

Least Squares Regression

Jerry Kiely

2023-12-17

Contents

1	Derivation	2
2	Estimation	3

1 Derivation

the model can be described as follows:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k + \epsilon$$

where y is the dependent variable, x_1, x_2, \dots, x_k are the independent variables, $\beta_0, \beta_1, \beta_2, \dots, \beta_k$ are the parameters, and ϵ is the error term. The object is to estimate the parameters. We can arrange n observations in matrix / vector form as follows:

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} 1 & x_{11} & x_{12} & x_{13} & \cdots & x_{1k} \\ 1 & x_{21} & x_{22} & x_{23} & \cdots & x_{2k} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_{n1} & x_{n2} & x_{n3} & \cdots & x_{nk} \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_k \end{bmatrix} + \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{bmatrix}$$

which can be simplified as follows:

$$Y = X\beta + \epsilon$$

the vector of parameter estimates is denoted $\hat{\beta}$. To calculate the parameter estimates we want to minimize the sum of the squared errors. The vector of errors is:

$$\epsilon = Y - X\hat{\beta}$$

and hence the sum of the squared errors is:

$$\epsilon^T \epsilon = (Y - X\hat{\beta})^T (Y - X\hat{\beta})$$

to find the minimum of the sum of the squared errors we differentiate the above expression with respect to $\hat{\beta}$:

$$\begin{aligned}
\frac{\partial}{\partial \hat{\beta}} \epsilon^T \epsilon &= \frac{\partial}{\partial \hat{\beta}} (Y - X\hat{\beta})^T (Y - X\hat{\beta}) \\
&= \frac{\partial}{\partial \hat{\beta}} (Y^T - \hat{\beta}^T X^T) (Y - X\hat{\beta}) \\
&= \frac{\partial}{\partial \hat{\beta}} (Y^T Y - Y^T X\hat{\beta} - \hat{\beta}^T X^T Y + \hat{\beta}^T X^T X\hat{\beta}) \\
&= \frac{\partial}{\partial \hat{\beta}} (Y^T Y - 2\hat{\beta}^T X^T Y + \hat{\beta}^T X^T X\hat{\beta}) \\
&= -2X^T Y + 2X^T X\hat{\beta}
\end{aligned}$$

setting this equal to zero we get:

$$\begin{aligned}
-2X^T Y + 2X^T X\hat{\beta} &= 0 \\
2X^T X\hat{\beta} &= 2X^T Y \\
X^T X\hat{\beta} &= X^T Y \\
\hat{\beta} &= (X^T X)^{-1} X^T Y
\end{aligned}$$

2 Estimation

we can now estimate the parameters with real data using the previously derived equation:

```
head(data)
```

```
##      empid   jpi aptitude   tol technical general
## 1      1 45.52   43.83 55.92    51.82   43.58
## 2      2 40.10   32.71 32.56    51.49   51.03
## 3      3 50.61   56.64 54.84    52.29   52.47
## 4      4 38.97   51.53 59.69    47.48   47.69
## 5      5 41.87   51.35 51.50    47.59   45.77
## 6      6 38.71   39.60 43.63    48.34   42.06
```

we construct our X matrix using the independent variables with a column of 1s for the intercept:

```
X <- cbind(rep(1, nrow(data)), as.matrix(data[, c("aptitude", "tol", "technical", "general"))))
head(X)
```

```
##           aptitude    tol technical general
## [1,] 1      43.83 55.92      51.82  43.58
## [2,] 1      32.71 32.56      51.49  51.03
## [3,] 1      56.64 54.84      52.29  52.47
## [4,] 1      51.53 59.69      47.48  47.69
## [5,] 1      51.35 51.50      47.59  45.77
## [6,] 1      39.60 43.63      48.34  42.06
```

next we construct our Y vector from the dependent variable:

```
Y <- as.matrix(data[, "jpi"])
head(Y)
```

```
##           [,1]
## [1,] 45.52
## [2,] 40.10
## [3,] 50.61
## [4,] 38.97
## [5,] 41.87
## [6,] 38.71
```

and finally we solve for our $\hat{\beta}$ vector of estimated parameters using our expression derived earlier:

$$\hat{\beta} = (X^T X)^{-1} X^T Y$$

```
B <- solve(t(X) %*% X) %*% t(X) %*% Y
round(B[, 1], 5)
```

```
##           aptitude      tol technical  general
## -54.28225    0.32356   0.03337   1.09547   0.53683
```

here we use the *solve* function to get the inverse of a matrix, and the *t* function to get the transpose of a vector or matrix. Lets compare our parameters with the parameter estimates using the standard *lm* function:

```
lm(jpi ~ aptitude + tol + technical + general, data = data)
```

```
##
## Call:
## lm(formula = jpi ~ aptitude + tol + technical + general, data = data)
##
## Coefficients:
## (Intercept)      aptitude          tol      technical      general
##   -54.28225       0.32356       0.03337       1.09547       0.53683
```