

Problem Set 2: Linear Regression

Sagnik Roy

05-02-2026

Problem 1

To demonstrate that the population regression line is fixed, but the least squares regression line varies

Suppose the population regression line is given by $Y = 2 + 3x$, while the observed data are generated from the model $y = 2 + 3x + \varepsilon$.

Step 1: For x in the range (5,10), graph the population regression line.

Step 2: Generate x_i ($i = 1, 2, \dots, n$) from the uniform distribution $x_i \sim \text{Uniform}(5, 10)$, and generate $\varepsilon_i \sim N(0, 4^2)$. Hence compute $y_i = 2 + 3x_i + \varepsilon_i$, $i = 1, 2, \dots, n$.

Step 3: Based on the generated data (x_i, y_i) , obtain the least squares regression line.

Step 4: Repeat Steps 2 and 3 five times. Plot the five least squares regression lines along with the population regression line obtained in Step 1.

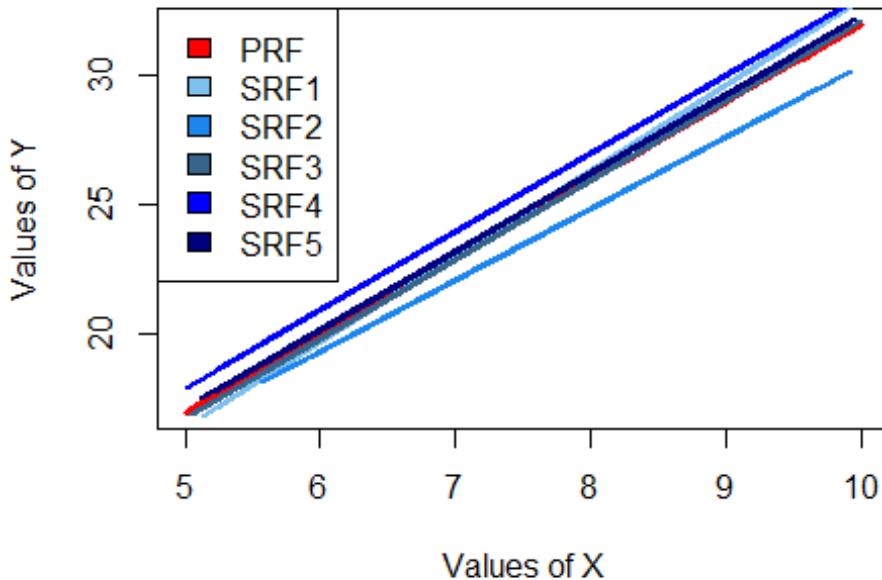
Interpretation: Comment on the variability of the least squares regression lines around the fixed population regression line.

Take $n = 50$ and set the seed as 123.

Solution

```
rm(list=ls())
set.seed(123)
b0=c();b1=c()
color=c("skyblue2", "dodgerblue2", "steelblue4", "blue1", "navy")
x=seq(5,10,length.out=50);y=2+3*x
plot(x,y,type="l",col="red",lwd=3,main="The regression
functions",ylab="Values of Y",xlab="Values of X")
for(i in 1:5){
  xi=runif(50,5,10);ei=rnorm(50,0,4)
  yi=2+3*xi+ei
  model=lm(yi~xi)
  summary(model)
  b0=c(b0,as.data.frame(model$coefficients)[1,1])
  b1=c(b1,as.data.frame(model$coefficients)[2,1])
  lines(xi,predict(model),type="l",col=color[i],lwd=2)}
legend("topleft",legend=c("PRF","SRF1","SRF2","SRF3","SRF4","SRF5"),fill=c("red",color))
```

The regression functions



```
coefs=data.frame("Function"=c("SRF1","SRF2","SRF3","SRF4","SRF5"), "B0"=b0, "B1"  
"=b1);coefs  
  
##   Function      B0      B1  
## 1    SRF1 -0.09638929 3.305396  
## 2    SRF2  2.79218839 2.761042  
## 3    SRF3  1.39299737 3.073267  
## 4    SRF4  2.82308856 3.023608  
## 5    SRF5  2.03250638 3.028097
```

Interpretation

Although the population regression line is fixed, the least squares regression line varies from sample to sample due to random error.

Problem 2

To demonstrate that $\hat{\beta}_0$ and $\hat{\beta}$ minimise RSS

Step 1: Generate $x_i \sim \text{Uniform}(5,10)$, and mean-centre the values of x_i . Generate $\varepsilon_i \sim N(0,1)$. Compute $y_i = 2 + 3x_i + \varepsilon_i$, $i = 1, 2, \dots, n$.

Take $n = 50$ and set the seed as 123.

Step 2: Assume a linear regression model of the form $y_i = \beta_0 + \beta x_i + \varepsilon_i$. Using only the observed data (x_i, y_i) , obtain the least squares estimates of β_0 and β .

Step 3: Consider a large grid of values for (β_0, β) that includes the least squares estimates obtained above. For each combination, compute the residual sum of squares $RSS = \sum_{i=1}^n (y_i - \beta_0 - \beta x_i)^2$. Identify the values of (β_0, β) for which the RSS is minimised.

Solution

```
rm(list=ls())
set.seed(123)
xi=runif(50,5,10)
xi=xi-mean(xi)
ei=rnorm(50,0,1)
yi=2+3*xi+ei
model=lm(yi~xi)
summary(model)

##
## Call:
## lm(formula = yi ~ xi)
##
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -2.25578 -0.55786 -0.06567  0.54926  2.18613 
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  2.0562    0.1330   15.46   <2e-16 ***
## xi          3.0764    0.0913   33.70   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9404 on 48 degrees of freedom
## Multiple R-squared:  0.9594, Adjusted R-squared:  0.9586 
## F-statistic: 1135 on 1 and 48 DF,  p-value: < 2.2e-16

b0=(as.data.frame(model$coefficients)[1,1])
b1=(as.data.frame(model$coefficients)[2,1])
rss=sum((yi-(b0+b1*xi))^2)
b0

## [1] 2.056189

b1

## [1] 3.076349

rss

## [1] 42.4455
```

```

r=c()
a=seq(b0-1,b0+1,0.05)
b=seq(b1-1,b1+1,0.05)
for(i in 1:length(a)){
  r=c(r,sum((yi-(a[i]+b[i]*xi))^2)))
d=data.frame("B0"=a,"B1"=b,"RSS"=r);d

##          B0        B1      RSS
## 1  1.056189 2.076349 198.53732
## 2  1.106189 2.126349 183.31837
## 3  1.156189 2.176349 168.87987
## 4  1.206189 2.226349 155.22184
## 5  1.256189 2.276349 142.34426
## 6  1.306189 2.326349 130.24715
## 7  1.356189 2.376349 118.93049
## 8  1.406189 2.426349 108.39429
## 9  1.456189 2.476349  98.63855
## 10 1.506189 2.526349  89.66327
## 11 1.556189 2.576349  81.46845
## 12 1.606189 2.626349  74.05409
## 13 1.656189 2.676349  67.42019
## 14 1.706189 2.726349  61.56674
## 15 1.756189 2.776349  56.49376
## 16 1.806189 2.826349  52.20124
## 17 1.856189 2.876349  48.68917
## 18 1.906189 2.926349  45.95756
## 19 1.956189 2.976349  44.00641
## 20 2.006189 3.026349  42.83573
## 21 2.056189 3.076349  42.44550
## 22 2.106189 3.126349  42.83573
## 23 2.156189 3.176349  44.00641
## 24 2.206189 3.226349  45.95756
## 25 2.256189 3.276349  48.68917
## 26 2.306189 3.326349  52.20124
## 27 2.356189 3.376349  56.49376
## 28 2.406189 3.426349  61.56674
## 29 2.456189 3.476349  67.42019
## 30 2.506189 3.526349  74.05409
## 31 2.556189 3.576349  81.46845
## 32 2.606189 3.626349  89.66327
## 33 2.656189 3.676349  98.63855
## 34 2.706189 3.726349 108.39429
## 35 2.756189 3.776349 118.93049
## 36 2.806189 3.826349 130.24715
## 37 2.856189 3.876349 142.34426
## 38 2.906189 3.926349 155.22184
## 39 2.956189 3.976349 168.87987
## 40 3.006189 4.026349 183.31837
## 41 3.056189 4.076349 198.53732

```

```
d[which(r==min(r)),]
##          B0        B1      RSS
## 21  2.056189 3.076349 42.4455
```

We observe that the RSS is minimised at the least squares estimates of β_0 and β .

Problem 3

To demonstrate that least squares estimators are unbiased

Step 1: Generate $x_i \sim \text{Uniform}(0,1)$, $\varepsilon_i \sim N(0,1)$, and compute $y_i = \beta_0 + \beta x_i + \varepsilon_i$, where $\beta_0 = 2$ and $\beta = 3$.

Step 2: Based on the generated data (x_i, y_i) , obtain the least squares estimates $\hat{\beta}_0$ and $\hat{\beta}$.

Repeat Steps 1 and 2 for $R = 1000$ simulations.

The final estimates are given by the averages of the simulated values of $\hat{\beta}_0$ and $\hat{\beta}$.

Compare these averages with the true values β_0 and β , and comment.

Take $n = 50$ and set the seed as 123.

Solution

```
rm(list=ls())
set.seed(123)
BE0=c();BE1=c()
for(i in 1:1000){
  xi=runif(50,0,1)
  xi=xi-mean(xi)
  ei=rnorm(50,0,1)
  yi=2+3*xi+ei
  model=lm(yi~xi)
  BE0=c(BE0,(as.data.frame(model$coefficients)[1,1]))
  BE1=c(BE1,(as.data.frame(model$coefficients)[2,1])))
mean(BE0) #estimate of beta0

## [1] 2.003898

mean(BE1) #estimate of beta1
## [1] 2.996422
```

Comment

The average of the estimated coefficients is close to the true parameter values $\beta_0 = 2$ and $\beta = 3$, demonstrating unbiasedness.

Problem 4

Comparing several simple linear regression models

Attach the **Boston** dataset from the **MASS** library in R. Let the median value of owner-occupied homes be the response variable. Consider the following predictors:

- Per capita crime rate
- Nitrogen oxides concentration
- Proportion of blacks
- Percentage of lower status population

(a) Fit four separate simple linear regression models by selecting one predictor at a time. Present the outputs in a single table.

(b) Identify the model that provides the best fit.

(c) Compare the estimated coefficients across models and comment on the usefulness of the predictors.

Solution

```
rm(list=ls())
library(stargazer)
library(MASS)
attach(Boston)
v=c(14,1,5,12,13)
d=Boston[,v]
m1=lm(medv~crim,data=Boston);m2=lm(medv~nox,data=Boston)
m3=lm(medv~black,data=Boston);m4=lm(medv~lstat,data=Boston)
stargazer(m1,m2,m3,m4,type="text")

## -----
##                               Dependent variable:
##                               -----
##                               medv
##                               (1)      (2)      (3)      (4)
## -----
##   crim                  -0.415***  

##                           (0.044)
## 
##   nox                  -33.916***  

##                           (3.196)
## 
##   black                 0.034***  

##                           (0.004)
## 
##   lstat                -0.950***  

##                           (0.039)
```

```

## 
## Constant          24.033*** 41.346*** 10.551*** 34.554*** 
##                               (0.409)   (1.811)   (1.557)   (0.563)
## 
## -----
## Observations      506       506       506       506
## R2                0.151     0.183     0.111     0.544
## Adjusted R2       0.149     0.181     0.109     0.543
## Residual Std. Error (df = 504) 8.484     8.323     8.679     6.216
## F Statistic (df = 1; 504)    89.486*** 112.591*** 63.054*** 601.618*** 
## =====
## Note:           *p<0.1; **p<0.05; ***p<0.01

```

Comments

Here model 4, where medv is explained by lstat, has the highest R^2 value, 0.544 .

The coefficients indicate the direction and strength of the relationship between the response and each predictor. Among these, {lstat} shows the strongest association with {medv}.