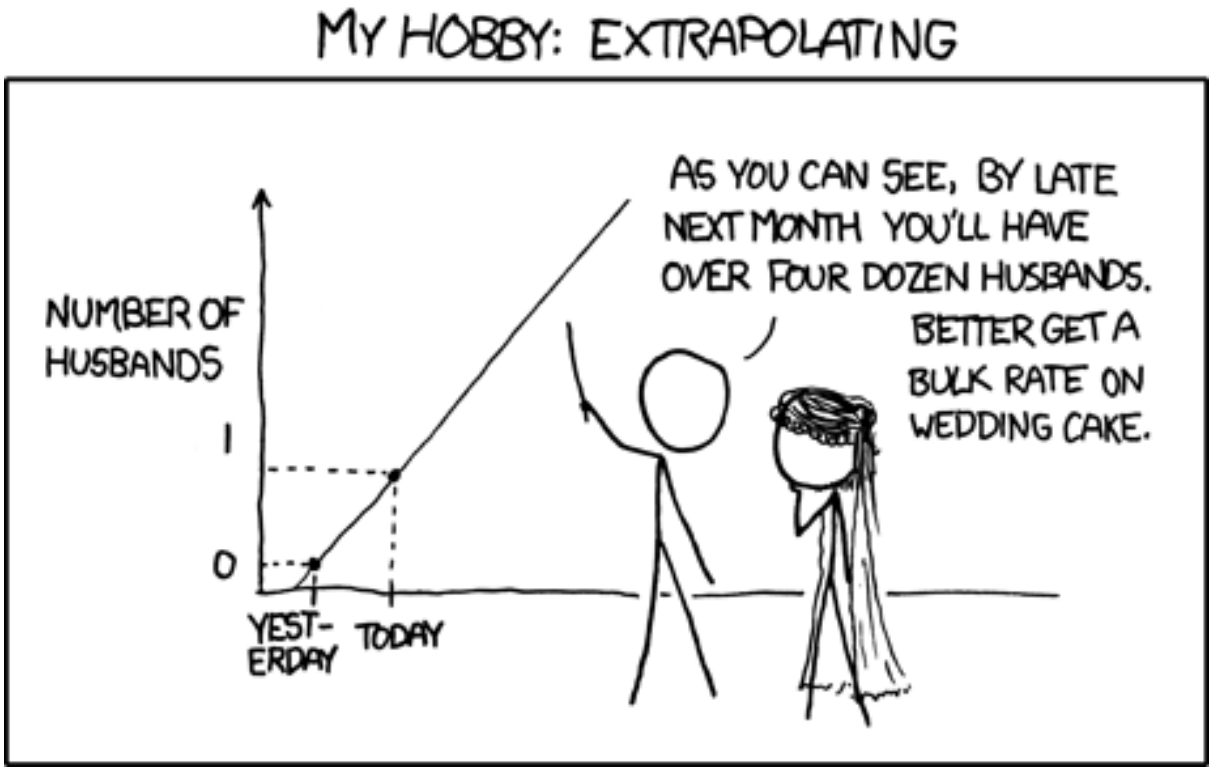


Figure: Source: xkcd