Regression

Week 3

McMaster University
CENTRE FOR CONTINUING EDUCATION

What is Regression?

Input variables, Features, X

Output, Label, Target, Y

Crime Rate	RM	Age	Tax	Price
0.00632	6.575	65.2	296	240000
0.02731	6.421	78.9	242	216000
0.02729	7.185	61.1	242	347000
0.03237	6.998	45.8	222	334000
0.06905	7.147	54.2	222	362000
0.02985	6.43	58.7	222	287000
0.08829	6.012	66.6	311	229000
0.14455	6.172	96.1	311	??

Can we predict the price of a home based on input features?

Other Examples of Regression

Label (Y)	Features (X)
Salary	Occupation, Education, Work Experience, etc.
Pollution	Engine Size, Year, Make, Model, etc.
Sales	Age, Education, Work Experience, etc.
Weather	Hour, Month, Location, etc.

Practice: Identify the regression problem,

1- Predicting Normal vs Abnormal

2- Predicting if stock will rise or fall

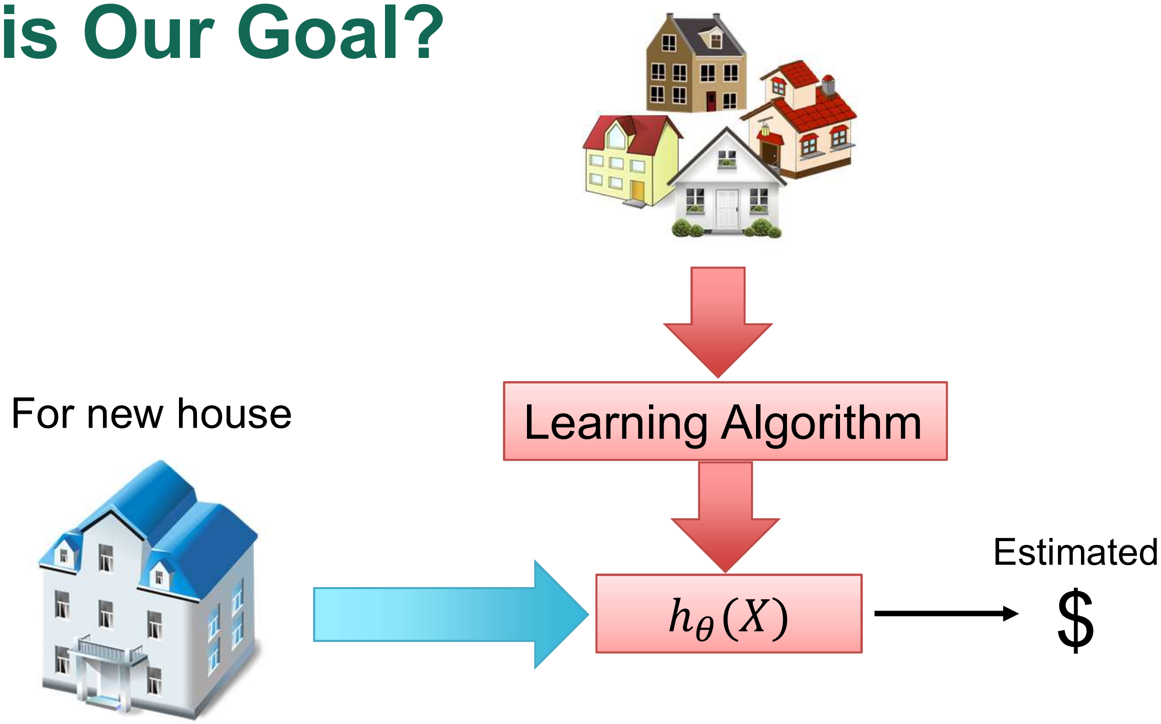
3- Grouping similar cars

4- Predicting stock price for tomorrow

Regression Algorithms

- Bayesian linear regression
- Neural network regression
- k-nearest neighbors
- Ordinal regression
- Linear, polynomial,…

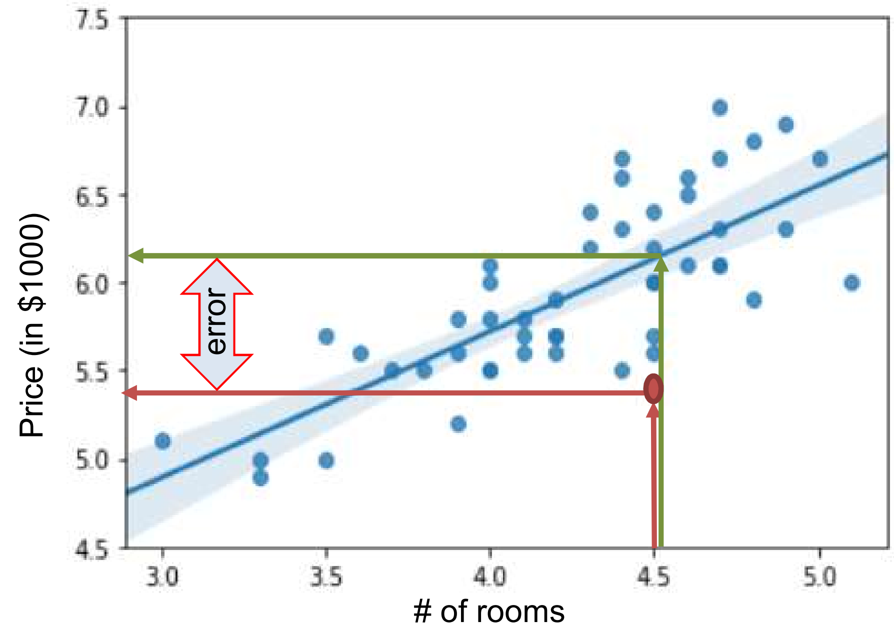
What is Our Goal?



- Learning algorithm outputs a function called hypothesis.
- Hypothesis function gets X_new and transforms them to y_hat.

Linear Regression

$$h_\theta(X) = \theta_0 + \theta_1 x_1$$



Problem:

Find θ_0 and θ_1 such that mean square error gets minimum for all training samples

$$\underset{\theta}{Minimize}: \quad \frac{1}{2m}\sum_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)^2$$

also call cost function $J(\theta_0, \theta_1)$

Recap

Hypothesis:

$$h_\theta(X) = \theta_0 + \theta_1 x_1$$

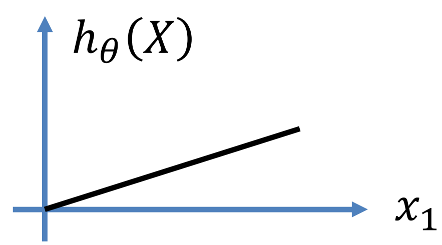
Cost Function:

$$J(\theta_0, \theta_1) = \frac{1}{2m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2$$

Goal:

$$\underset{\theta_0, \theta_1}{Minimize}: \quad J(\theta_0, \theta_1)$$

For better visualization of $J(\theta_0, \theta_1)$, lets assume $\theta_0 = \mathbf{0}$



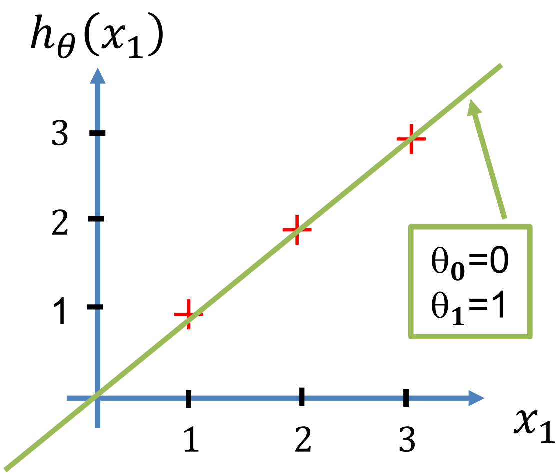
Visualizing $h_\theta(X)$ and $\boldsymbol{J}(\theta)$

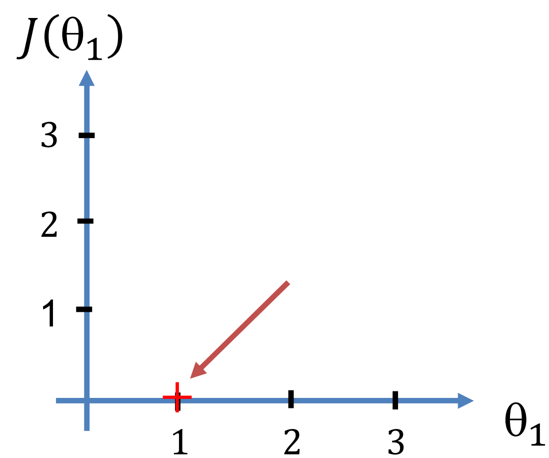
Hypothesis:

$$h_\theta(X) = \theta_{\mathbf{0}} + \theta_{\mathbf{1}} x_1$$

Cost Function:

$$\boldsymbol{J}(\theta_{\mathbf{0}}, \theta_{\mathbf{1}}) = \frac{1}{2m}\sum_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)^2$$





$J(0,1) = \frac{1}{6}((1-1)^2 + (2-2)^2 + (3-3)^2) = 0$

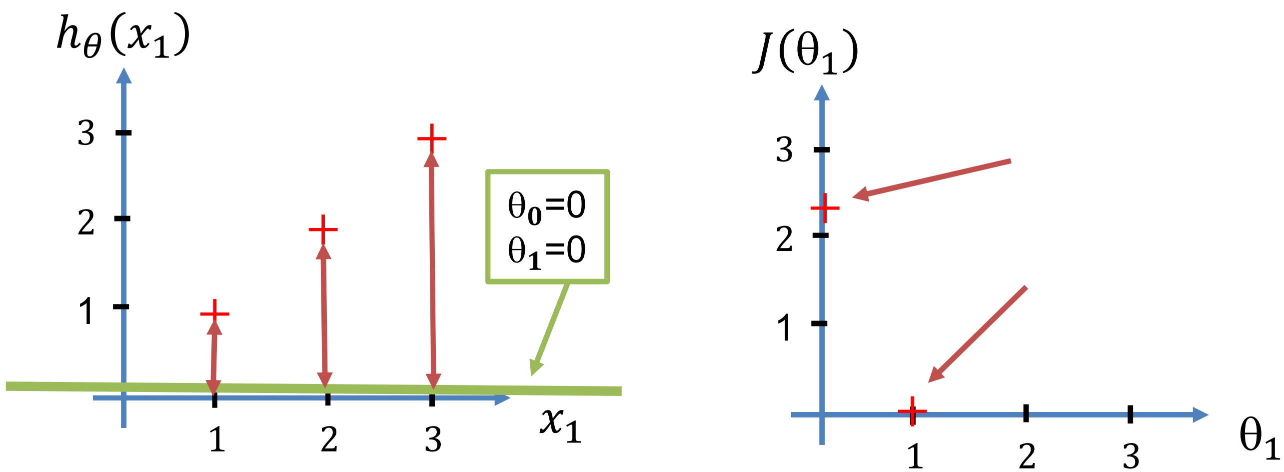
Visualizing $h_\theta(X)$ and $J(\theta)$

Hypothesis:

$$h_\theta(X) = \theta_0 + \theta_1 x_1$$

Cost Function:

$$J(\theta_0, \theta_1) = \frac{1}{2m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2$$

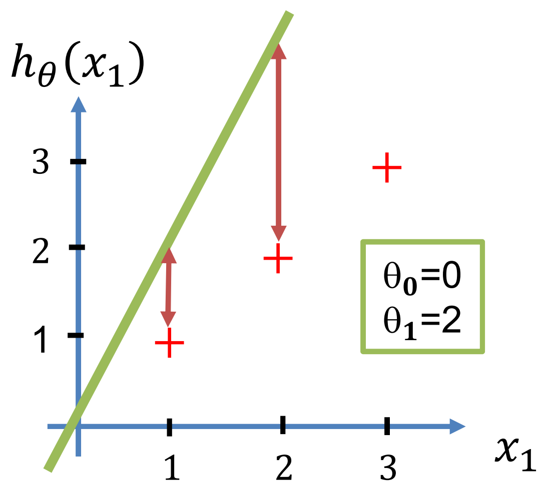


$J(0,0) = \frac{1}{6}((0-1)^2 + (0-2)^2 + (0-3)^2)$=2.33

Visualizing $h_\theta(X)$ and $\boldsymbol{J}(\theta)$

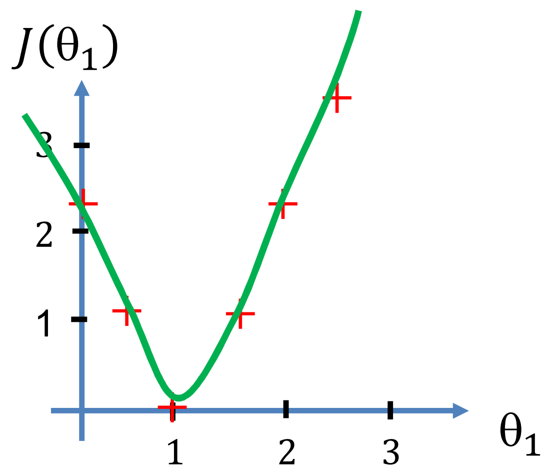
Hypothesis:

$$h_\theta(X) = \theta_0 + \theta_1 x_1$$



Cost Function:

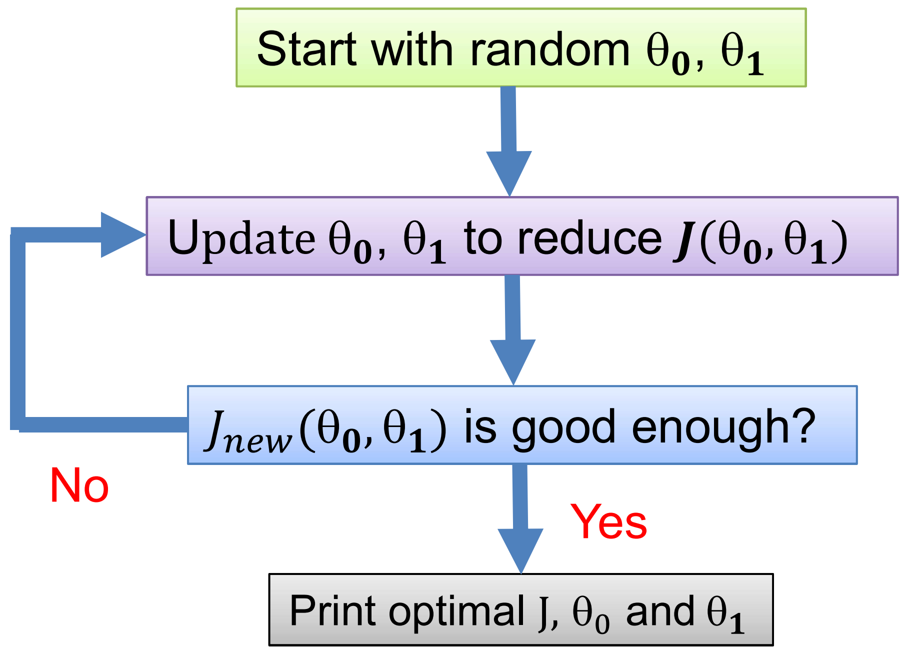
$$\boldsymbol{J}(\theta_0, \theta_1) = \frac{1}{2m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2$$



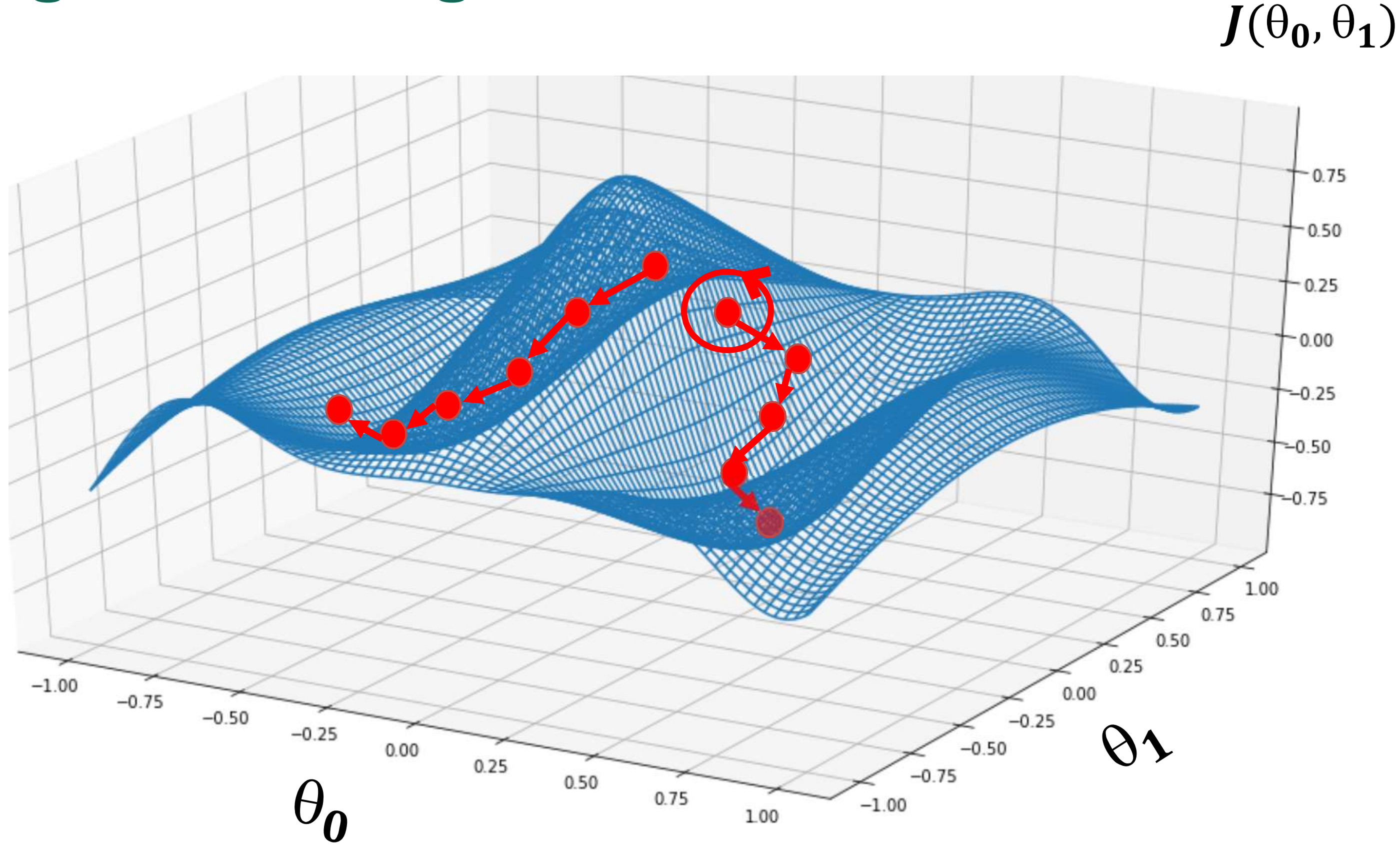
$J(0,2) = \frac{1}{6}((2-1)^2 + (4-2)^2 + (6-3)^2\ )$=2.33

Finding Optimal θ_0, θ_1

Gradient Descent:



Regression Algorithms



Gradient Descent

Repeat until convergence {

$$\theta_0 := \theta_0 - \alpha \frac{\partial}{\partial \theta_0} J(\theta_0, \theta_1)$$

$$\theta_1 := \theta_1 - \alpha \frac{\partial}{\partial \theta_1} J(\theta_0, \theta_1)$$

}

$$\text{Temp1} := \theta_0 - \alpha \frac{\partial}{\partial \theta_0} J(\theta_0, \theta_1)$$

$$\text{Temp2} := \theta_1 - \alpha \frac{\partial}{\partial \theta_1} J(\theta_0, \theta_1)$$

$$\theta_0 := \text{Temp1}$$

$$\theta_1 := \text{Temp2}$$

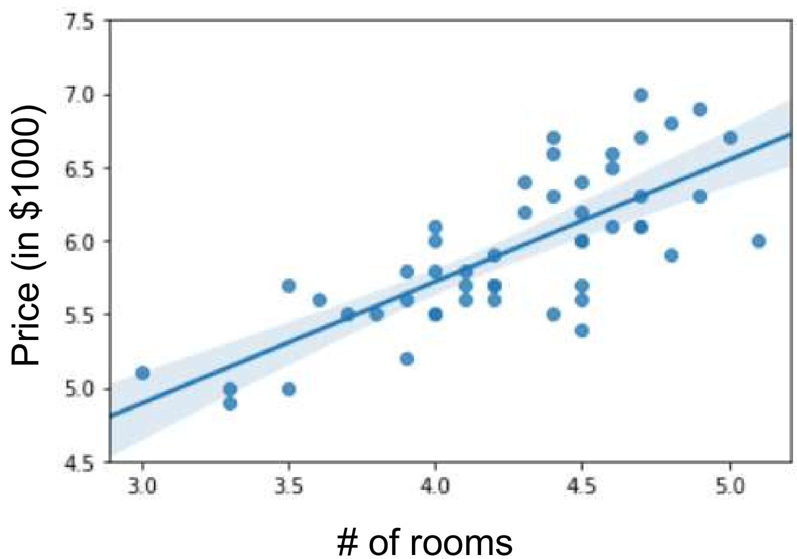
$$\text{Temp1} := \theta_0 - \alpha \frac{\partial}{\partial \theta_0} J(\theta_0, \theta_1)$$

$$\theta_0 := \text{Temp1}$$

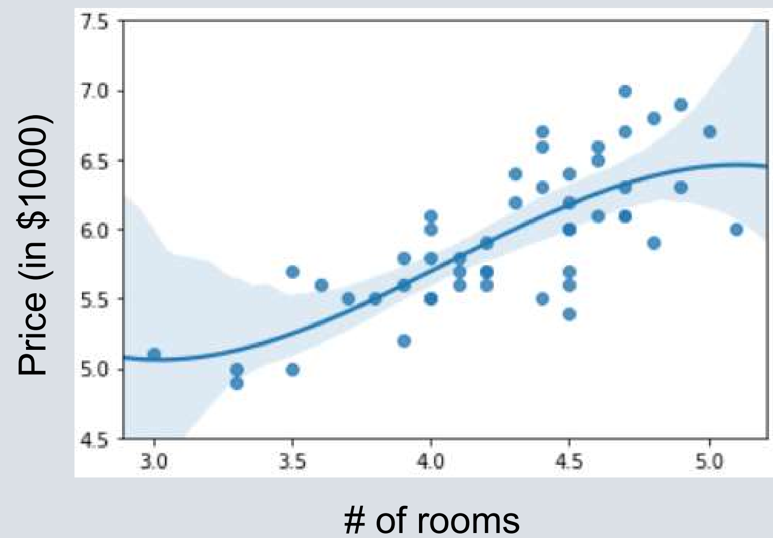
$$\text{Temp2} := \theta_1 - \alpha \frac{\partial}{\partial \theta_1} J(\theta_0, \theta_1)$$

$$\theta_1 := \text{Temp2}$$

Linear Regression



Non-Linear Regression

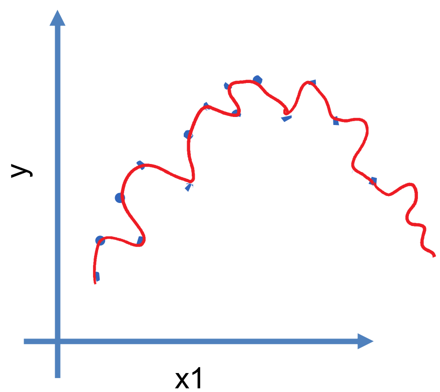


Training vs Testing

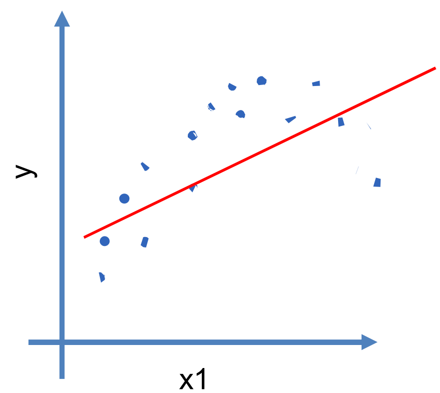
Why we should split the data to, Train set and Test set?

1- To make sure model will perform well on future data (not just historical data)

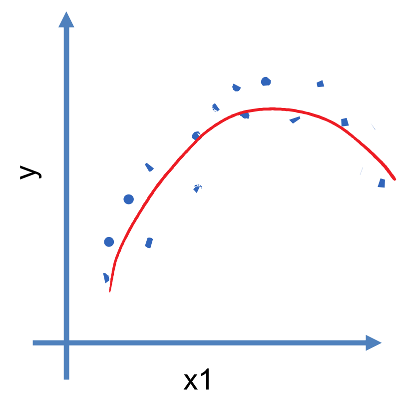
2- To make sure we are not **over-fitting** or **under-fitting** data



Over-fitted



Under-fitted



Just right!