Regression Matrix Notation

Seema Singh Saharan

March 24, 2017

Let us suppose that we wish to create an OLS model using Matrix Notation.

In this we are going to use a hypothetical case where we generate a population data and then we will draw a sample from it

Where in this model we are using 3 regressors. This can be denoted in matrix notation as follows:

$$Y = \beta_0 + \beta_1 x_1 + \varepsilon$$

$$Y = X\beta + \varepsilon$$

```
# Loading and Running the lm function to generate the Regression Model
Prestige<-read_excel("Prestige.xlsx")
Prestige <- as.data.frame(unclass(Prestige))
summary(Prestige)
```

```
##  c..GOV.ADMINISTRATORS....GENERAL.MANAGERS....ACCOUNTANTS....PURCHASING.OF
FICERS...
##  ACCOUNTANTS         : 1
##  AIRCRAFT.REPAIRMEN: 1
##  AIRCRAFT.WORKERS    : 1
##  ARCHITECTS          : 1
##  AUTO.REPAIRMEN      : 1
##  AUTO.WORKERS        : 1
##  (Other)             :92
##    education          income          women            prestige
##  Min.   : 6.380   Min.   : 1656   Min.   : 0.000   Min.   :17.30
```

```
##  1st Qu.: 8.445   1st Qu.: 4250   1st Qu.: 3.268   1st Qu.:35.38  
##  Median :10.605   Median : 6036   Median :14.475   Median :43.60  
##  Mean   :10.795   Mean   : 6939   Mean   :28.986   Mean   :47.33  
##  3rd Qu.:12.755   3rd Qu.: 8226   3rd Qu.:52.203   3rd Qu.:59.90  
##  Max.   :15.970   Max.   :25879   Max.   :97.510   Max.   :87.20  
##                                                                   
##      census       type   
##  Min.   :1113   bc  :44  
##  1st Qu.:3116   prof:31  
##  Median :5132   wc  :23  
##  Mean   :5400            
##  3rd Qu.:8328            
##  Max.   :9517            
##
```

```
estimated.model<-lm(prestige~income+education,data=Prestige)
summary(estimated.model)
```

```
## 
## Call:
## lm(formula = prestige ~ income + education, data = Prestige)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -16.9367  -4.8881   0.0116   4.9690  15.9280 
## 
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)    
## (Intercept) -7.6210352  3.1162309  -2.446   0.0163 *  
## income       0.0012415  0.0002185   5.682 1.45e-07 ***
## education    4.2921076  0.3360645  12.772  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 7.45 on 95 degrees of freedom
## Multiple R-squared:  0.814,  Adjusted R-squared:  0.8101 
## F-statistic: 207.9 on 2 and 95 DF,  p-value: < 2.2e-16
```

```
# Creating a Regression Model by Matrix Method.This is to confirm

# that the model created by the lm() function is the same as the Regression

# Model created by the Matrix Method.
# Generating the Regressor X matrix and Response matrix for this

# Regrssion Model
X = as.matrix(cbind(1,Prestige$income,Prestige$education))
Y = as.matrix(Prestige$prestige)
head(X)
```

```
##      [,1]  [,2]  [,3]
## [1,]    1 12351 13.11
## [2,]    1 25879 12.26
## [3,]    1  9271 12.77
## [4,]    1  8865 11.42
## [5,]    1  8403 14.62
## [6,]    1 11030 15.64
```

```r
head(Y)
```

```
##      [,1]
## [1,] 68.8
## [2,] 69.1
## [3,] 63.4
## [4,] 56.8
## [5,] 73.5
## [6,] 77.6
```

```
## Estimated Slope Coefficients matrix:
```

$b = (X'X)^{-1} X'y$

```r
beta.hat = solve(t(X)%*%X)%*%t(X)%*%Y


beta.hat.coefficient = as.data.frame(cbind(c("Intercept","Income","Education"
),beta.hat))
names(beta.hat.coefficient) = c("Slope Coefficient.","Estimates")
beta.hat.coefficient
```

```
##   Slope Coefficient.           Estimates
## 1          Intercept   -7.62103523845697
## 2             Income 0.00124153683867422
## 3          Education    4.29210759866114
```

```r
# To calculate the Standard Error of the estimated slope coefficients

# we first calculate the residuals then the Variance Covariance Matrix

and finally the Standard Errors.
```

$residual = Y - (\beta_0 + \beta_1 x_1 + \beta_2 x_2)$

```r
res = as.matrix(Prestige$prestige-beta.hat[1]-beta.hat[2]*Prestige$income-bet
a.hat[3]*Prestige$education)
head(res)
```

```
##           [,1]
## [1,]  4.817283
## [2,] -8.029936
## [3,]  4.700533
## [4,]  4.398942
```

```
## [5,]  7.937788
## [6,]  4.398321

fittedy<-beta.hat[1]-beta.hat[2]*Prestige$income-beta.hat[3]*Prestige$educati
on

## Define n and k parameters
n = nrow(Prestige)
k = ncol(X)
```

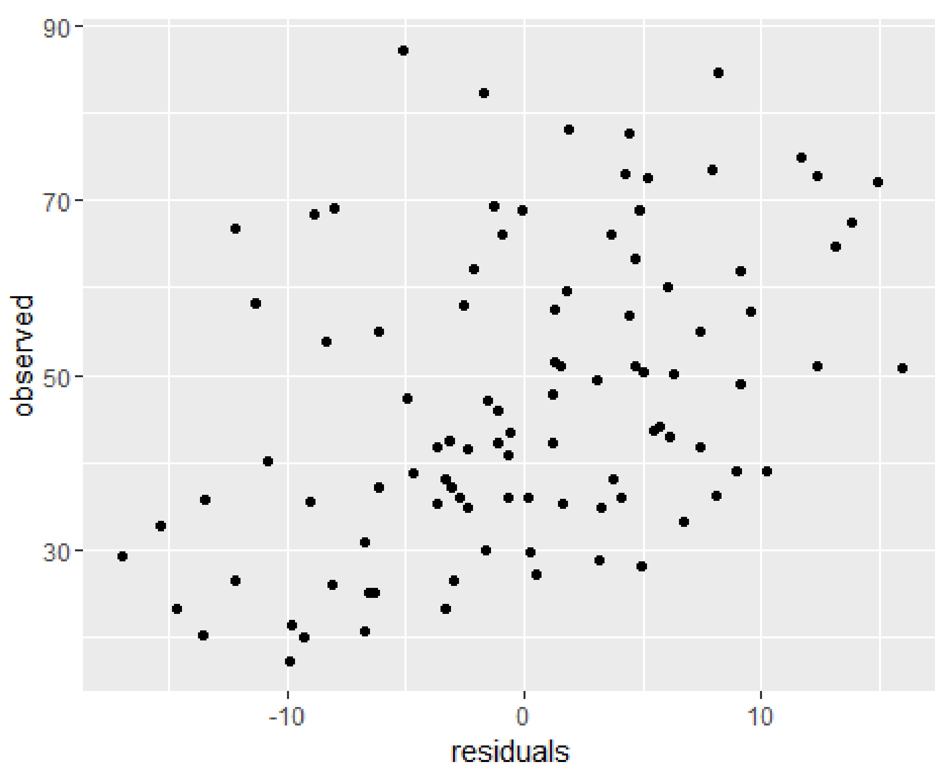
$$\hat{V}(b) = s_E^2 (X'X)^{-1} = \frac{e'e}{n-k}(X'X)^{-1}$$

```
## Calculate Variance-Covariance Matrix
VCV = 1/(n-k) * as.numeric(t(res)%*%res) * solve(t(X)%*%X)
VCV

##                [,1]          [,2]          [,3]
## [1,]  9.7108949267  1.238084e-04 -9.266854e-01
## [2,]  0.0001238084  4.773946e-08 -4.215483e-05
## [3,] -0.9266854238 -4.215483e-05  1.129393e-01

## Standard errors of the estimated coefficients
StdErr = sqrt(diag(VCV))
StdErr

## [1] 3.1162308847 0.0002184936 0.3360644973

# To conduct the individual hypothesis for the estimated coefficients

# we calculate the t values
```

$$t = \frac{B_j}{SE(B_j)}$$

```
t.value <- rbind(beta.hat[1]/StdErr[1],beta.hat[2]/StdErr[2],
                 beta.hat[3]/StdErr[3])
t.value

##           [,1]
## [1,] -2.445594
## [2,]  5.682257
## [3,] 12.771678

## Calculating the  p-value for a t-test for determining coefficient

# significance
P.Value = rbind(2*pt(abs(beta.hat[1]/StdErr[1]), df=n-k,lower.tail= FALSE),
                2*pt(abs(beta.hat[2]/StdErr[2]), df=n-k,lower.tail= FALSE),
                2*pt(abs(beta.hat[3]/StdErr[3]), df=n-k,lower.tail= FALSE))

P.Value
```

```
##              [,1]
## [1,] 1.630283e-02
## [2,] 1.451954e-07
## [3,] 2.453045e-22

## concatenating estimated coefficients, Standard Error, t.value and

# p.value into a single # #data.frame

matrix.notation.results = as.data.frame(cbind(beta.hat,StdErr,t.value,P.Value
))
names(matrix.notation.results) = c("slope Estimates","Standard Errors","tValu
e","pvalue")
matrix.notation.results

##   slope Estimates Standard Errors     tValue       pvalue
## 1    -7.621035238    3.1162308847  -2.445594 1.630283e-02
## 2     0.001241537    0.0002184936   5.682257 1.451954e-07
## 3     4.292107599    0.3360644973  12.771678 2.453045e-22

# creating the residual plots
residual.dataframe <- data.frame(residuals = res, fitted = fittedy, observed
= Y)

sqplot <- ggplot(residual.dataframe, aes(y = observed, x = residuals)) + geom
_point()
print(sqplot)
```



```
# Creating the density plot for residuals
density.plot <- ggplot(residual.dataframe, aes(x = residuals)) + geom_density
()
print(density.plot)
```