

Problem Set 2

Sayan Ganguly, Roll: 714

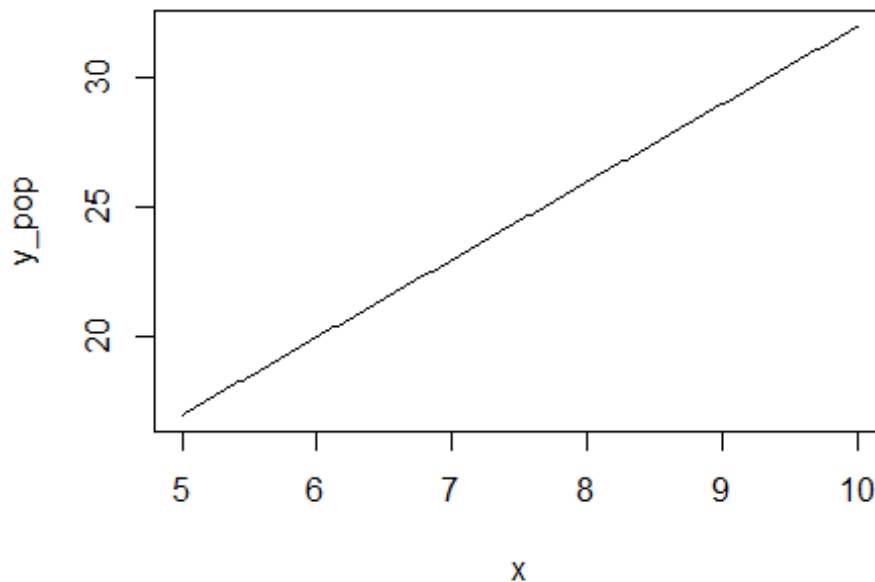
2026-02-05

1) Problem to demonstrate that the population regression line is fixed, but least square regression line varies

Suppose the population regression line is given by $Y = 2 + 3x$, while the data comes from the model $y = 2 + 3x + \epsilon$

Step 1: For x in the range $[5, 10]$ graph the population regression line.

```
x=seq(5,10,length.out=200)
y_pop=2+3*x
a=plot(x,y_pop,type="l")
```



Step 2: Generate $x_i (i = 1, 2, \dots, n)$ from $\text{Uniform}(5, 10)$ and $\epsilon_i (i = 1, 2, \dots, n)$ from $N(0, 4^2)$. Hence, compute y_1, y_2, \dots, y_n

```
x_sample=runif(50,5,10)
e=rnorm(50,0,4)
y_sample=2+3*x_sample+e
y_sample
```

```
## [1] 42.558584 27.934896 23.542664 27.444938 16.764002 21.814665 25.081830
## [8] 21.474486 33.081130 21.865673 8.513729 29.746176 17.096656 27.591482
## [15] 30.759136 29.148207 34.307076 14.205154 21.805306 20.817767 17.441317
## [22] 11.431731 17.479198 26.343469 29.101395 26.778737 21.585650 23.820879
## [29] 28.714906 29.461075 2.603918 19.404206 28.444448 25.537231 16.815919
## [36] 21.714891 14.831169 29.643912 17.518890 18.897566 18.605517 26.929903
## [43] 30.600922 30.454761 28.004982 23.746895 18.974762 32.001756 15.858780
## [50] 30.321136
```

Step 3: On the basis of the data (x_i, y_i) ($i = 1, 2, \dots, n$) generated in Step 2, report the least squares regression line

```
set.seed(123)
lin.reg=lm(y_sample~x_sample)
c=coef(lin.reg)
c

## (Intercept)    x_sample
##   -3.998129     3.687003
```

Step 4: Repeat steps 2-3 five times. Graph the 5 least squares regression lines over the population regression line obtained in Step 1. Interpret the findings

```
x_sample1=runif(50,5,10)
e1=rnorm(50,0,4)
y_sample1=2+3*x_sample1+e1
lin.reg1=lm(y_sample1~x_sample1)

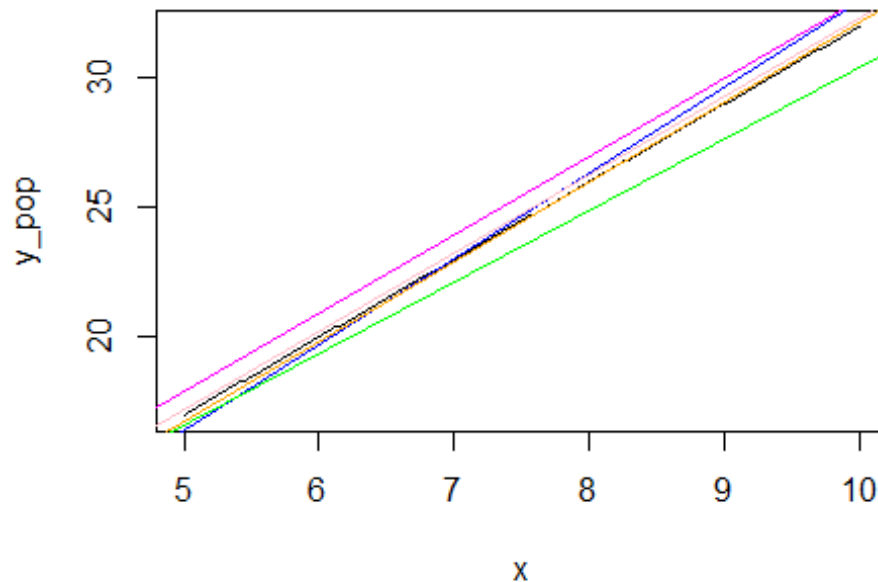
x_sample2=runif(50,5,10)
e2=rnorm(50,0,4)
y_sample2=2+3*x_sample2+e2
lin.reg2=lm(y_sample2~x_sample2)

x_sample3=runif(50,5,10)
e3=rnorm(50,0,4)
y_sample3=2+3*x_sample3+e3
lin.reg3=lm(y_sample3~x_sample3)

x_sample4=runif(50,5,10)
e4=rnorm(50,0,4)
y_sample4=2+3*x_sample4+e4
lin.reg4=lm(y_sample4~x_sample4)

x_sample5=runif(50,5,10)
e5=rnorm(50,0,4)
y_sample5=2+3*x_sample5+e5
lin.reg5=lm(y_sample5~x_sample5)
plot(x,y_pop,type="l")
```

```
abline(lin.reg1,col="blue")
abline(lin.reg2,col="green")
abline(lin.reg3,col="orange")
abline(lin.reg4,col="magenta")
abline(lin.reg5,col="pink")
```



Conclusion: We observe that the regression lines are all distinct as each time the sample produced is different and they obviously differ from the population regression line. But the closeness of the sample regression lines indicate that the mathematical relationship between x and y though not true, still is approximately true.

2 Problem to demonstrate that $\hat{\beta}_0$ and $\hat{\beta}$ minimises RSS

Step 1: Generate x_i from Uniform(5, 10) and mean centre the values. Generate ϵ_i from $N(0,1)$. Calculate $y_i = 2 + 3x_i + \epsilon_i, 1, 2, \dots, n$. Take $n=50$ and seed=123

```
set.seed(123)
xi=runif(50,5,10)
xi_centre=xi-mean(xi)
ei=rnorm(50,0,1)
yi=2+3*xi_centre+ei
yi
```

```
## [1] -3.17439510  6.86099949  0.48666236  6.30575953  9.55945960 -
4.69155288
## [7]  1.82514625  8.48004674  3.34829411  1.86943752  9.23977584
1.55256541
```

```
## [13]  4.30028323  2.48217378 -4.63796535  7.00130299 -2.31796585 -
6.43586794
## [19]  1.28640216  9.72415215  6.41861657  4.18780167  3.33958226
9.89264718
## [25]  3.95085333  5.07991095  2.33107903  3.06789526 -0.09536625 -
3.82043087
## [31] 10.16046950  6.18436828  5.14382834  6.25450092 -5.21621775
1.74521446
## [37]  5.07320502 -2.88845294 -2.04722486 -3.39876904 -3.35583561
0.86504023
## [43]  0.45750453  0.65358464 -1.46460869 -4.21030480 -4.61502197
2.19381069
## [49] -2.52097575  6.37804252
```

Step 2: Now imagine that you only have the data on $(x_i, y_i), i = 1, 2, \dots, n$, without knowing the mechanism that was used to generate the data in step 1. Assuming a linear regression of the type $y_i = \beta_0 + \beta x_i + \epsilon_i$, and based on these data $(x_i, y_i), i = 1, 2, \dots, n$, obtain the least squares estimates of β_0 and β .

```
model=lm(yi~xi_centre)
coef(model)

## (Intercept)    xi_centre
##      2.056189      3.076349
```

****Step 3: Take a large number of grid values of (β_0, β) that also include the least squares estimates obtained from step 2. Compute the RSS for each parametric choice of (β_0, β) , where $RSS = (y_1 - \beta_0 - \beta x_1)^2 + (y_2 - \beta_0 - \beta x_2)^2 + \dots + (y_n - \beta_0 - \beta x_n)^2$. Find out for which combination of (β_0, β) , RSS is minimum.**

```
beta0_grid=seq(coef(model)[1]-2,coef(model)[1]+2,length.out=51)
beta1_grid=seq(coef(model)[2]-2,coef(model)[2]+2,length.out=51)
RSS=matrix(NA,51,51)
for(i in 1:51){
  for(j in 1:51){
    RSS[i,j]=sum((yi-beta0_grid[i]-beta1_grid[j])^2)
    if(RSS[i,j]==1046.49){
      print(beta0_grid[i])
      print(beta1_grid[j])
    }
  }
}
```

3 Problem to demonstrate that least square estimators are unbiased

Step 1: Generate $x_i (i = 1, 2, \dots, n)$ from Uniform(0, 1), $\epsilon_i (i = 1, 2, \dots, n)$ from $N(0, 1)$ and hence generate y using $y_i = \beta_0 + \beta x_i + \epsilon_i$. (Take $\beta_0 = 2, \beta = 3$).

```
set.seed(123)
x=runif(100,0,1)
```

```

e=rnorm(100,0,1)
y=2+3*x+e
y

## [1] 3.1160511 4.3363687 3.1840603 6.0176545 4.5956309 3.6531401
2.0355637
## [8] 5.2618709 3.7781593 3.5857858 5.2501395 2.8576790 3.6995045
2.6993248
## [15] 1.2369828 5.0030036 3.1864730 2.1791828 3.9060296 6.9135956
4.1775868
## [22] 1.7692413 4.9272590 4.2736086 3.2791088 5.1511628 3.3474251
2.5617083
## [29] 3.0487827 2.3024496 4.8948369 5.0921775 3.7014558 5.0307788
1.8533545
## [36] 3.7651699 5.3722176 3.0844053 2.6286114 3.8436850 3.4219039
3.7920360
## [43] 3.4799047 2.4786303 3.8179867 1.8161586 4.8864353 4.9304980
2.5622176
## [50] 3.5470622 1.4270869 3.5834839 4.1500827 2.0181552 2.7312254
2.5745664
## [57] 1.5976905 2.5919817 4.3049096 4.0423849 3.4199986 2.8924863
1.5340262
## [64] 2.7675890 4.9633273 3.6467024 4.5358693 3.7964625 3.5333226
2.2953663
## [71] 4.3810721 2.9401888 3.6399898 1.7457821 5.2698117 2.0084068
3.3748362
## [78] 3.9162739 2.0935371 2.2620982 4.1754093 4.4556708 3.2941733
3.9420907
## [85] 0.2553467 4.4360154 3.4942309 5.4191009 6.5685108 1.0812648
3.0938714
## [92] 3.6971083 1.4584053 2.4556067 1.3595836 2.0321668 2.8851273
2.9687017
## [99] 5.5004461 2.2474859

model=lm(y~x)
coef(model)

## (Intercept)          x
## 1.991040    2.910169

```

Step 2: On the basis of the data (x_i, y_i) ($i = 1, 2, \dots, n$) generated in Step 1, obtain the least square estimates of β_0 and β . Repeat Steps 1-2, $R = 1000$ times. In each simulation obtain $\hat{\beta}_0$ and $\hat{\beta}$. Finally, the least-square estimates will be given by the average of these estimated values. Compare these with the true β_0 and β and comment.

```

z1=c()
z2=c()
for(i in 1:1000){
  xi=runif(50,0,1)
  ei=rnorm(50,0,1)

```

```

yi=2+3*xi+ei
model_i=lm(yi~xi)
z1=append(z1,coef(model_i)[1])
z2=append(z2,coef(model_i)[2])
}
mean(z1)

## [1] 2.013008

mean(z2)

## [1] 2.982695

```

Conclusion: Thus the average values of the estimates of the parameters in different models is so close to the true values thus verifying the unbiased property of the least square estimates.

4 Comparing several simple linear regressions

Attach “Boston” data from MASS library in R. Select median value of owner occupied homes, as the response and per capita crime rate, nitrogen oxides concentration, proportion of blacks and percentage of lower status of the population as predictors.

(a) Selecting the predictors one by one, run four separate linear regressions to the data. Present the output in a single table. (b) Which model gives the best fit? (c)

Compare the coefficients of the predictors from each model and comment on the usefulness of the predictors

```

library(MASS)
head(Boston)

##      crim zn indus chas   nox   rm  age   dis rad tax ptratio  black
lstat
## 1 0.00632 18  2.31    0 0.538 6.575 65.2 4.0900   1  296    15.3 396.90
4.98
## 2 0.02731  0  7.07    0 0.469 6.421 78.9 4.9671   2  242    17.8 396.90
9.14
## 3 0.02729  0  7.07    0 0.469 7.185 61.1 4.9671   2  242    17.8 392.83
4.03
## 4 0.03237  0  2.18    0 0.458 6.998 45.8 6.0622   3  222    18.7 394.63
2.94
## 5 0.06905  0  2.18    0 0.458 7.147 54.2 6.0622   3  222    18.7 396.90
5.33
## 6 0.02985  0  2.18    0 0.458 6.430 58.7 6.0622   3  222    18.7 394.12
5.21
##      medv
## 1 24.0
## 2 21.6
## 3 34.7
## 4 33.4

```

```
## 6 28.7
```

```
library(stargazer)
```

##

Please cite as:

```
## Hlavac, Marek (2022). stargazer: Well-Formatted Regression and Summary
Statistics Tables.
```

```
## R package version 5.2.3. https://CRAN.R-project.org/package=stargazer
```

```
model_1=lm(medv~crim,data=Boston)
```

```
model_2=lm(medv~nox, data=Boston)
```

```
model_3=lm(medv~black,data=Boston)
```

```
model_4=lm(medv~lstat,data=Boston)
```

```
stargazer(model_1,model_2,model_3,model_4,type="text",title="output")
```

##

```
## output
```

```
## =====
```

Dependent variable:

medv

##	(1)	(2)	(3)	(4)
----	-----	-----	-----	-----

```
## crim -0.415***
```

(0.044)

##

```
## nox -33.916***
```

$$\#\# \quad (3.196)$$

##

```
## 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

b) From the values of the R squared and adjusted R squared we can say that the model 4 gives the best fit, i.e. using "lstat" as the predictor we get the best fitted model.

c) We see that all the predictors are significant here since the p values of them are quite small. So the predictors are indeed useful in predicting the response. The coefficients are negative except in case of "black". The negative coefficients indicate that with one unit increase in the predictor value how much the response decreases. We see that with one unit increase in the nox concentration the median price drops sharply.