STAT 8010 R Lab 17: Simple Linear Regression III

Whitney

November 23, 2020

Contents

Understanding Sampling Distributions and Confident Intervals via simulation

Simulate the "data" $\{x_i, y_i\}_{i=1}^n$ where $y_i = \beta_0 + \beta_1 x_i + \varepsilon_i$, $\varepsilon \sim \mathrm{N}(0, \sigma^2)$. Repeat this process N times.

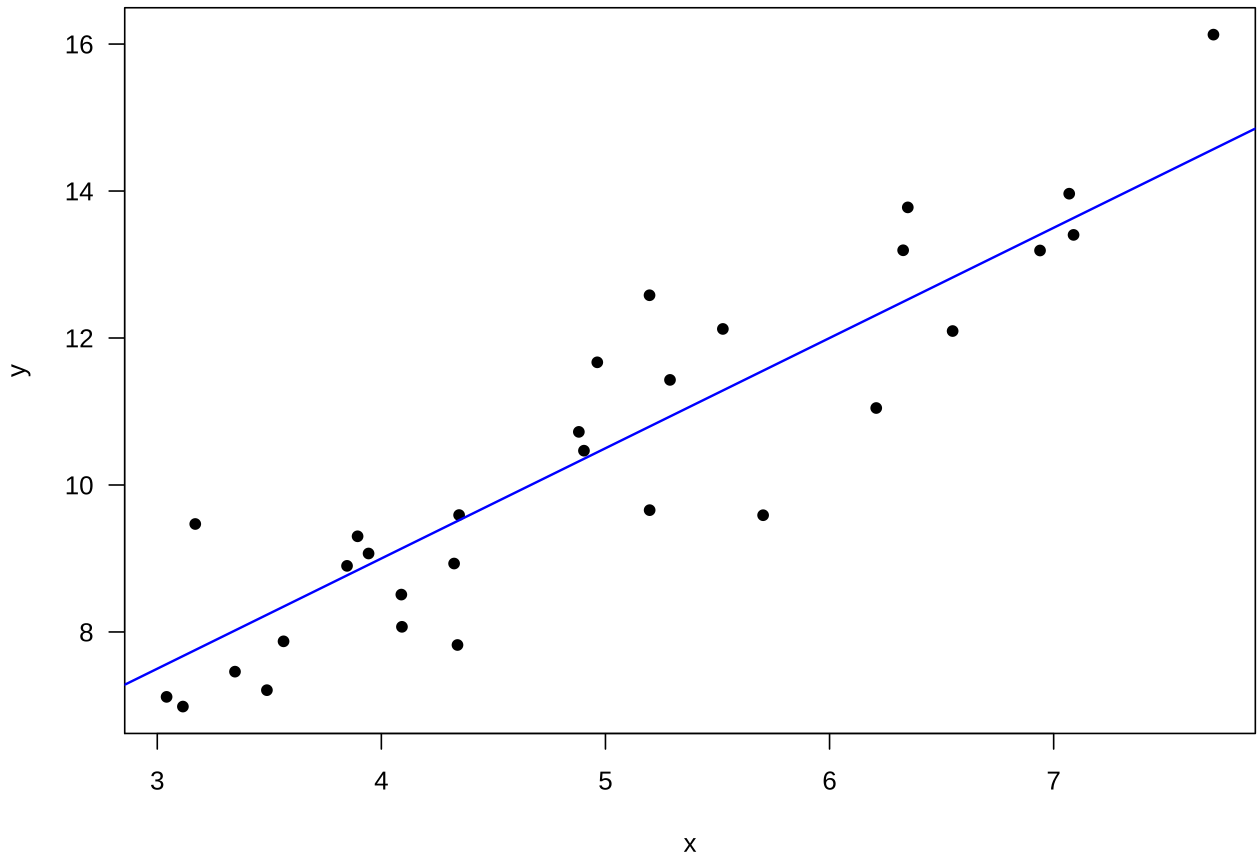
Generate data in R

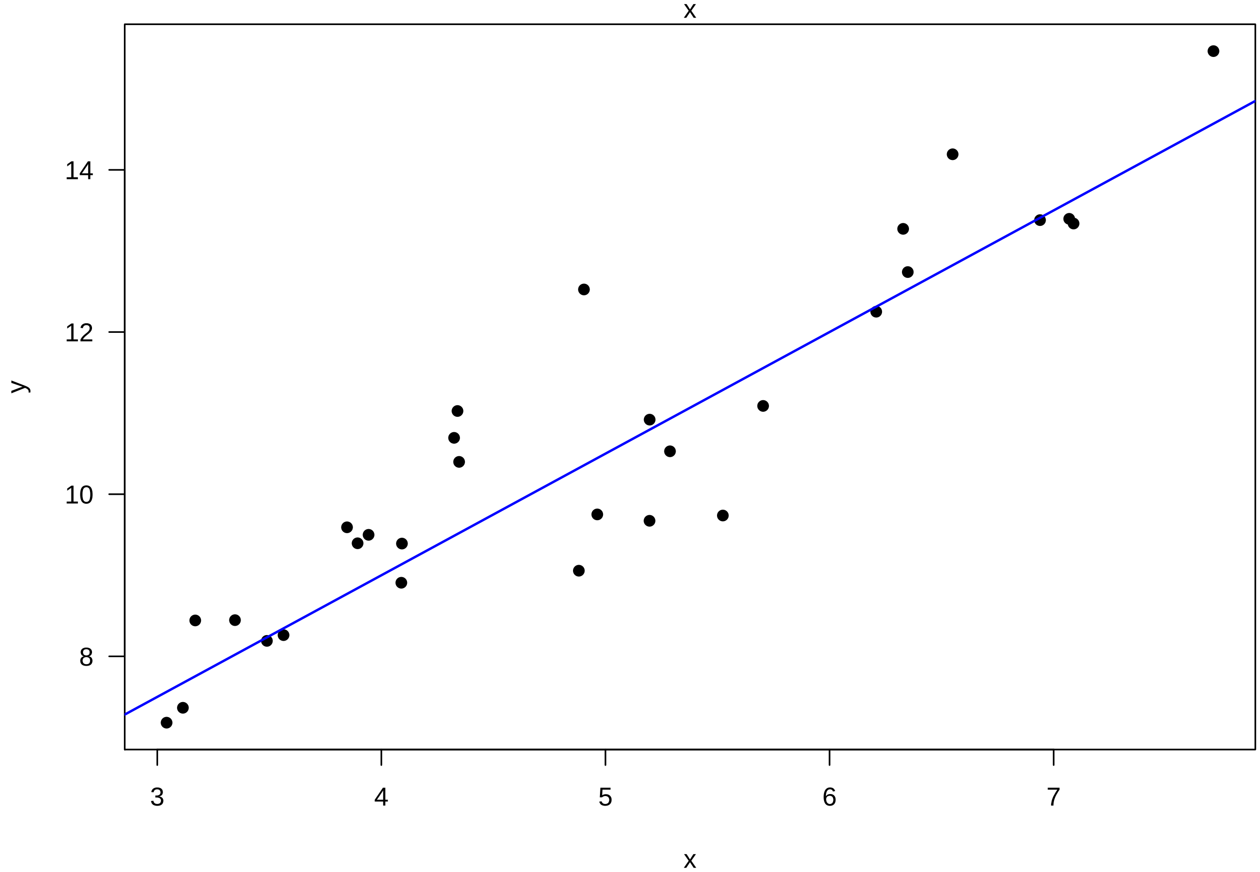
```
set.seed(12)
n = 30; beta0 = 3; beta1 = 1.5; N = 100; sigma2 = 1
x <- 3 + 5 * runif(n)
set.seed(123)
y <- replicate(N, beta0 + beta1 * x + rnorm(n, mean = 0, sd = sqrt(sigma2)))
dim(y)
```

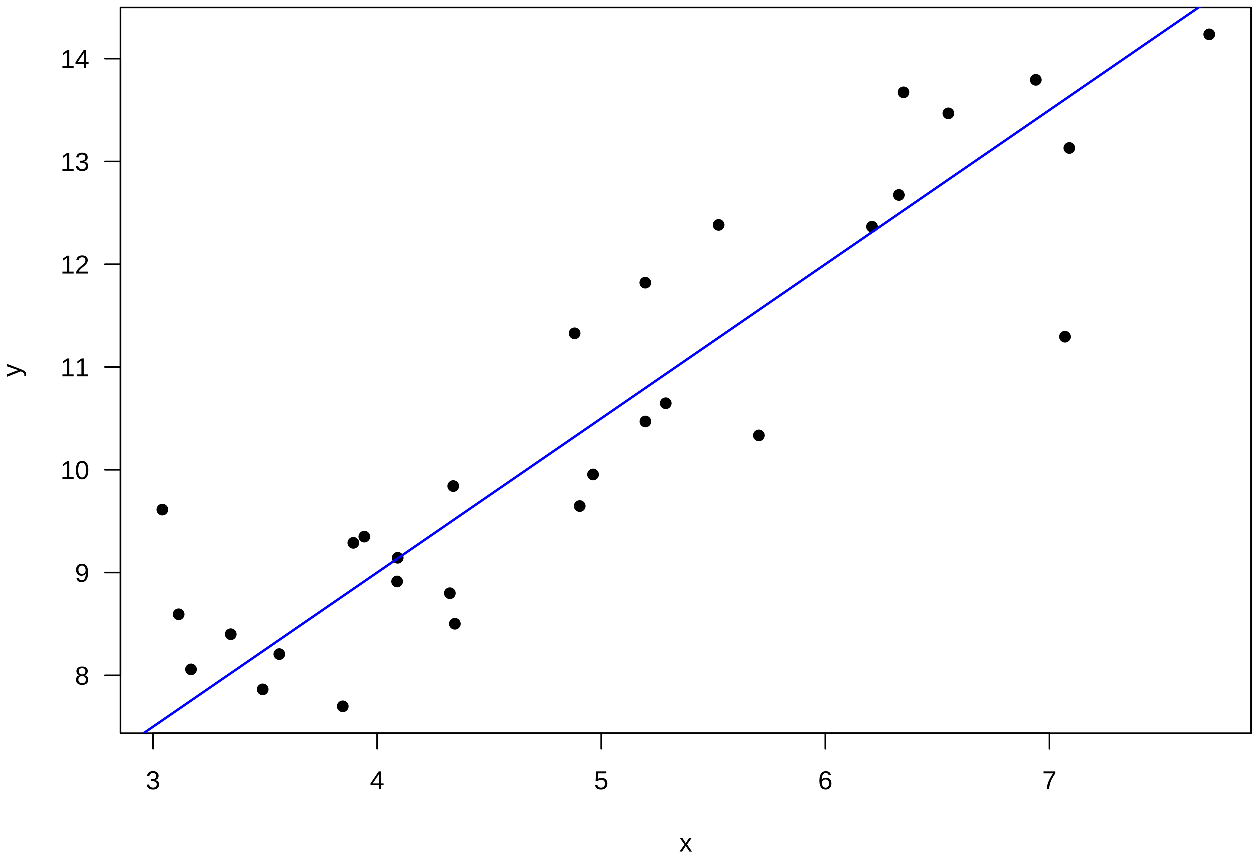
```
## [1]  30 100
```

Plot the first few simulated datasets

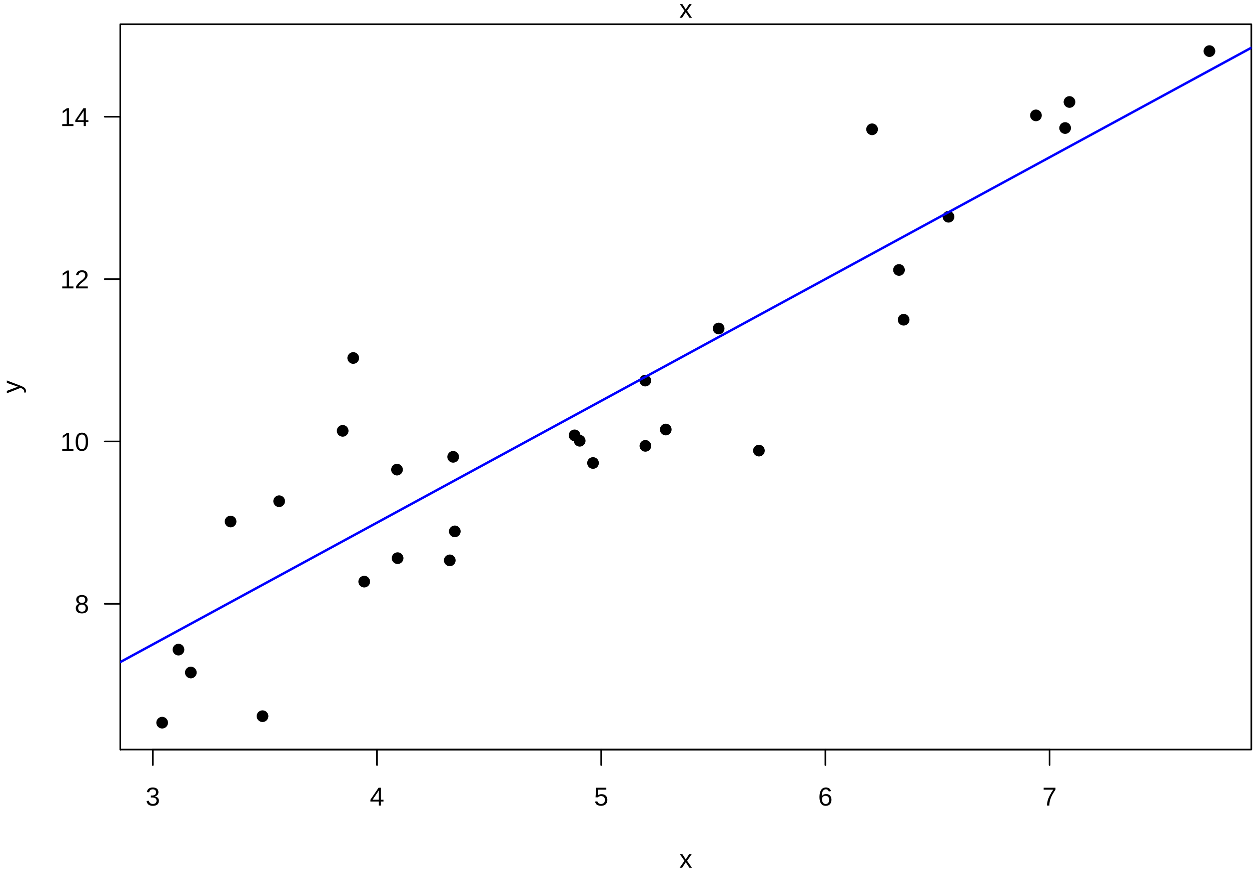
```
for (i in 1:5){
  plot(x, y[, i], pch = 16, las = 1, ylab = "y")
  abline(3, 1.5, col = "blue", lwd = 1.5)
}
```


16
14
12
10
8
y
3
4
5
6
7
x

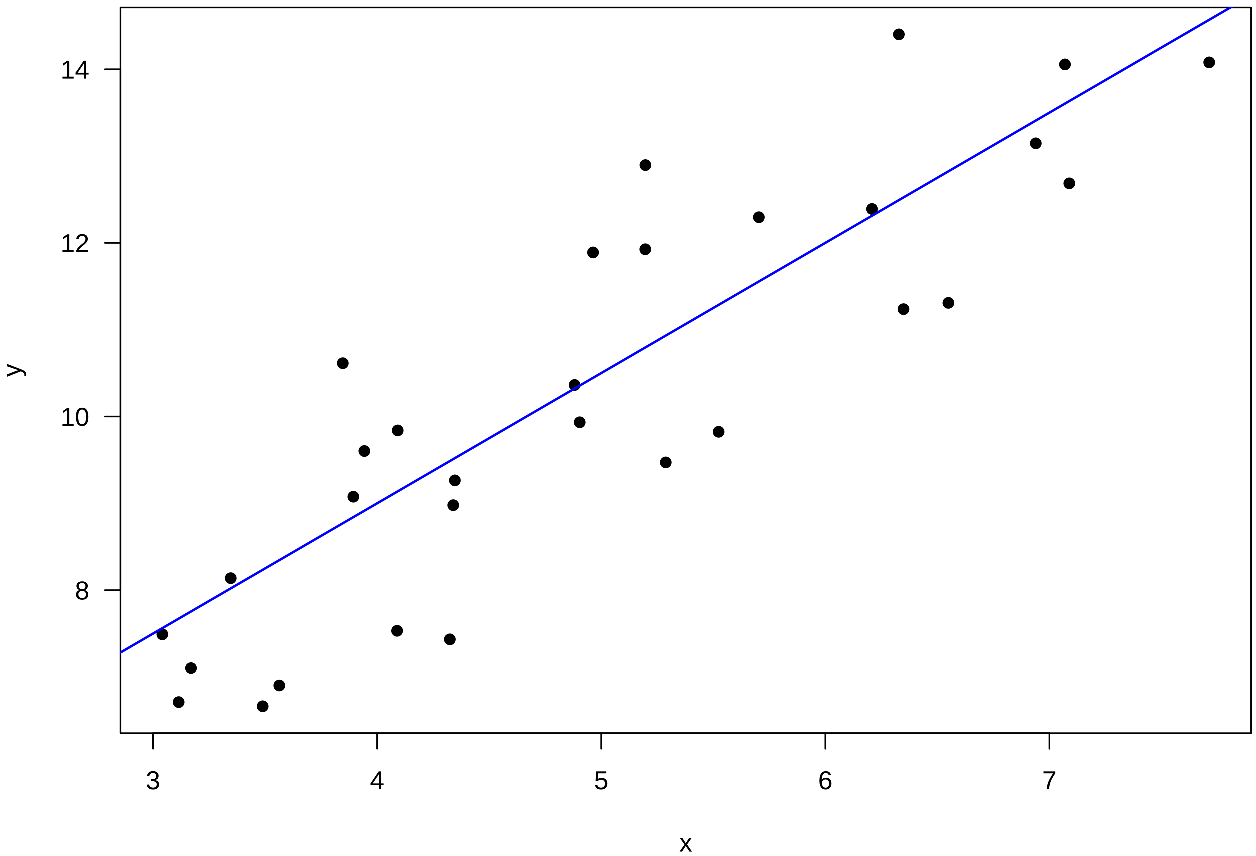

14
12
10
8
y
3
4
5
6
7
x



14
13
12
11
10
9
8
y
3
4
5
6
7
x



14
12
10
8
y
3
4
5
6
7
x

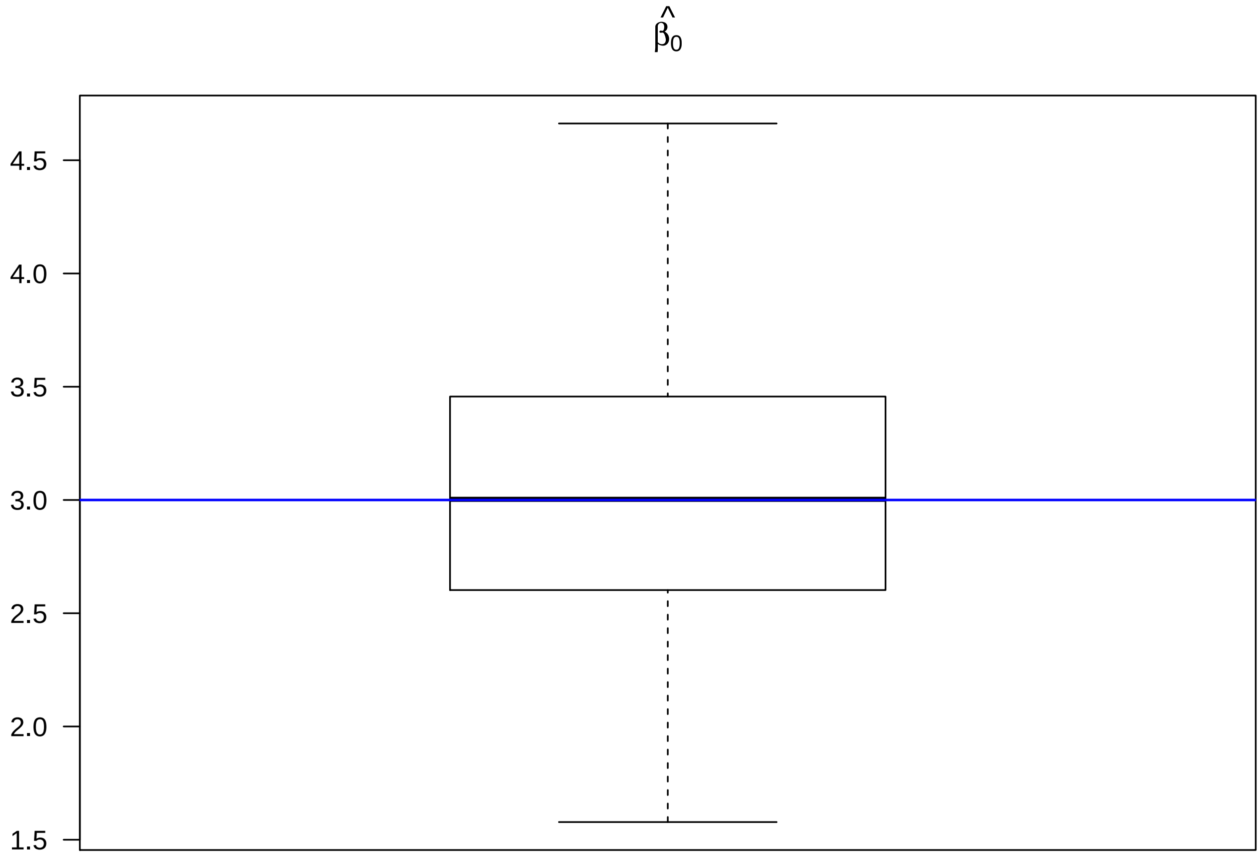


Estimate the β_0, β_1, and σ^2 for each simulated dataset

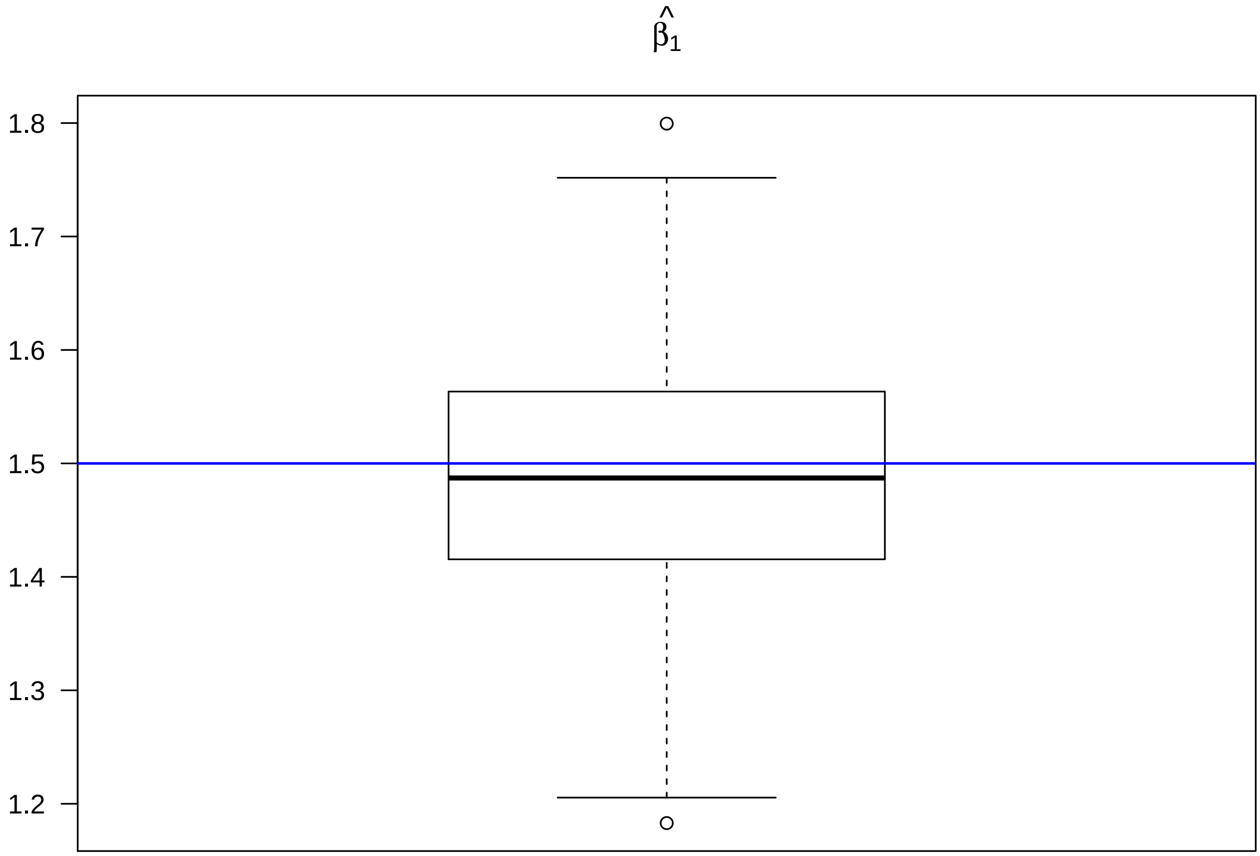
```
beta0_hat <- beta1_hat <- sigma2_hat <- se_beta1 <- numeric(N)
for (i in 1:100){
  fit <- lm(lm(y[, i] ~ x))
  beta0_hat[i] <- summary(fit)[["coefficients"]][, 1][1]
  beta1_hat[i] <- summary(fit)[["coefficients"]][, 1][2]
  se_beta1[i] <- summary(fit)[["coefficients"]][, 2][2]
  sigma2_hat[i] <- summary(fit)[["sigma"]]^2
}
```

Assess the estimation perfromance

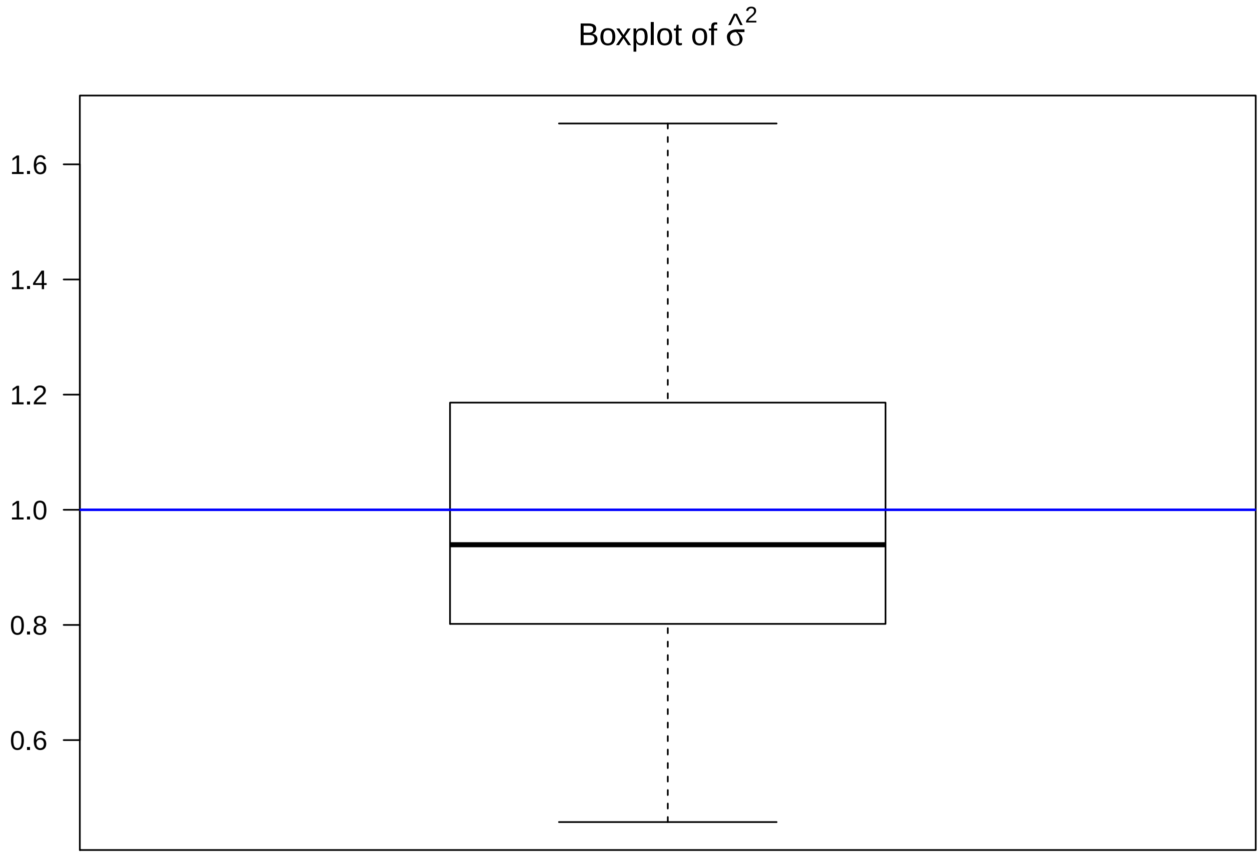
```
boxplot(beta0_hat, las = 1, main = expression(hat(beta[0])))
abline(h = beta0, col = "blue", lwd = 1.5)
```




```
boxplot(beta1_hat, las = 1, main = expression(hat(beta[1])))
abline(h = beta1, col = "blue", lwd = 1.5)
```

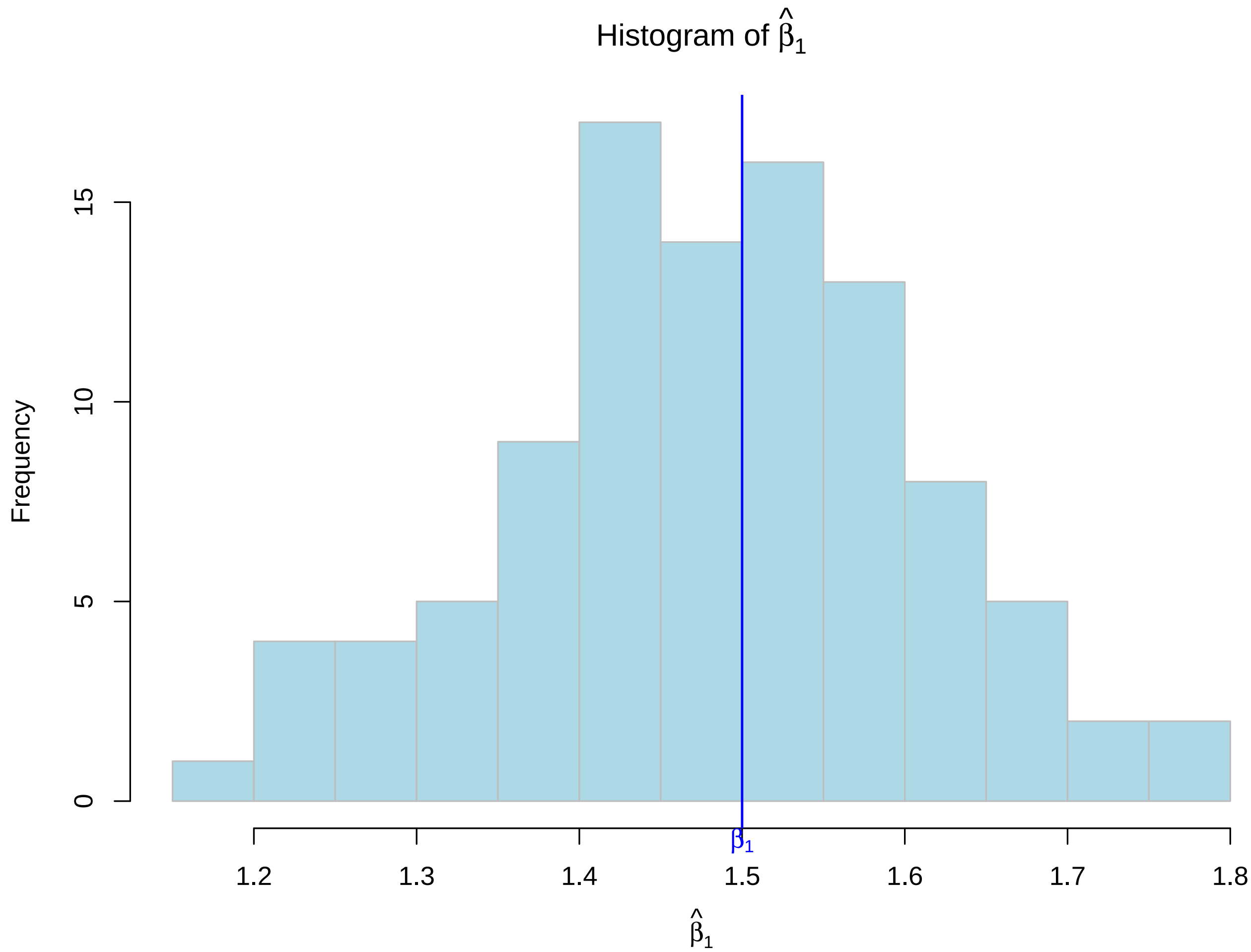



```
boxplot(sigma2_hat, las = 1, main = expression(paste("Boxplot of ", hat(sigma)^2)))
abline(h = sigma2, col = "blue", lwd = 1.5)
```



Sampling distribution

```
hist(beta1_hat, 16, col = "lightblue", border = "gray",
     main = expression(paste("Histogram of ", hat(beta)[1])),
     xlab = expression(hat(beta)[1]))
abline(v = beta1, col = "blue", lwd = 1.5)
mtext(expression(beta[1]), 1, at = beta1, col = "blue")
```

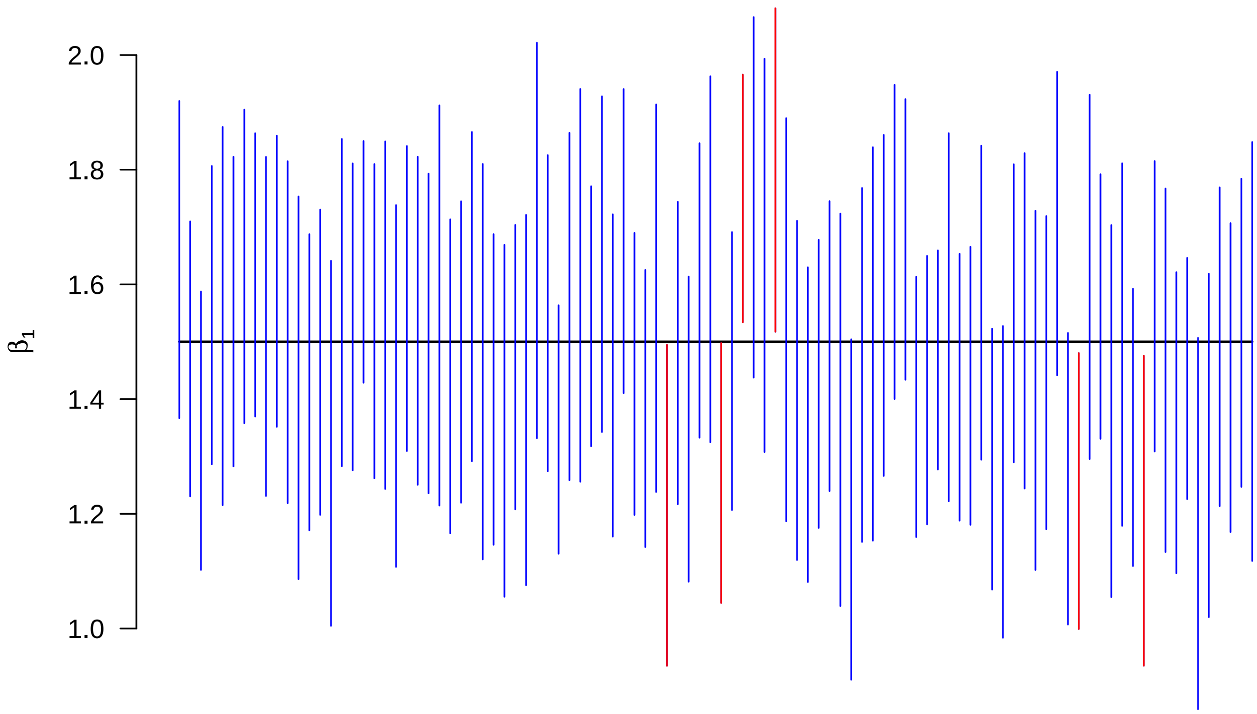


CI's for all the simulated datasets

```
t <- qt(1 - 0.05 / 2, n - 2)
LL <- beta1_hat - t * se_beta1
UL <- beta1_hat + t * se_beta1
miss <- which((LL - beta1) * (UL - beta1) > 0)


par(las = 1)
plot(1:100, rep(beta1, N), type = "l", bty = "n", xaxt = "n", xlab = "",
     lwd = 1.5, ylab = expression(hat(beta)[1]))
for (i in 1:100){
  segments(i, LL[i], i, UL[i], col = "blue")
}

for (i in miss){
  segments(i, LL[i], i, UL[i], col = "red")
}
```



ANOVA

First Step: Load the data

```r
dat <- read.csv('http://whitneyhuang83.github.io/STAT8010/Data/maxHeartRate.csv', header = T)
head(dat)
```

```
##   Age MaxHeartRate
## 1  18          202
## 2  23          186
## 3  25          187
## 4  35          180
## 5  65          156
## 6  54          169
```

```r
attach(dat)
```

Fitting a simple linear regression

```r
fit <- lm(MaxHeartRate ~ Age)
summary(fit)
```

```
##
## Call:
## lm(formula = MaxHeartRate ~ Age)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.9258 -2.5383  0.3879  3.1867  6.6242
##
## Coefficients:
```

```
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 210.04846    2.86694   73.27  < 2e-16 ***
## Age          -0.79773    0.06996  -11.40 3.85e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.578 on 13 degrees of freedom
## Multiple R-squared:  0.9091, Adjusted R-squared:  0.9021
## F-statistic:   130 on 1 and 13 DF,  p-value: 3.848e-08
```

```
R.sq <- summary(fit)[["r.squared"]]
r <- cor(dat$Age, dat$MaxHeartRate)
r^2; R.sq
```

```
## [1] 0.9090967
```

```
## [1] 0.9090967
```

ANOVA

```
anova(fit)
```

```
## Analysis of Variance Table
##
## Response: MaxHeartRate
##           Df  Sum Sq Mean Sq F value    Pr(>F)
## Age        1 2724.50 2724.50  130.01 3.848e-08 ***
## Residuals 13  272.43   20.96
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```