Linear Regression

Linear regression

- Linear regression is one of the most well known algorithm in statistics and machine learning.
- linear regression was developed in the field of statistics
- It is studied as a model for understanding the relationship between input and output numerical variables
- It is both a statistical algorithm and a machine learning algorithm.
- Regression analysis is used for studying the correlation between two sets of variables — between a dependent variable and one or more independent variables.
- Ex: weight loss (dependent variable) depends on the number of hours spent in the gym (independent variable).

Linear regression

- Linear regression is a linear model
- Ex: a model that assumes a linear relationship between the input variables (x) and the single output variable (y).
- More specifically, y can be calculated from a linear combination of the input variables (x).
- When there is a single input variable (x), the method is referred to as simple linear regression.
- When there are multiple input variables, literature from statistics often refers to the method as multiple linear regression.

Linear regression

- Different techniques can be used to prepare or train the linear regression equation from data
- the most common of which is called Ordinary Least Squares.
- It is common to refer to a model prepared this way as Ordinary Least Squares Linear Regression or Least Squares Regression.
- The core idea is to obtain a line that best fits the data.
- The best fit line is the one for which the total prediction error (all data points) are as small as possible.
- Error is the distance between the point to the regression line.

Linear Regression Model Representation

- The representation is a linear equation that combines a set of input values (x) along with its solution which is the predicted output for that set of input values (y).
- Both the input values (x) and the output value are numeric.
- in a simple regression problem (a single x and a single y), the form of the model would be:
- y = B0 + B1 * x
- B1 - scale factor to each input value or column, called a coefficient
- B0 – giving the line an additional degree of freedom (e.g. moving up and down on a two-dimensional plot) and is called the intercept or the bias coefficient
- In higher dimensions when we have more than one input (x), the line is called a plane or a hyper-plane.

Linear Regression Model Representation

- Complexity of a regression model (like linear regression)
- This refers to the number of coefficients used in the model.
- When a coefficient becomes zero, it removes the influence of the input variable on the model and therefore from the prediction made from the model (0 * x = 0).

Linear Regression Learning the Model

- Learning a linear regression model requires estimating the values of the coefficients used in the representation
- Techniques to prepare a linear regression model.

Linear Regression Learning the Model

1. Simple Linear Regression:

- With simple linear regression when we have a single input, we can use statistics to estimate the coefficients.
- This requires that we calculate statistical properties from the data such as mean, standard deviations, correlations and covariance.

Linear Regression Learning the Model

2. Ordinary Least Squares

- When we have more than one input we can use Ordinary Least Squares to estimate the values of the coefficients.
- The Ordinary Least Squares procedure seeks to minimize the sum of the squared residuals.
- This means that given a regression line through the data we calculate the distance from each data point to the regression line, square it, and sum all of the squared errors together.
- This is the quantity that ordinary least squares seeks to minimize.
- This approach treats the data as a matrix and uses linear algebra operations to estimate the optimal values for the coefficients.

Linear Regression Learning the Model

2. Ordinary Least Squares

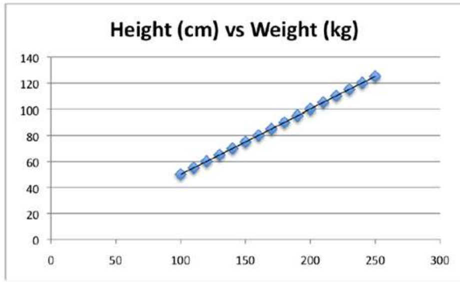
- When we have more than one input we can use Ordinary Least Squares to estimate the values of the coefficients.
- The Ordinary Least Squares procedure seeks to minimize the sum of the squared residuals.
- This means that given a regression line through the data we calculate the distance from each data point to the regression line, square it, and sum all of the squared errors together.
- This is the quantity that ordinary least squares seeks to minimize.
- This approach treats the data as a matrix and uses linear algebra operations to estimate the optimal values for the coefficients.

Making Predictions with Linear Regression

- Given the representation is a linear equation, making predictions
- involves solving the equation for a specific set of inputs
- Ex: Imagine we are predicting weight (y) from height (x).
- Our linear regression model representation for this problem:
- y = B0 + B1 * X1
- weight = B0 + B1 * height
- B0 is the bias coefficient and B1 is the coefficient for the height column.
- We use a learning technique to find a good set of coefficient values. Once found, we can plug in different height values to predict the weight.
- Ex: let's use B0 = 0.1 and B1 = 0.5. Calculate the weight (in kilograms) for a person with the height of 182 centimeters.
- weight = 0.1 + 0.5* 182=91.1
- Plot the above equation as a line in two-dimensions.
- The B0 is our starting point regardless of height.

Making Predictions with Linear Regression

Sample Height vs Weight Linear Regression



Run through heights from 100 to 250 centimeters and get weight values, creating our line

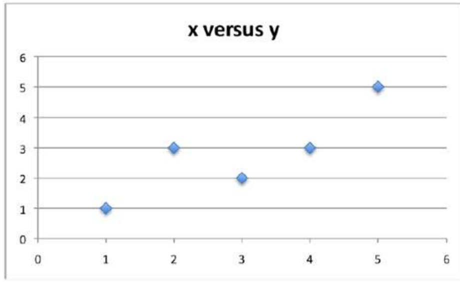
Simple Linear Regression- Example

x	y
1	1
2	3
4	3
3	2
5	5

- Data Set
- The attribute x is the input variable
- y is the output variable we are trying to predict.
- We can see the relationship between x and y looks kind-of linear.
- Plot the line
- to generally describe the relationship between the data.
- This is a good indication that using linear regression might be appropriate for this dataset.

Simple Linear Regression- Example

- Simple Linear Regression Dataset: Scatter plot of x versus y.



Simple Linear Regression- Example

- Simple linear regression: When we have a single input attribute (x) and we use linear regression
- Multiple linear regression: we have multiple input attributes (e.g. X1, X2, X3, etc.)
- Problem Statement: Create a simple linear regression model from our training data, then make predictions for our training data and identify how well the model learned the relationship in the data.

Simple Linear Regression- Example

- With simple linear regression we model our data as follows:
- y = B0 + B1 * x
- This is a line where
  - y is the output variable we want to predict,
  - x is the input variable
  - B0 and B1 are coefficients
  - B0 is called the intercept because it determines where the line intercepts the y-axis.
  - In machine learning we can call this the bias, because it is added to offset all predictions that we make.
  - The B1 term is called the slope because it defines the slope of the line or how x translates into a y value before we add our bias.

Simple Linear Regression- Example

- The goal is to find the best estimates for the coefficients to minimize the errors in predicting y from x
- In Simple regression we can estimate coefficients directly from our data.
- We can start off by estimating the value for B1 as:

$$B1 = \frac{\sum_{i=1}^{n}(x_i - mean(x)) \times (y_i - mean(y))}{\sum_{i=1}^{n}(x_i - mean(x))^2}$$

- Where mean() is the average value for the variable in our dataset.
- The xi and yi refer to the fact that we need to repeat these calculations across all values in our dataset
- i refers to the ith value of x or y.
- We can calculate B0 using B1 and some statistics from our dataset, as follows:
- B0 = mean(y) - B1 * mean(x)

Simple Linear Regression- Estimating The Slope (B1)

- Let's start with the top part of the equation, the numerator. First we need to calculate the
- mean value of x and y. The mean is calculated as:

$$\frac{1}{n} \times \sum_{i=1}^{n} x_i$$

- Where n is the number of values (5 in this case).
- calculate the mean value of our x and y variables:
- mean(x) = 3
- mean(y) = 2.8

Simple Linear Regression- Example

- Calculate the error of each variable from the mean

Residual of each x value from the mean.

x	mean(x)	x - mean(x)
1	3	-2
2		-1
4		1
3		0
5		2

Residual of each y value from the mean.

y	mean(y)	y - mean(y)
1	2.8	-1.8
3		0.2
3		0.2
2		-0.8
5		2.2

Simple Linear Regression- Example

- multiply the error for each x with the error for each y and calculate the sum of these multiplications.

x - mean(x)	y - mean(y)	Multiplication
-2	-1.8	3.6
-1	0.2	-0.2
1	0.2	0.2
0	-0.8	0
2	2.2	4.4

Simple Linear Regression- Example

- Summing the final column we have calculated our numerator as 8.
- Now we need to calculate the denominator of the equation for calculating B1
- This is calculated as the sum of the squared differences of each x value from the mean.
- We have already calculated the difference of each x value from the mean, so we square each value and calculate the sum.
- Squared residual of each x value from the mean.

x - mean(x)	squared
-2	4
-1	1
1	1
0	0
2	4

Simple Linear Regression- Example

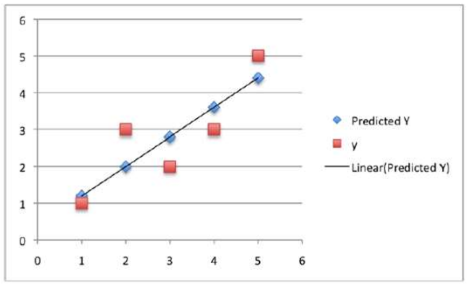
- Calculating the sum of these squared values gives the denominator of 10.
- Now we can calculate the value of our slope.
- B1 = 8/10=0.8

Simple Linear Regression- Example

- Estimating the Intercept (B0)
- B0 = mean(y) - B1 * mean(x)
- B0 = 2.8 – 0.8 * 3 = 0.4
- Making Predictions
- We now have the coefficients for our simple linear regression equation.
- y = B0 + B1 * x
- y = 0.4 + 0.8 * x
- Problem: Try out the model by making predictions for our training data and plot these predictions as a line with our data. This gives a visual idea of how well the line models our data.

Simple Linear Regression- Example

Simple Linear Regression Predictions (x vs y in red and x vs prediction in blue)



Note: create a scatter plot of the data points and a line plot of the predictions.

Simple Linear Regression- Example

- Estimating Error
- We can calculate an error score for our predictions called the Root Mean Squared Error or RMSE.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(p_i - y_i)^2}{n}}$$

- Error for predicted values

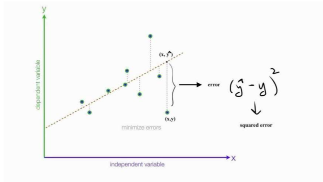
Predicted	y	Predicted - y
1.2	1	0.2
2	3	-1
3.6	3	0.6
2.8	2	0.8
4.4	5	-0.6

Squared error for predicted values

Predicted - y	squared error
0.2	0.04
-1	1
0.6	0.36
0.8	0.64
-0.6	0.36

Simple Linear Regression- Example

- The sum of these errors is 2.4 units, dividing by 5 and taking the square root gives us:
- RMSE = 0.692820323
- Each prediction is on average wrong by about 0.692 units.



Simple Regression Model

- y = B0 + B1 * x
- y- response variable, x- predictor variable
- Regression model is said to be simple, linear in the parameters, and linear in the predictor variable.
- "simple" - there is only one predictor variable,
- "linear in the parameters" - because no parameter appears as an exponent or is multiplied or divided by another parameter
- "linear in the predictor variable" because this variable appears only in the first power.
- A model that is linear in the parameters and in the predictor variable is also called a first-order model.

Simple Linear Regression – Derivation of coefficients

1) Pedhazur formula – intuitive
2) Calculus method – Partial Derivatives
3) Matrix Formulation

Pedhazur formula – intuitive

- To find the slope, Pedhazur formula is:
- This says that the slope B1(here b) is the sum of deviation cross products divided by the sum of squares for X

$$B1 = \frac{\sum_{i=1}^{n}(x_i - mean(x)) \times (y_i - mean(y))}{\sum_{i=1}^{n}(x_i - mean(x))^2} \quad \text{or} \quad b = \frac{\sum xy}{\sum x^2}$$

- To find B0(here a), subtract and rearrange to find the intercept
- B0= y- B1 * x Here y=(y_i-mean(y) and x=(x_i-mean(x)

https://medium.com/analytics-vidhya/linear-regression-an-intuitive-approach-ded05ed39a87

Pedhazur formula – intuitive

- If there is no relationship between X and Y, the best guess for all values of X is the mean of Y.
- If there is a relationship (slope is not zero), the best guess for the mean of X is still the mean of Y
- As X departs from the mean, so does Y.
- At any rate, the regression line always passes through the mean of X and mean of Y.
- This means that, regardless of the value of the slope, when X is at its mean, so is Y

Pedhazur formula – intuitive

Example: you want to know to what degree the tip amount can be predicted by the bill studied. The tip is the dependent variable (response variable) and the bill is the independent variable (predictor variable).

x	y
Total Bill($)	Tip Amount($)
34	5
108	17
64	11
88	8
99	14
51	5

Pedhazur formula – intuitive

Example: you want to know to what degree the tip amount can be predicted by the bill studied. The tip is the dependent variable (response variable) and the bill is the independent variable (predictor variable).

$$B1 = \frac{\sum_{i=1}^{n}(x_i - mean(x)) \times (y_i - mean(y))}{\sum_{i=1}^{n}(x_i - mean(x))^2}$$

B0= y- B1 * x Here y=(y_i-mean(y) and x=(x_i-mean(x)

	x	y	(x-x_bar)	(y-y_bar)	(x-x_bar)(y-y_bar)	(x-x_bar)^2
Meal	Total Bill($)	Tip Amount($)	Bill Deviation	Tip Deviations	Deviation Products	Bill deviation squared
1	34	5	-40	-5	200	1600
2	108	17	34	7	238	1156
3	64	11	-10	1	-10	100
4	88	8	14	-2	-28	196
5	99	14	25	4	100	625
6	51	5	-23	-5	115	529
Total	74	10	0	0	615	4206

Pedhazur formula – intuitive

Example: you want to know to what degree the tip amount can be predicted by the bill studied. The tip is the dependent variable (response variable) and the bill is the independent variable (predictor variable).

$\bar{x}$ = 74, $\bar{y}$ = 10 , Σ (x-x̄)(y-ȳ) = 615 , Σ (x-x̄)^2 = 4206.
Putting all the values in β1, you get β1 0.1462.
This means that if you increase the bill by 1 unit, the tip amount will increase by 0.1462 units.
Similarly, β0 = -0.8203.
The intercept may or may not have any meaning in real life.
Hence, the equation of the best fit line is Ŷ = 0.1462x - 0.8203.

Implementation: 1. Plot the points.
2. Use the numpy/for loop
3. Estimate B0 & B1 substitute in the y = B0 +B1*x
4. Test the model with x value inorder to predict y value

Calculus method - Method of least-squares

- To find "good" estimators of the regression parameters B0 and B1, we employ the method of least squares.
- For the observations (Xi, Yi) for each case, the method of least squares considers the deviation of Yi from its expected value:

$$Y_i - (\beta_0 + \beta_1 X_i)$$

- In particular, the method of least squares requires that we consider the sum of the n squared deviations denoted by S_r

$$S_r = \sum_{i=1}^{n} E_i^2 = \sum_{i=1}^{n} (y_i - a_0 - a_1 x_i)^2$$

- Use calculus: take derivative with respect to B0 and with respect to B1 and then set the two result equations equal to zero and solve for B0 and B1
- Note: β_0 and β_1 denoted by a_0 and a_1
- One way to compute the minimum of a function is to set the partial derivatives to zero.

Calculus method - Method of least-squares

$$S_r = \sum_{i=1}^{n} E_i^2 = \sum_{i=1}^{n} (y_i - a_0 - a_1 x_i)^2$$

S_r is called the sum of the square of the residuals.

To find a_0 and a_1, we minimize S_r with respect to a_0 and a_1

$$\frac{\partial S_r}{\partial a_0} = 2\sum_{i=1}^{n} (y_i - a_0 - a_1 x_i)(-1) = 0$$

$$\frac{\partial S_r}{\partial a_1} = 2\sum_{i=1}^{n} (y_i - a_0 - a_1 x_i)(-x_i) = 0$$

Note: β_0 and β_1 denoted by a_0 and a_1

Calculus method - Method of least-squares

$$-\sum_{i=1}^{n} y_i + \sum_{i=1}^{n} a_0 + \sum_{i=1}^{n} a_1 x_i = 0$$

Noting that $\sum_{i=1}^{n} a_0 = a_0 + a_0 + \ldots + a_0 = na_0$

$$na_0 + a_1 \sum_{i=1}^{n} x_i = \sum_{i=1}^{n} y_i$$

$$-\sum_{i=1}^{n} y_i x_i + \sum_{i=1}^{n} a_0 x_i + \sum_{i=1}^{n} a_1 x_i^2 = 0$$

$$a_0 \sum_{i=1}^{n} x_i + a_1 \sum_{i=1}^{n} x_i^2 = \sum_{i=1}^{n} x_i y_i$$

Note: In matrix form: $\mathbf{X'X} = \begin{bmatrix} n & \sum_{i=1}^{n} X_i \\ \sum_{i=1}^{n} X_i & \sum_{i=1}^{n} X_i^2 \end{bmatrix}$ $\mathbf{X'Y} = \begin{bmatrix} \sum_{i=1}^{n} Y_i \\ \sum_{i=1}^{n} X_i Y_i \end{bmatrix}$ Defining $\mathbf{b} = \begin{bmatrix} b_0 \\ b_1 \end{bmatrix}$

$$\Rightarrow \mathbf{X'Xb} = \mathbf{X'Y} \Rightarrow \mathbf{b} = (\mathbf{X'X})^{-1} \mathbf{X'Y}$$

Note: β_0 and β_1 denoted by a_0 and a_1

Calculus method - Method of least-squares

- Least squares is a method to apply linear regression.
- It helps us predict results based on an existing set of data as well as clear anomalies in our data.
- Anomalies are values that are too good, or bad, to be true or that represent rare cases.

Calculus method - Method of least-squares

Example: we have a list of how many topics future engineers here at free Code Camp can solve if they invest 1, 2, or 3 hours continuously. Then we can predict how many topics will be covered after 4 hours of continuous study even without that data being available to us.

Hours (X)	Topics Solved (Y)
1	1.5
1.2	2
1.5	3
2	1.8
2.3	2.7
2.5	4.7
2.7	7.1
3	10
3.1	6
3.2	5
3.6	8.9

Hours (X)	Topics Solved (Y)	$(X - \bar{x})$	$(y - \bar{y})$	$(X - \bar{x})*(y - \bar{y})$	$(x - \bar{x})^2$
1	1.5	-1.37	-3.29	4.51	1.88
1.2	2	-1.17	-2.79	3.26	1.37
1.5	3	-0.87	-1.79	1.56	0.76
2	1.8	-0.37	-2.99	1.11	0.14
2.3	2.7	-0.07	-2.09	0.15	0.00
2.5	4.7	0.13	-0.09	-0.01	0.02
2.7	7.1	0.33	2.31	0.76	0.11
3	10	0.63	5.21	3.28	0.40
3.1	6	0.73	1.21	0.88	0.53
3.2	5	0.83	0.21	0.17	0.69
3.6	8.9	1.23	4.11	5.06	1.51

The formula
Y = a + bX

x -> 1+1.2+1.5+2+2.3+2.5+2.7+3+3.1+3.2+3.6 = **2.37**
$\bar{y}$ -> 1,5+2+3+1,8+2,7+4,7+7,1+10+6+5+8,9 / 11 = **4.79**

Calculus method - Method of least-squares

Calculating "b"

The weird symbol sigma (∑) tells us to sum everything up:
$\sum(x - \bar{x})*(y - \bar{y})$ -> 4.51+3.26+1.56+1.11+0.15+-0.01+0.76+3.28+0.88+0.17+5.06 = **20.73**
$\sum(x - \bar{x})^2$ -> 1.88+1.37+0.76+0.14+0.00+0.02+0.11+0.40+0.53+0.69+1.51 = **7.41**
And finally we do **20.73 / 7.41** and we get **b = 2.8**

Calculating "a"
All that is left is **a**, for which the formula is **$\bar{y}$ = a + b $\bar{x}$.** We've already obtained all those other values, so we can substitute them and we get:
- 4.79 = **a** + 2.8*2.37
- 4.79 = **a** + 6.64
- **a** = -6.64+4.79
- **a = -1.85**

Calculus method - Method of least-squares

The result
Our final formula becomes:
Y = -1.85 + 2.8*X

Now we replace the **X** in our formula with each value that we have:

Hours (X)	-1.85 + 2.8 * X
1	0.95
1.2	1.51
1.5	2.35
2	3.75
2.3	4.59
2.5	5.15
2.7	5.71
3	6.55
3.1	6.83
3.2	7.11
3.6	8.23

Implementation: 1. Plot the points.
2. Use the numpy/for loop
3. Estimate B0 & B1 substitute in the y = B0 +B1*x
4. Test the model with x value inorder to predict y value
5. Plot the line

Matrix Formulation of Regression Model

- In the multiple regression setting, there may be a potentially large number of predictors (input variables x)
- It is more efficient to use matrices to define the regression model and the subsequent analyses.
- So apply regression formulas in matrix form.

Matrix Formulation of Regression Model

Consider the following simple linear regression function

$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i \quad \text{for } i = 1, \ldots, n$$

Let i = 1, ..., n, we see that we obtain n equations:

$$\begin{aligned} y_1 &= \beta_0 + \beta_1 x_1 + \epsilon_1 \\ y_2 &= \beta_0 + \beta_1 x_2 + \epsilon_2 \\ &\vdots \\ y_n &= \beta_0 + \beta_1 x_n + \epsilon_n \end{aligned}$$

Formulate the above simple linear regression function in matrix notation:

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix}$$

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

Matrix Formulation of Regression Model

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix}$$

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

- ***X*** is an $n \times 2$ **matrix**.
- ***Y*** is an $n \times 1$ **column vector**, ***β*** is a 2×1 column vector, and ***ε*** is an $n \times 1$ column vector.
- The matrix ***X*** and vector ***β*** are multiplied together using the techniques of **matrix multiplication**.
- And, the vector ***Xβ*** is added to the vector ***ε*** using the techniques of **matrix addition**.
- Note: A column vector is an r × 1 matrix, that is, a matrix with only one column. A row vector is an 1 × c matrix, that is, a matrix with only one row.
- Note: A matrix denoted by an uppercase letter and a vector denoted by a single lowercase letter

Matrix Formulation of Regression Model

- Matrix multiplication: Xβ appears in the regression function

$$X = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix}$$

- if X is an n × (k+1) matrix and β is a (k+1) × 1 column vector, then the matrix multiplication Xβ is possible.
- The resulting matrix Xβ has n rows and 1 column. That is, Xβ is an n × 1 column vector
- Note: b is a (k+1) × 1 vector $b = \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_k \end{bmatrix}$
- β is (n*1) vector, here (2*1) vector $\begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix}$

Matrix Formulation of Regression Model

- Least squares estimates in matrix notation
- (k+1) × 1 vector containing the estimates of the (k+1) parameters of the regression function can be shown to equal:

$$b = \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_k \end{bmatrix} = (X'X)^{-1}X'Y$$

- $(X'X)^{-1}$ is the inverse of the X'X matrix, and
- X' is the transpose of the X matrix.
- Note: The inverse only exists for square matrices

Matrix Formulation of Regression Model

- X matrix in the simple linear regression setting

$$X = \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & \vdots \\ 1 & x_n \end{bmatrix}$$

- X'X matrix in the simple linear regression setting

$$X'X = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ x_1 & x_2 & \cdots & x_n \end{bmatrix} \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ \vdots & x_n \\ 1 & \end{bmatrix} = \begin{bmatrix} n & \sum_{i=1}^n x_i \\ \sum_{i=1}^n x_i & \sum_{i=1}^n x_i^2 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 1 & \cdots & 1 \\ x_1 & x_2 & \cdots & x_n \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \qquad X'Y = \begin{bmatrix} \sum_{i=1}^n y_i \\ \sum_{i=1}^n x_i y_i \end{bmatrix}$$

Matrix Formulation of Regression Model

$$X'X = \begin{bmatrix} n & \sum_{i=1}^n x_i \\ \sum_{i=1}^n x_i & \sum_{i=1}^n x_i^2 \end{bmatrix}$$

$$X'Y = \begin{bmatrix} \sum_{i=1}^n y_i \\ \sum_{i=1}^n x_i y_i \end{bmatrix}$$

find the inverse $(X'X)^{-1}$

- Find the least square estimate:

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

- Note: vector-matrix form y = bX can be solved as b= X^{-1} y

Problem Statement 1

- Check if we can obtain the same answer using the above matrix formula.

x	y
1	1
2	3
4	3
3	2
5	5

Preparing Data For Linear Regression

- How our data must be structured to make best use of the model?
- In practice, we can uses these rules as heuristics when using Ordinary Least Squares Regression, the most common implementation of linear regression.

1. Linear Assumption: Linear regression assumes that the relationship between our input and output is linear.

- When we have a lot of attributes. we may need to transform data to make the relationship linear (e.g. log transform for an exponential relationship).

2. Remove Noise: Linear regression assumes that our input and output variables are not noisy.

- Consider using data cleaning operations that better clarifies our data.
- This is most important for the output variable and we want to remove outliers in the output variable (y) if possible.

Preparing Data For Linear Regression

- How our data must be structured to make best use of the model?

3. Remove Collinearity: Linear regression will overfit our data when we have highly correlated input variables. Consider calculating pairwise correlations for our input data and removing the most correlated.

4. Gaussian Distributions. Linear regression will make more reliable predictions if our input and output variables have a Gaussian distribution.

- We may get benefit using transforms (e.g. log or BoxCox) on our variables to make their distribution more Gaussian looking.

5. Rescale Inputs: Linear regression will often make more reliable predictions if you rescale

- input variables using standardization or normalization.