

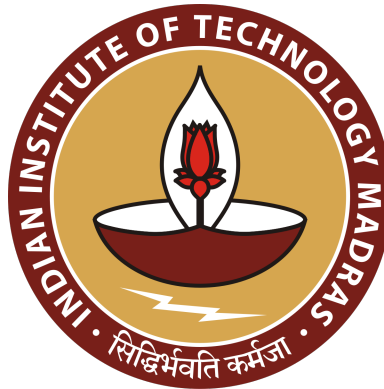
---

# Using Linear Regression to fit models to Data

---

Submitted by:

Atharva Aalok



Department of Aerospace Engineering,  
Indian Institute of Technology Madras

July 14, 2021

# Summary

1	Introduction	2
2	Applying Linear Regression	2
3	Comparing Model with Real Data	3
4	Inferences	6
5	References	7

# 1 Introduction

Linear Regression is a technique using which we can fit models to data which are linear with respect to the parameters used.

Let us say we have some variable  $y$  which depends on an independent variable  $x$ . We want to fit a model which can predict  $y$  given  $x$ . The process of Linear Regression requires us to assume a model to fit to the data. Here we choose the model  $y = mx + c$ .

We may not be able to fit the data perfectly for each data point but we try to fit the data as closely as possible. To do this we define a measure of error in our model. Specifically we choose the sum of squares of error in predicting  $y$  for each data point as our Cost(Error) function. We denote the cost function by  $J$ . We will also refer to our data points as training examples.

Our task then is to choose parameters  $m$  and  $c$  so as to minimize our cost function  $J$ . Hence we have the following problem to solve: If we have  $n$  training examples then our error on each training example is given by  $\epsilon_i$  as

$$\epsilon_1 = mx_1 + c - y_1 \tag{1}$$

$$\epsilon_2 = mx_2 + c - y_2 \tag{2}$$

$$\vdots$$

$$\epsilon_n = mx_n + c - y_n \tag{3}$$

$$\tag{4}$$

and our cost function which is a function of parameters  $m$  and  $c$  is given by,

$$J(m, c) = \sum_{i=1}^n \epsilon_i^2$$

We can write all this succinctly using vector notation as

$$\begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} - \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix} \begin{bmatrix} c \\ m \end{bmatrix}$$

We denote the vector of epsilons by  $\vec{\epsilon}$  the vector of  $c$  and  $m$  by  $\vec{v}$  the vector of  $y_i$ 's by  $\vec{y}$  and the matrix multiplying  $\vec{v}$  by  $\mathbf{A}$ .

We want to minimize:  $\|\vec{\epsilon}\|$

It can be shown using calculus that the parameter vector that minimizes our error function is given By

$$\vec{v} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \vec{y}$$

Using the above approach we have tried to fit a model of the form  $y = mx + c$  to our 3 datasets and analyze whether such a model is a good choice for our data <sup>†</sup>.

## 2 Applying Linear Regression

We perform linear regression on our 3 data sets. To analyze whether a model of the form  $y = mx + c$  is a good approximation we divide each dataset into 3 subsets. We perform linear regression using 50, 100 and 200 training examples of each data set.<sup>††</sup>

---

<sup>†</sup> Find the Datasets here - [GITHUB](#)

<sup>††</sup> For the MATLAB codes for linear regression refer - [GITHUB](#)

The values we obtained are:

DataSetSize	m	c
50	2.011	1.188
100	2.008	1.266
200	2.006	1.377

(a) Data Set 1

DataSetSize	m	c
50	2.011	1.196
100	2.008	1.272
200	2.006	1.381

(b) Data Set 2

DataSetSize	m	c
50	2.000	1.005
100	2.000	1.005
200	2.000	1.005

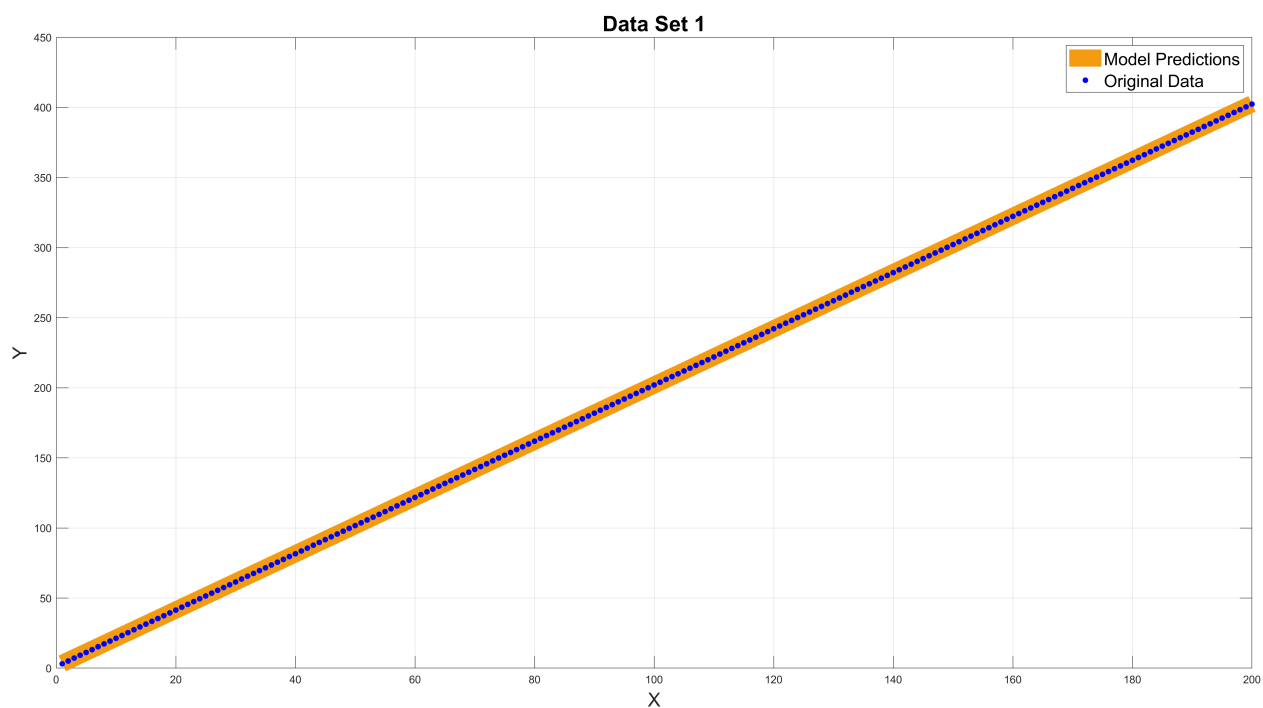
(c) Data Set 3

Table 1: Parameter Values obtained by Linear Regression

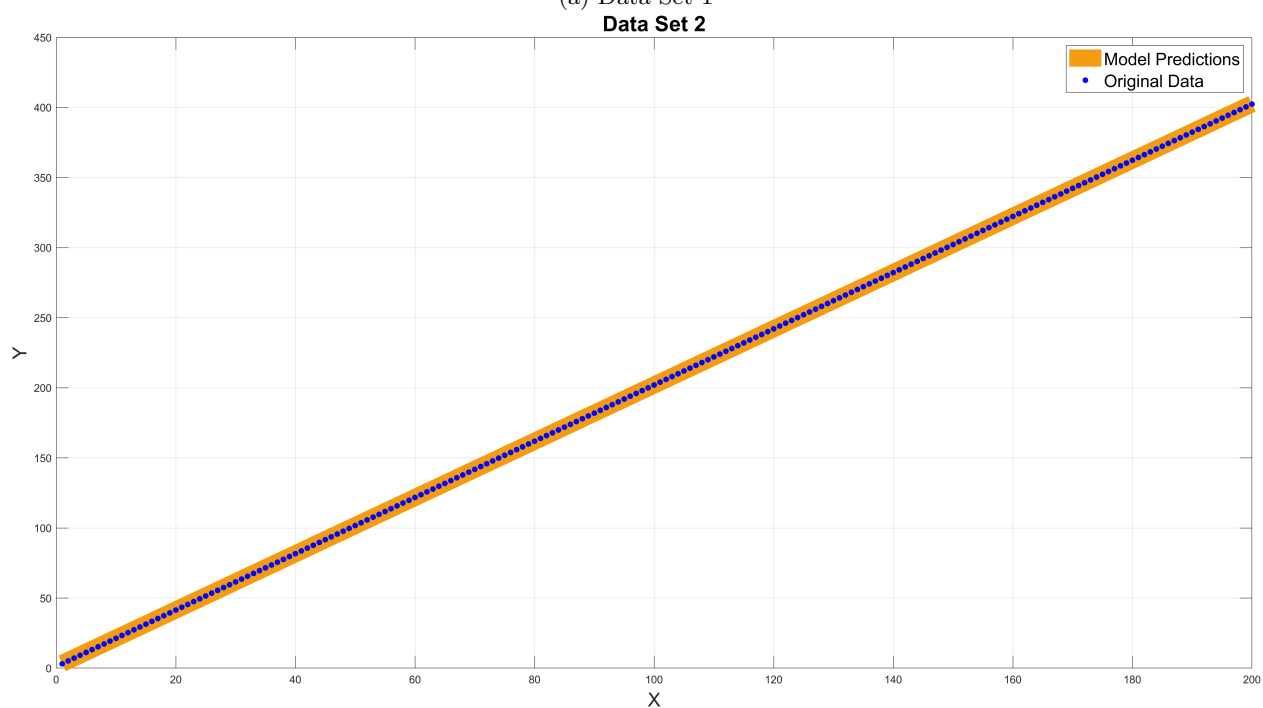
We observe that  $m$  and  $c$  remain almost constant as we vary number of training examples from 50 to 100 to 200. Hence a linear model is a good fit for our data.

### 3 Comparing Model with Real Data

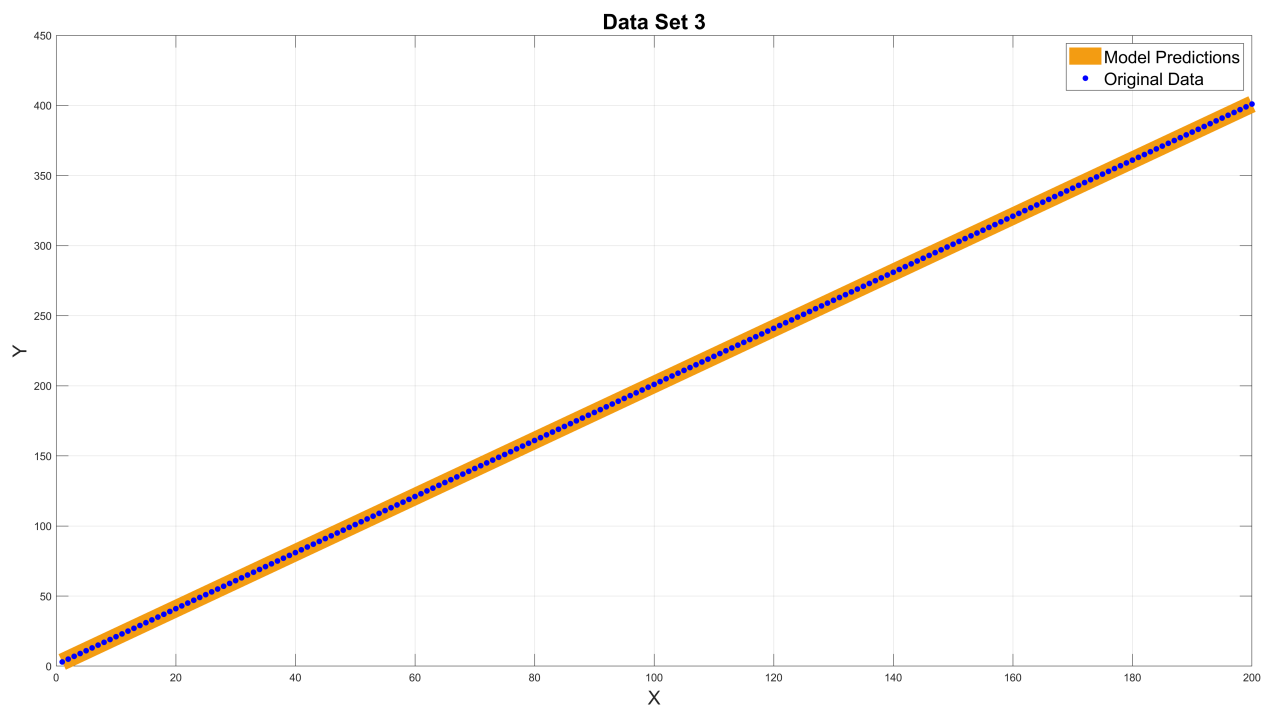
We plot the predictions that our model makes on the input  $x$  and also plot the real data points on top to compare the accuracy of our model visually.



(a) Data Set 1



(b) Data Set 2



(c) Data Set 3

Figure 1: Models fitted using Linear Regression

## 4 Inferences

We observe that  $m$  and  $c$  are almost constant for our data as we vary number of training examples from 50 to 100 to 200. Hence a linear model is a good fit for our data. Also it is visually clear that a linear model fits the original data distribution extremely well.

Since the data agrees very well with our model we use this to make predictions for  $y$ . Hence we make predictions on  $y$  given  $x$  for the 3 data sets as:

Model predictions for Data Set 1:

$$y = 2.006x + 1.377$$

Model predictions for Data Set 2:

$$y = 2.006x + 1.381$$

Model predictions for Data Set 3:

$$y = 2.000x + 1.005$$

## 5 References

- [1] Wikipedia. Gaussian elimination — Wikipedia, the free encyclopedia. <http://en.wikipedia.org/w/index.php?title=Gaussian%20elimination&oldid=1027192624>, 2021. [Online; accessed 16-July-2021].
- [2] Wikipedia. Linear regression — Wikipedia, the free encyclopedia. <http://en.wikipedia.org/w/index.php?title=Linear%20regression&oldid=1032820492>, 2021. [Online; accessed 14-July-2021].
- [3] Wikipedia. Polynomial regression — Wikipedia, the free encyclopedia. <http://en.wikipedia.org/w/index.php?title=Polynomial%20regression&oldid=1032260962>, 2021. [Online; accessed 16-July-2021].