Assignment2_2

Linfeng Zhou

September 25, 2015

Question 4-7

4. This is a very challenging set of questions, but they address several key topics in data analytics. You may work with other students on a solution with the recognition that you may not complete this set of questions. I have frequently used the phrase "data generating process" (or "DGP") to describe the hypothetical process by which observations of data arise in the real world. We discussed at some length the bivariate linear regression model,

$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i$$

In this problem, we will work with a specific DGP and evaluate features of $\widehat{\beta}_1$, the least squares estimate of β_1. Suppose your DGP is

$$y_i = 1 + 2x_i + \epsilon_i$$

where $x \sim N(0, 1)$ and $\epsilon \sim N(0, 1)$. Using R or Python, write code to generate 1,000 draws for x and ϵ. Use these draws to generate y in accordance with the given DGP. Using R or Python, write code to estimate the bivariate model,

$$y_i = \beta_0 + \beta_1 x_i$$

and to summarize the findings.

5. Repeat 4 above five different times for a new set of random draws for each replication. (This effort is called Monte Carlo simulation. Each time you generate a new set of data and estimate a model, you have a replication. For example, here you have five replications.)

6. Write code to automatically repeat 5 above 1,000 times (or 1,000 replications), each time automatically recording the estimated value of β_1. Generate a histogram of these 1,000 replications of your estimates of β_1. What does the dispersion of these replications measure?

7. Suppose that you were not interested in the estimate of β_1, but instead in some functional transformation, such as the estimate of $\exp(\beta_1)$. What might you do with your 1,000 replications from 6 above to inform you about the distribution of this transformation of β_1?

```
library(ggplot2)
x4 <- rnorm(1000, mean=0, sd=1)
e4 <- rnorm(1000, mean=0, sd=1)
y4 <- 1 + 2 * x4 + e4
fit4 <- lm(y4~x4)
summary(fit4)
```

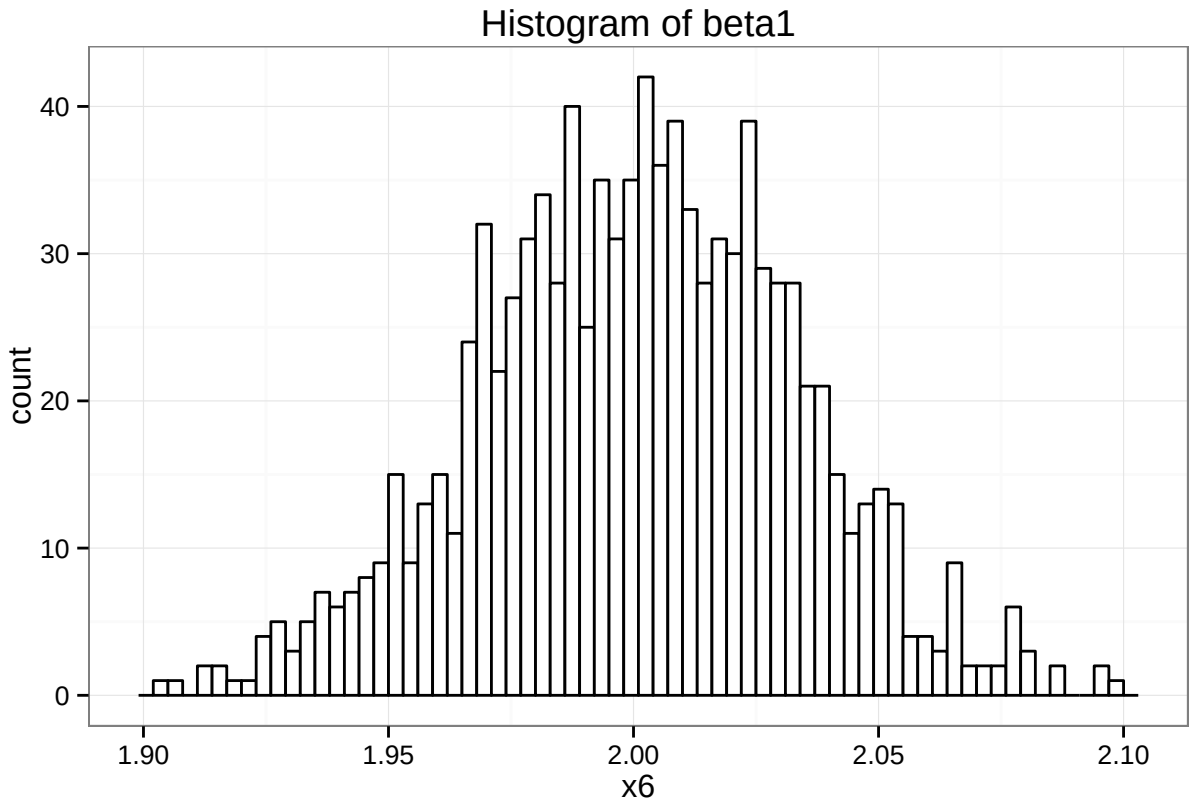
Question 4

```
##
## Call:
## lm(formula = y4 ~ x4)
##
```

```
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.3216 -0.6423 -0.0198  0.6325  3.6149
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.01166    0.03104   32.59   <2e-16 ***
## x4           1.96389    0.03206   61.25   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9811 on 998 degrees of freedom
## Multiple R-squared:  0.7899, Adjusted R-squared:  0.7897
## F-statistic:  3751 on 1 and 998 DF,  p-value: < 2.2e-16
```

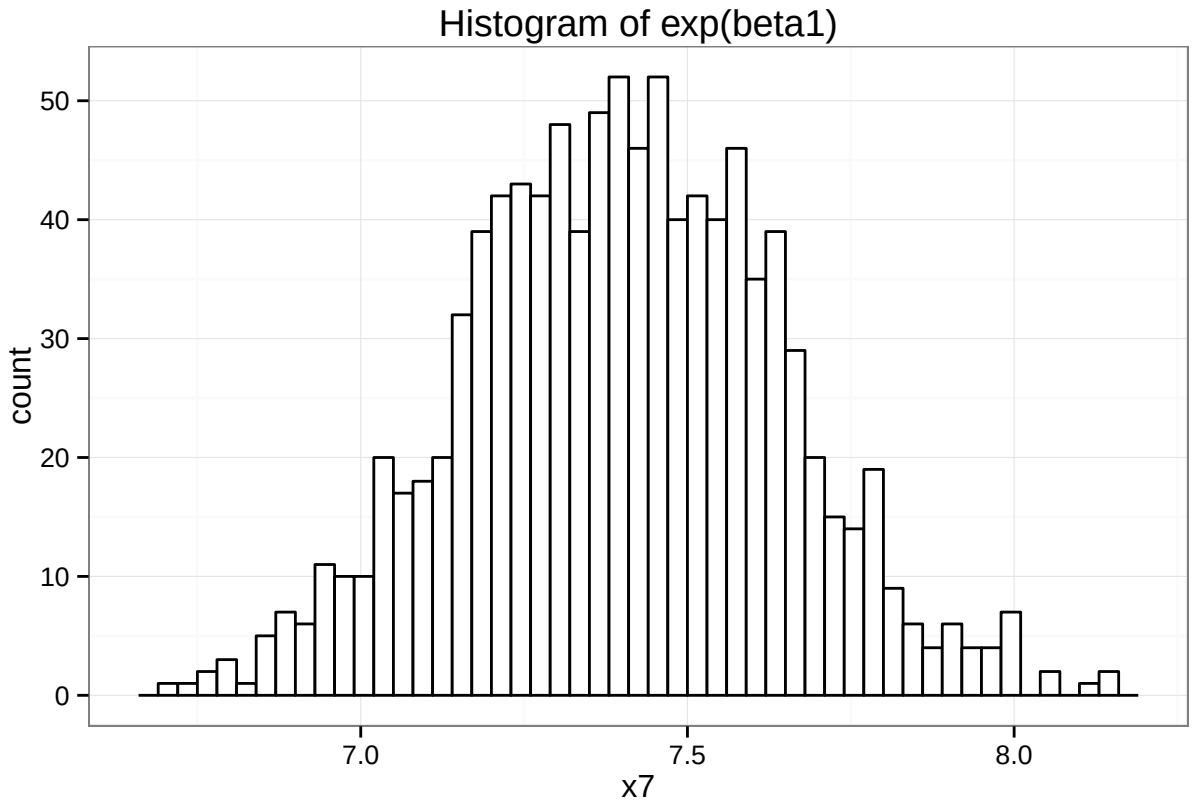
```
##Question 5
## generate norm
q5 <- function(reptime){
   d5 <- c()
   f5 <- c()
   for (i in 1:reptime){
      e5 <- rnorm(1000, mean=0, sd=1)
      x5 <- rnorm(1000, mean=0, sd=1)
      y5 <- 1 + 2 * x5 + e5
      fit5 <- lm(y5~x5)
      f5 <- rbind(f5,fit5[1]$coefficients)
     }
   return(f5)
}
a5<-q5(5)

# Question 6
a6<-q5(1000)
x6<-data.frame(a6[,2])
colnames(x6)<-"x6"
ggplot(x6,aes(x=x6)) +
  geom_histogram(binwidth=0.003,fill="white",colour="black")+
  ggtitle("Histogram of beta1")+
  theme_bw()
```




```
# Question 7
x7<-exp(x6)
colnames(x7)<-"x7"
ggplot(x7,aes(x=x7)) +
  geom_histogram(binwidth=0.03,fill="white",colour="black")+
  ggtitle("Histogram of exp(beta1)")+
  theme_bw()
```

```
## Warning in loop_apply(n, do.ply): position_stack requires constant width:
## output may be incorrect
```


Histogram of exp(beta1)
count
50
40
30
20
10
0
7.0
7.5
8.0
x7