Machine Learning crash course

(Part-2)

Ahmed Hammad

Introduction to machine learning, linear and non-linear regression models

Introduction to ML

Artificial intelligence

A program that can sense, act and adopt with the user

Machine learning

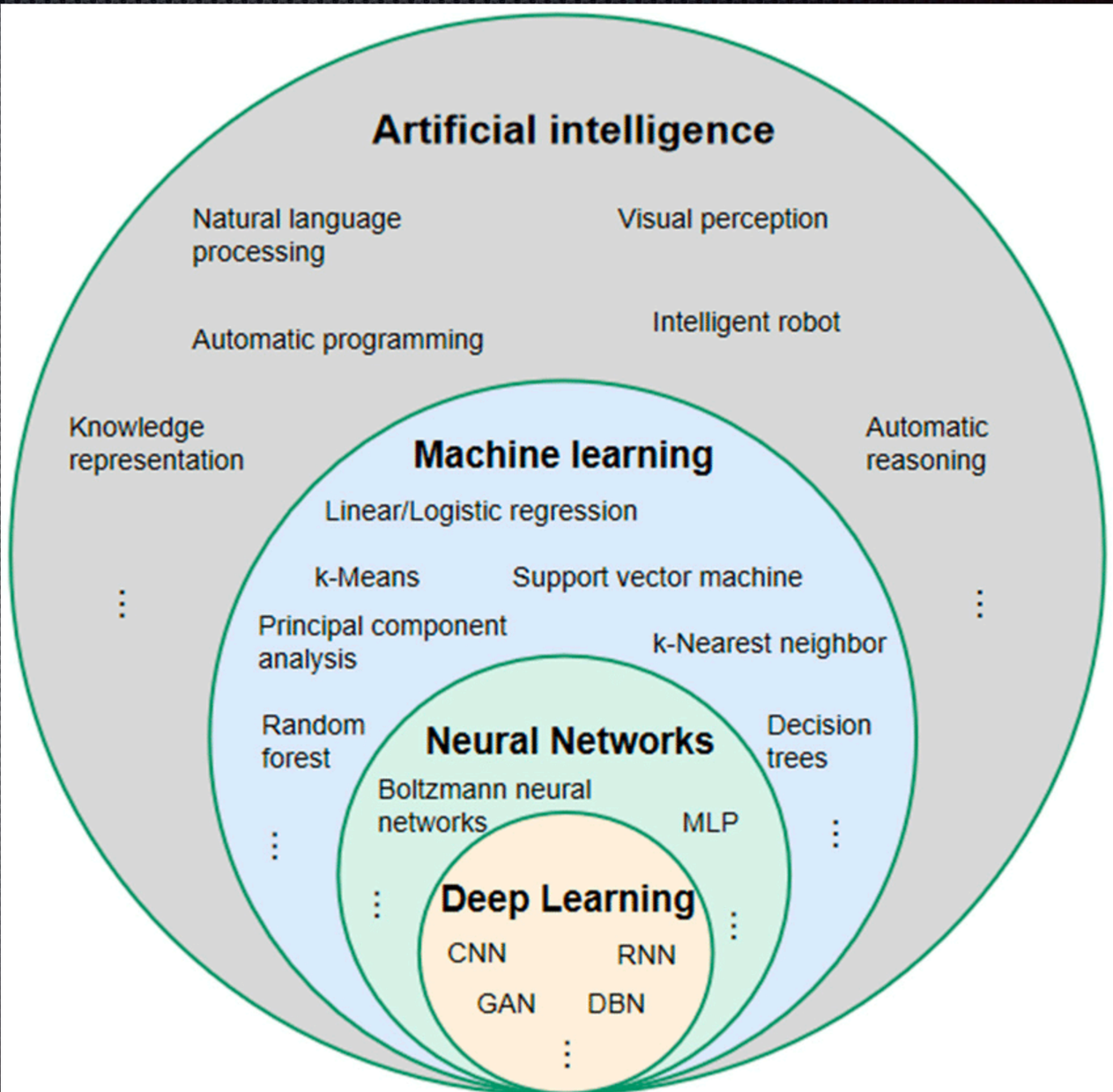
Algorithms whose performance improves as they are expose to more data over time

Deep learning

A subset of machine learning algorithms in which the algorithms learns deeply hidden features into the data

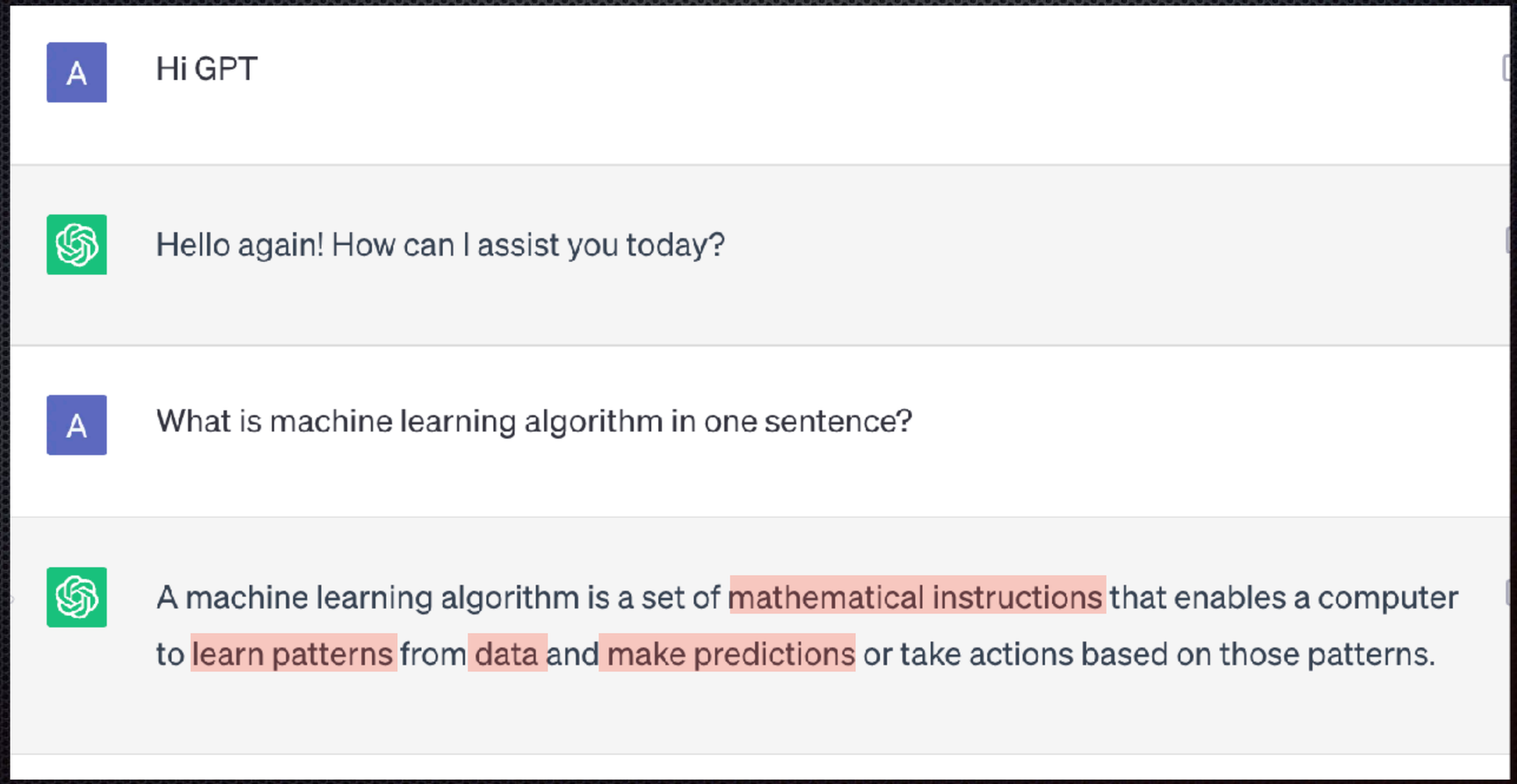
Taken from

https://www.researchgate.net/figure/Relationship-between-artificial-intelligence-machine-learning-neural-network-and-deep_fig3_354124420



What is machine learning ?

Just remind you with ChatGPT answer



Introduction to ML

Machine Learning Algorithms consist of two main parts: Training and Test

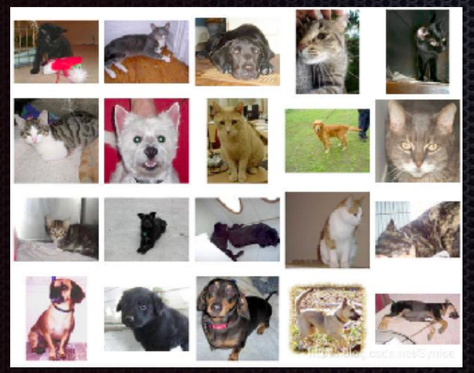
Just like a baby in kindergarten

1+1=2
1+2=3
.
.
.
1+9=10

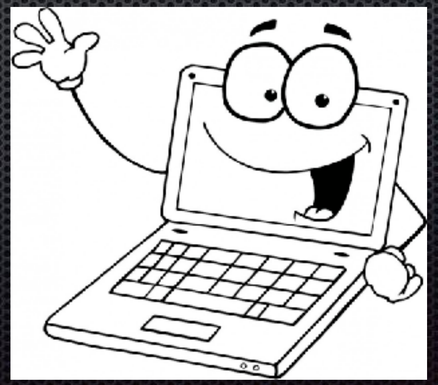
Learning

Test

2+5=?



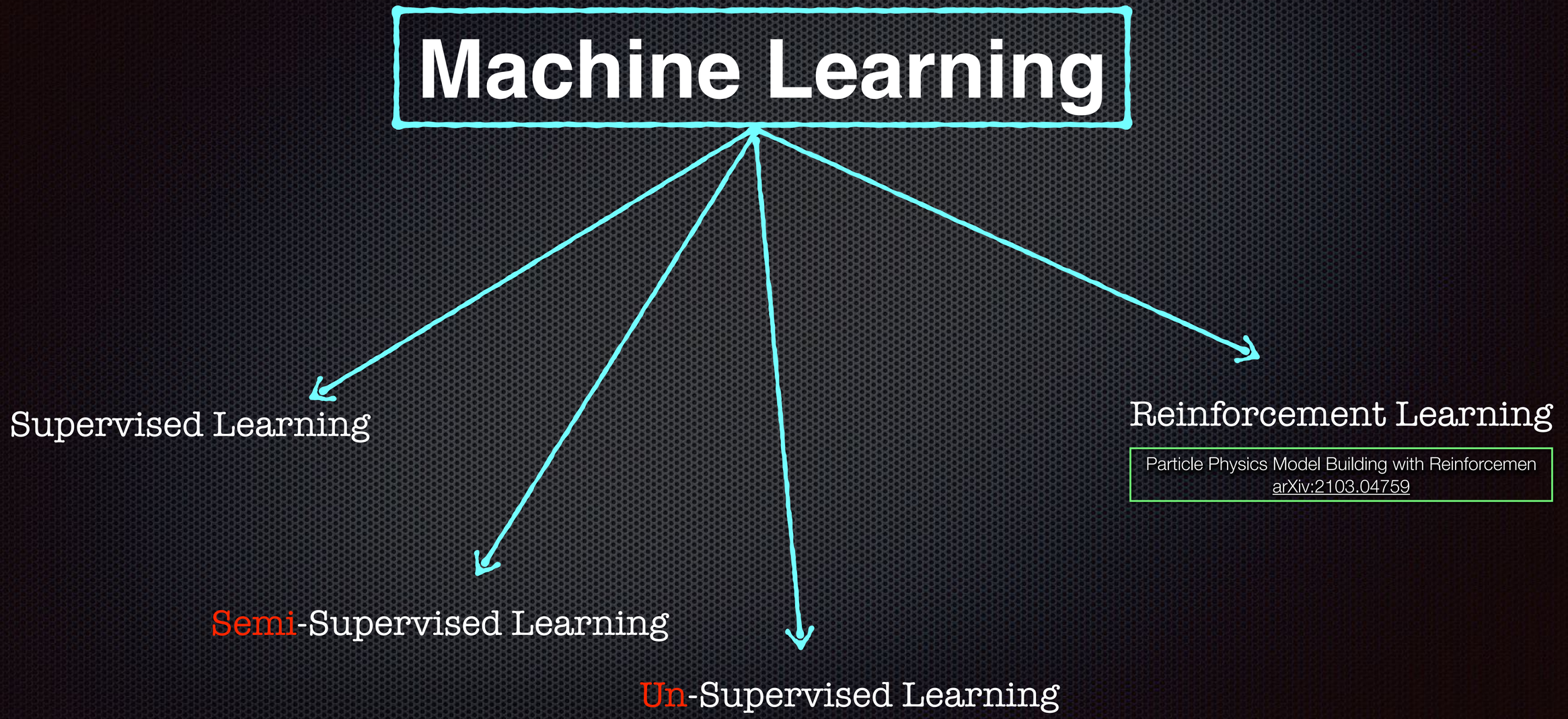
Learning



Test

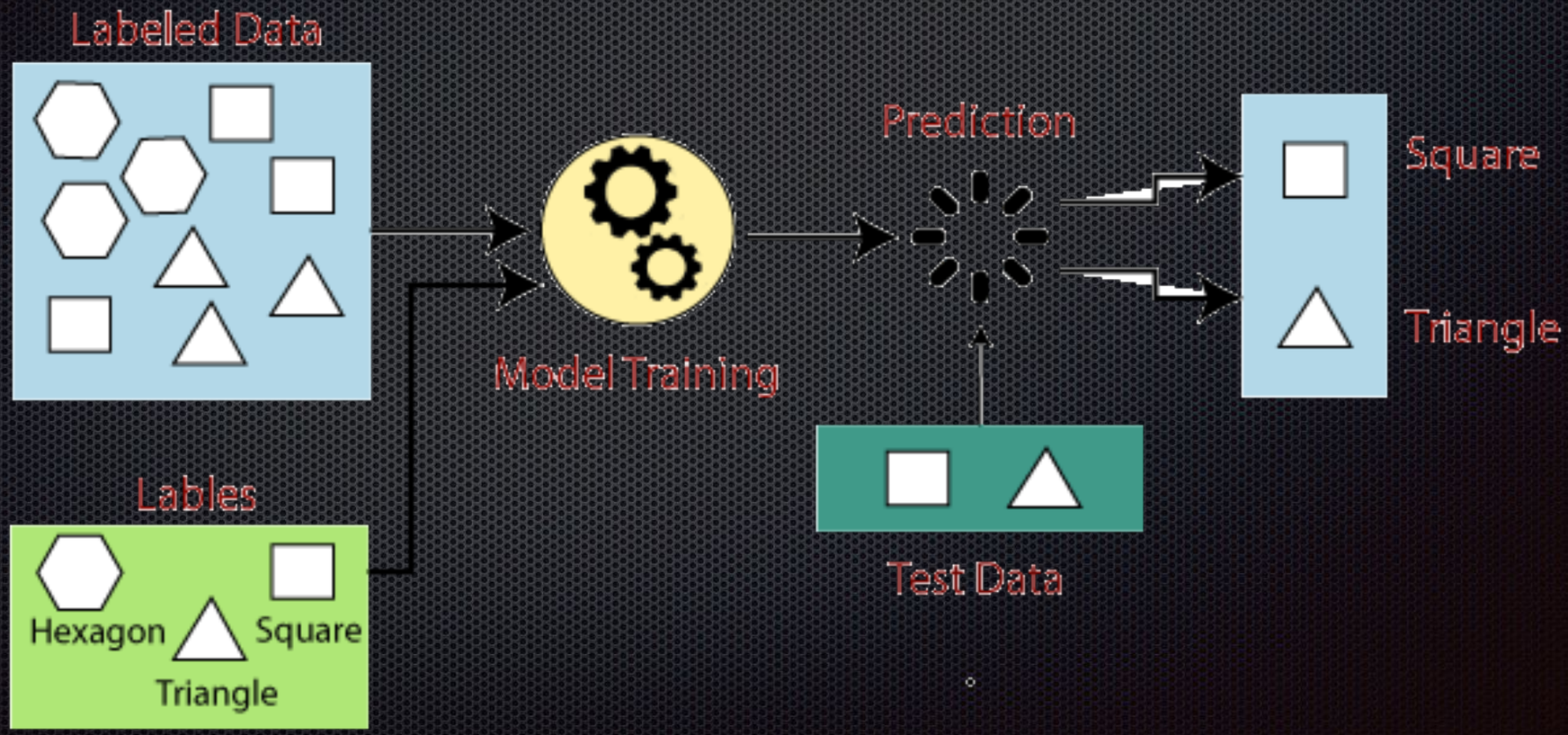
?

Introduction to ML



Supervised Learning

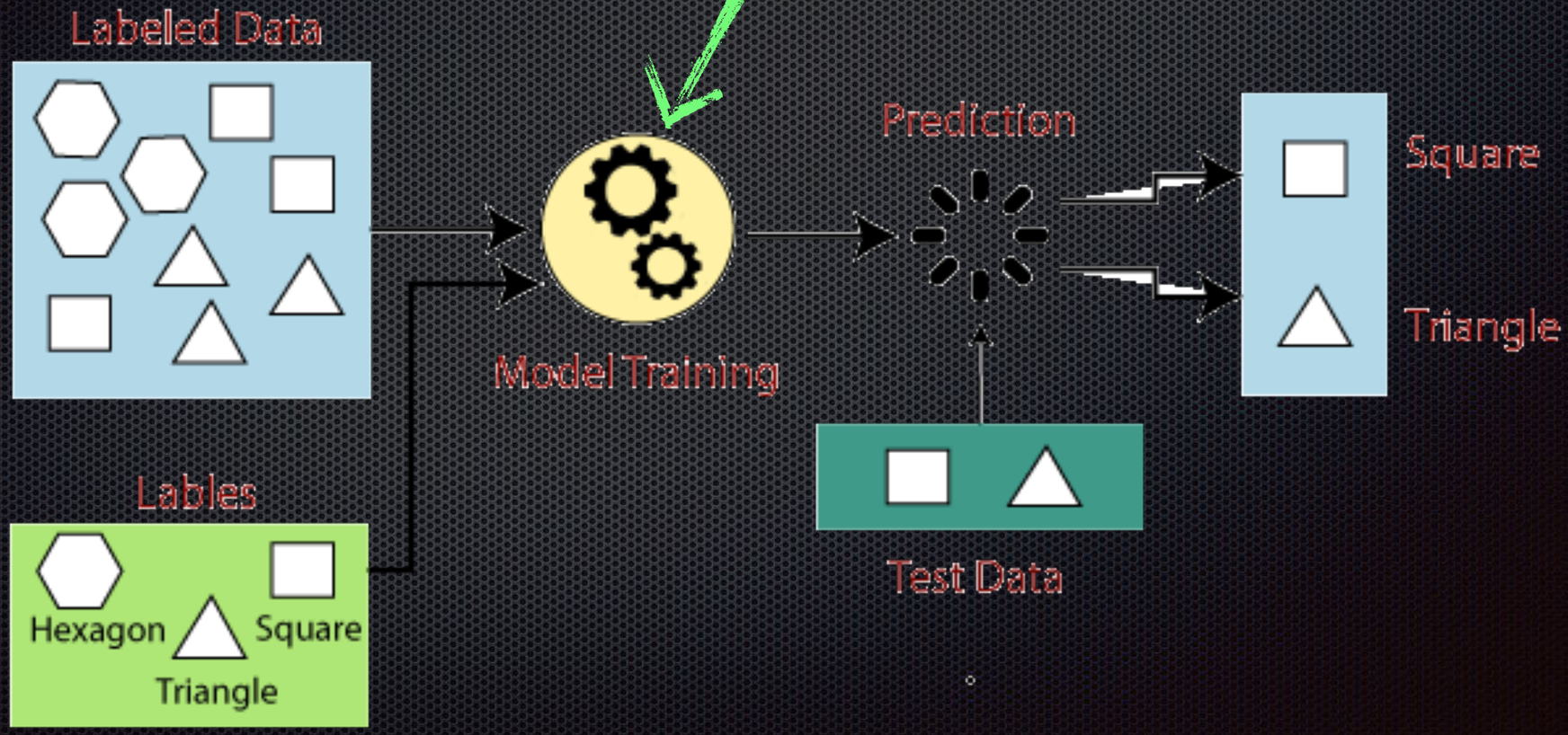
Modeling the relationship between measured features of data and some label associated with the data; once this model is determined, it can be used to apply labels to new, unknown data.



Supervised Learning

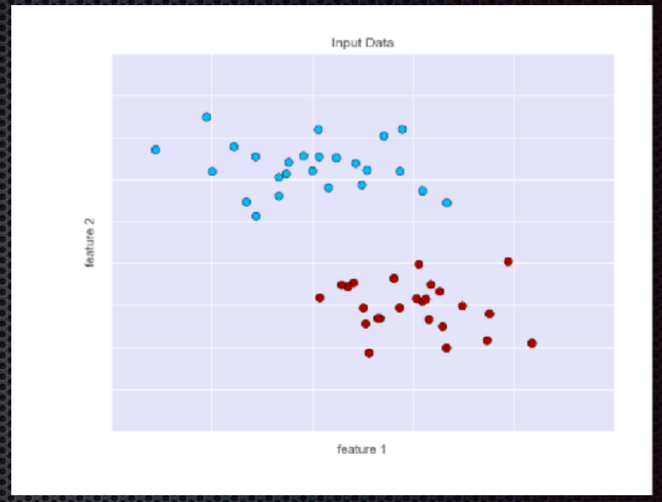
Modelin
label as

For the Supervised learning the complexity is how to train your model to find the best classifier for the given data

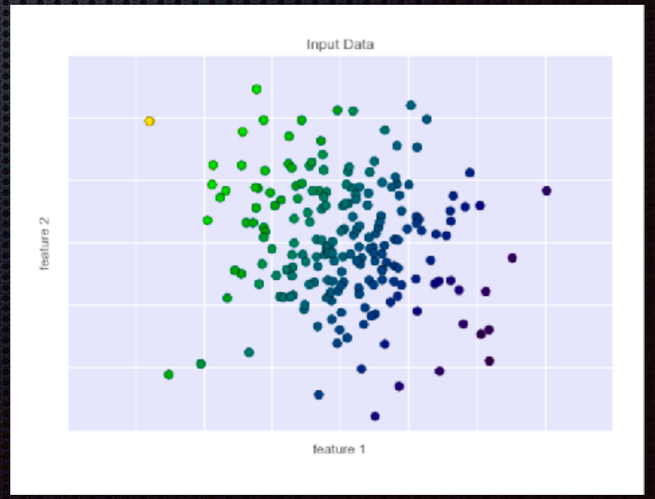


Supervised Learning

Classification : For Binary (colored) data set



Regression : For Continuous (colored) data set

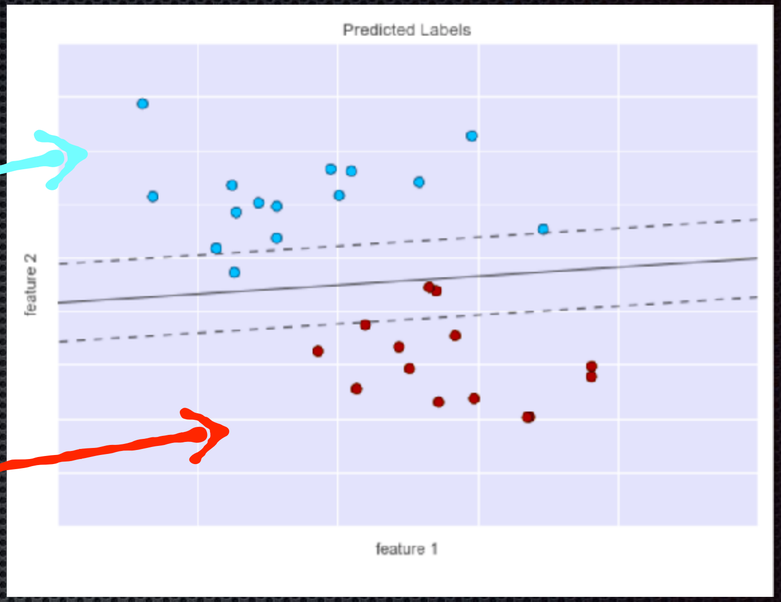


Supervised Learning

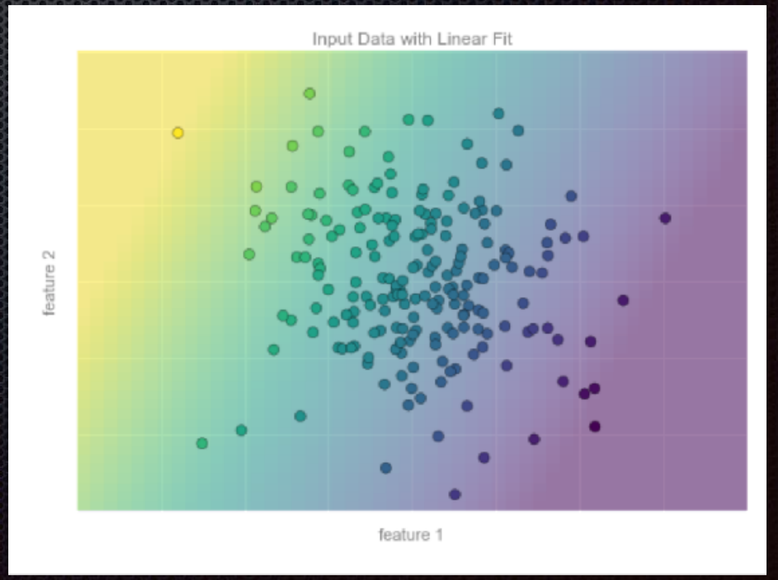
Blue points = 0

Linear classification

Red points = 1

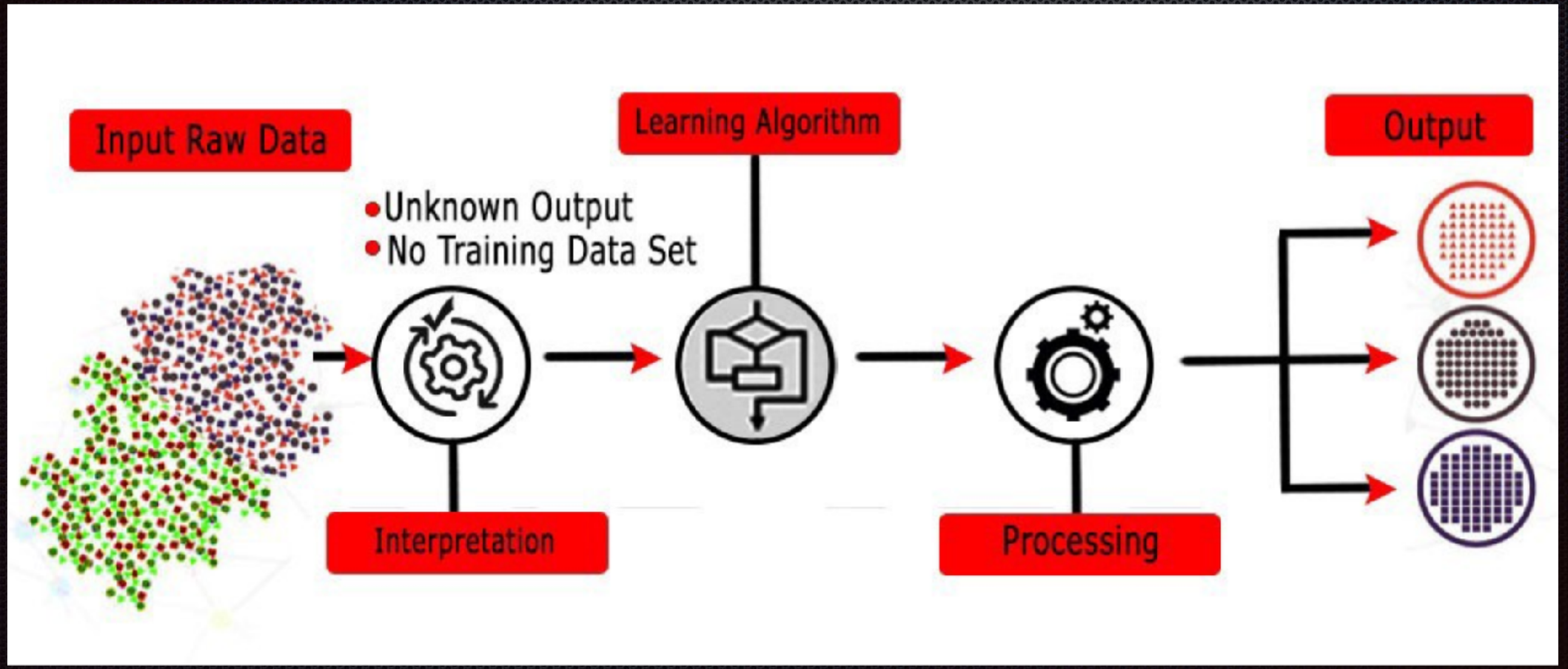


Classifier: continuous color spectrum



Unsupervised Learning

Modeling the features of a dataset without reference to any label, and is often described as "letting the dataset speaks for itself."

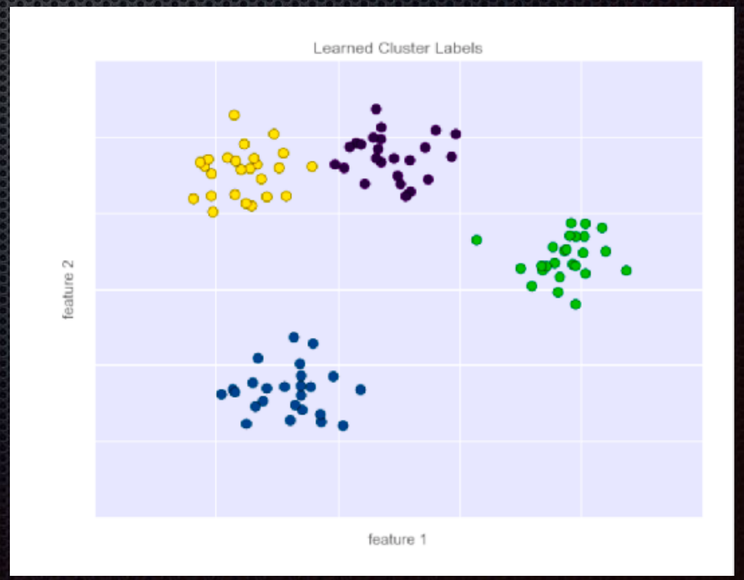


Unsupervised Learning

Unsupervised Learning ⟷ No labels

Classification problem

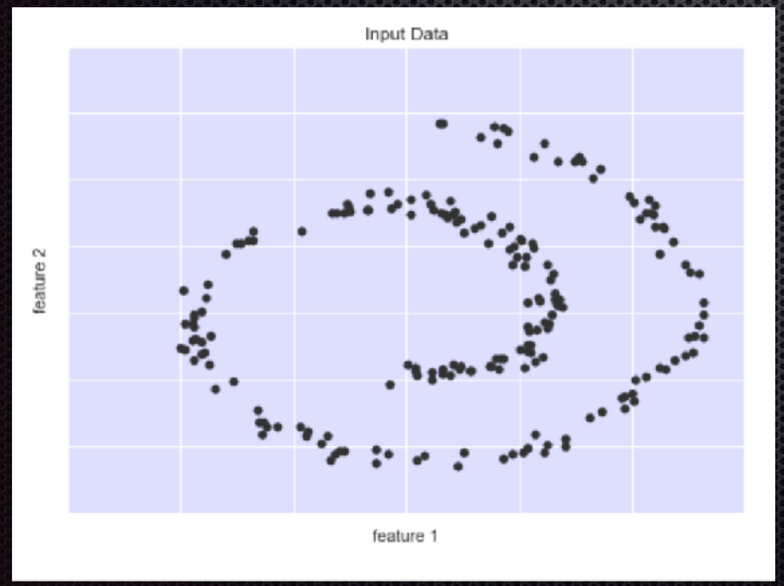
Clustering



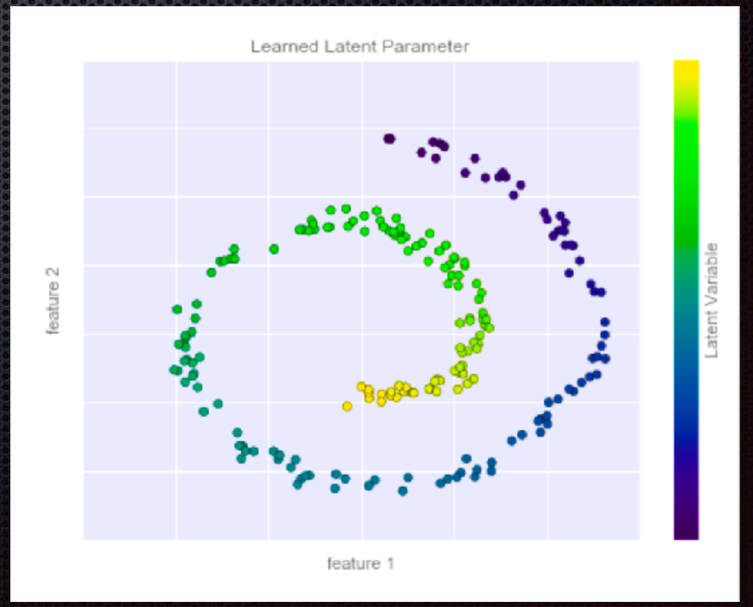
Unsupervised Learning

Unsupervised Learning ⟷ No labels

Regression problem

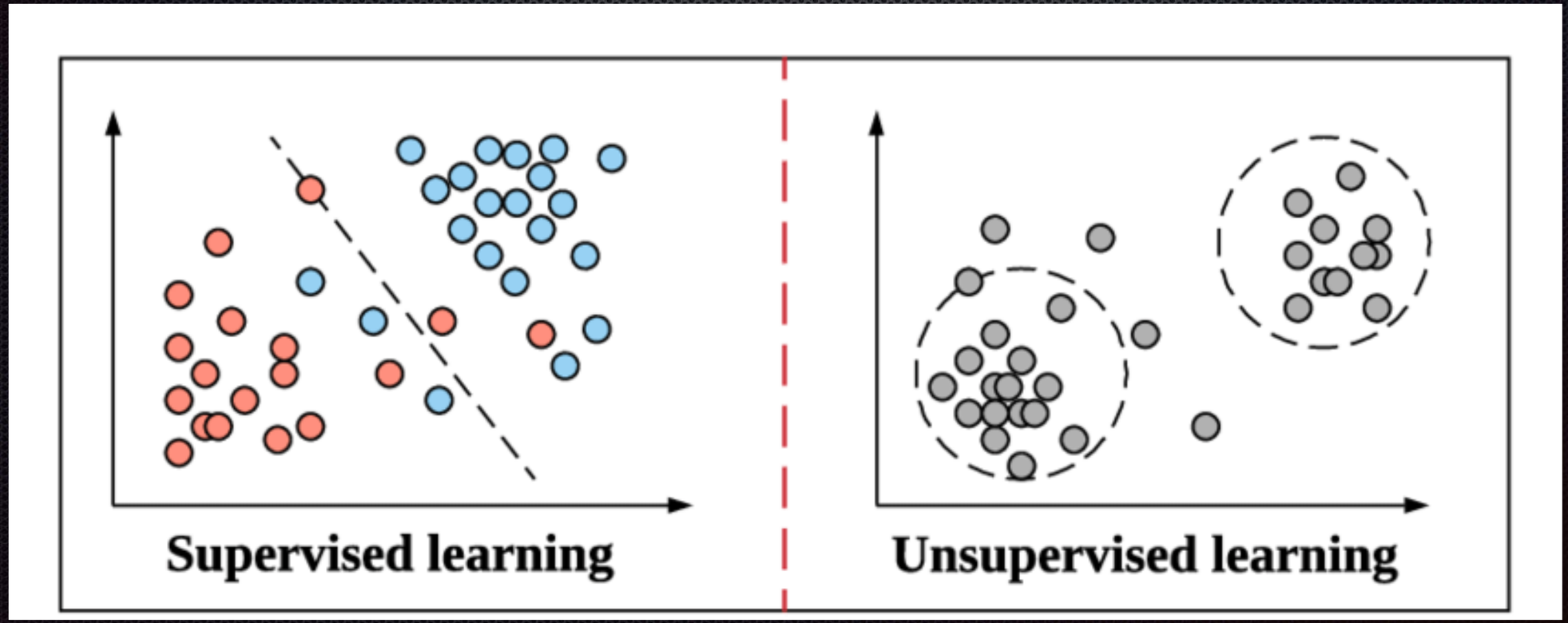


Clustering



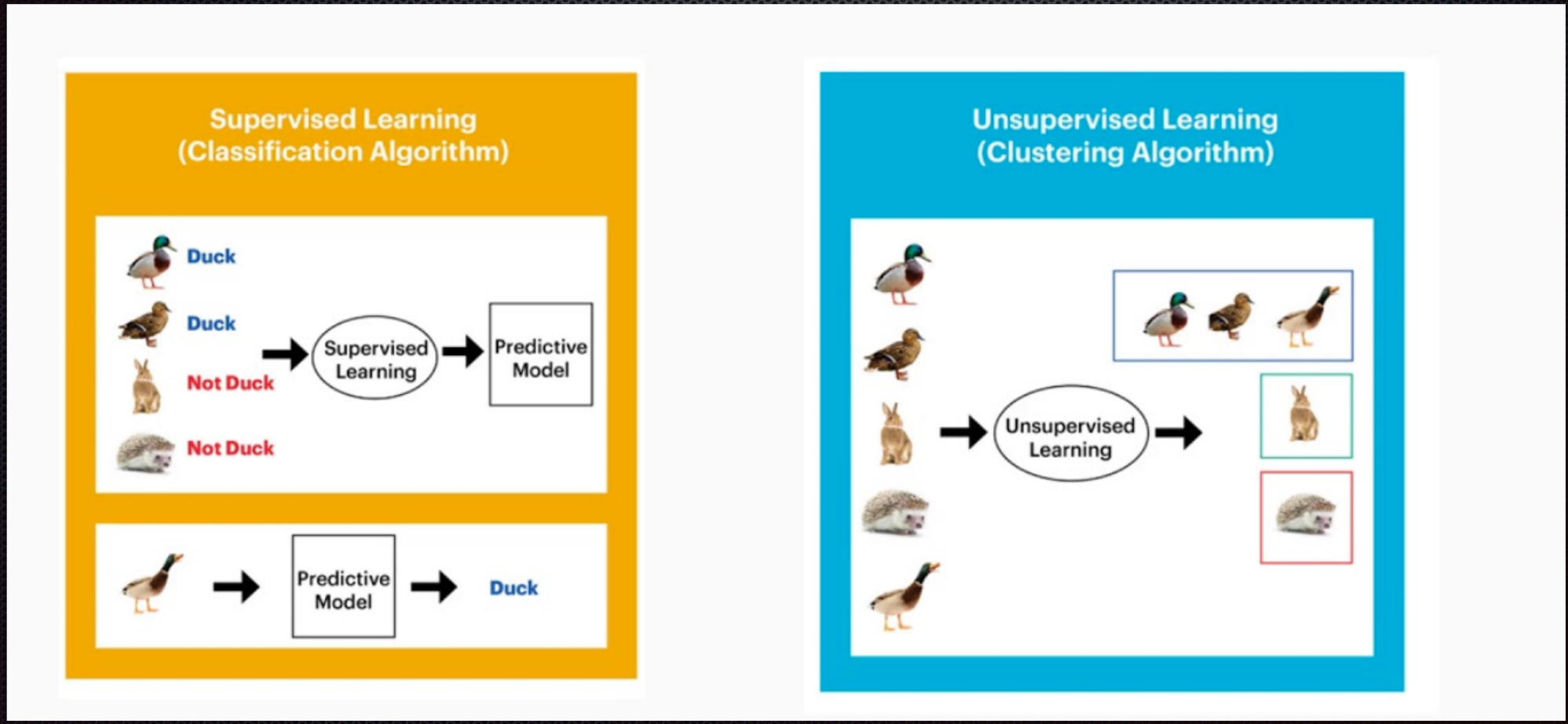
Examples

Supervised Learning vs Unsupervised Learning



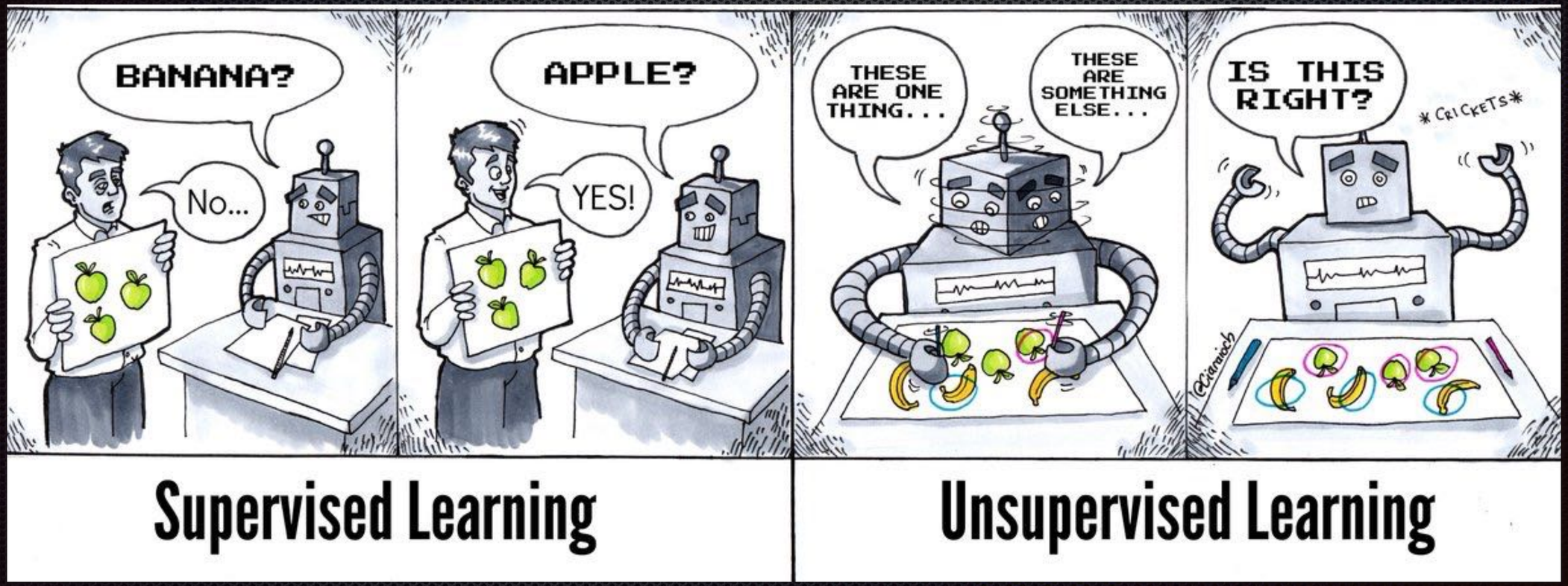
Examples

Supervised Learning vs Unsupervised Learning

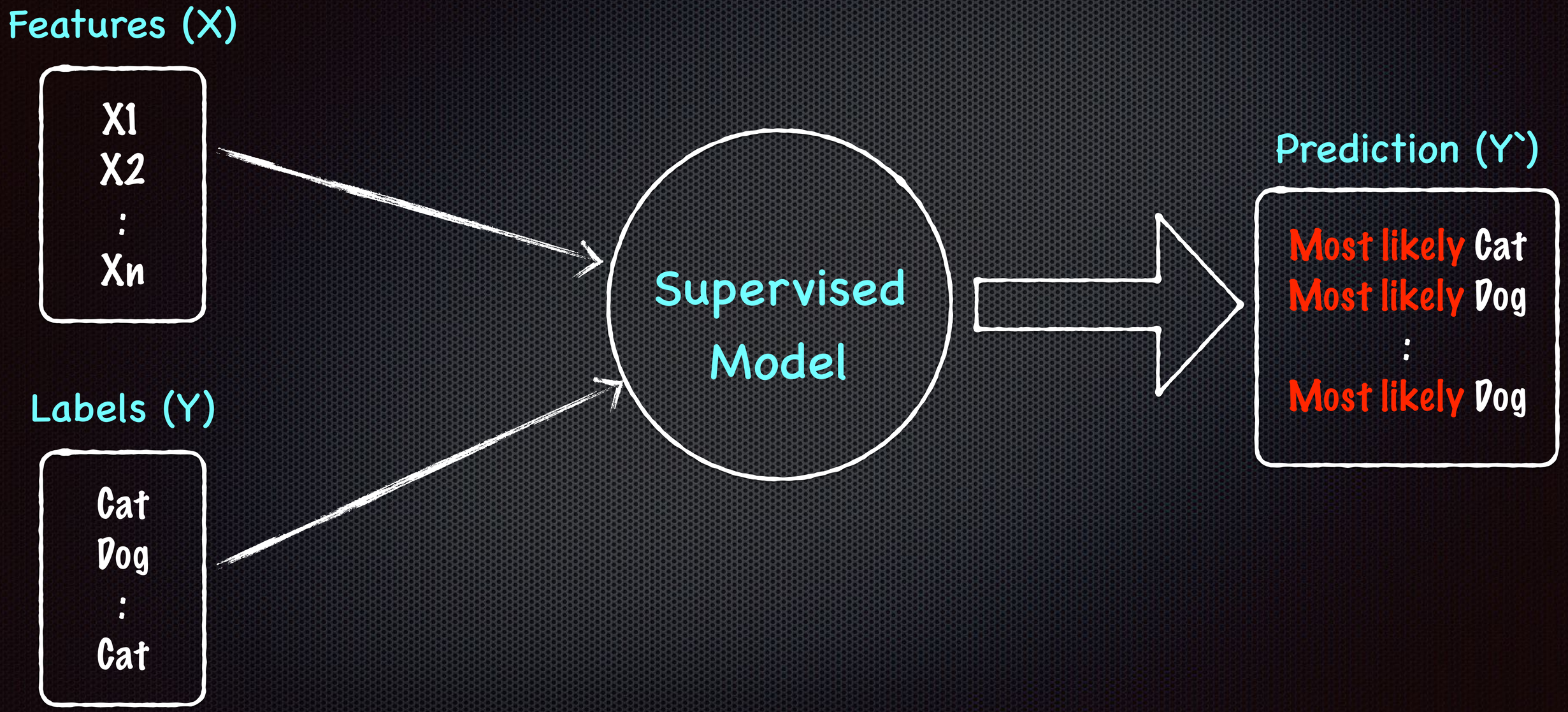


Examples

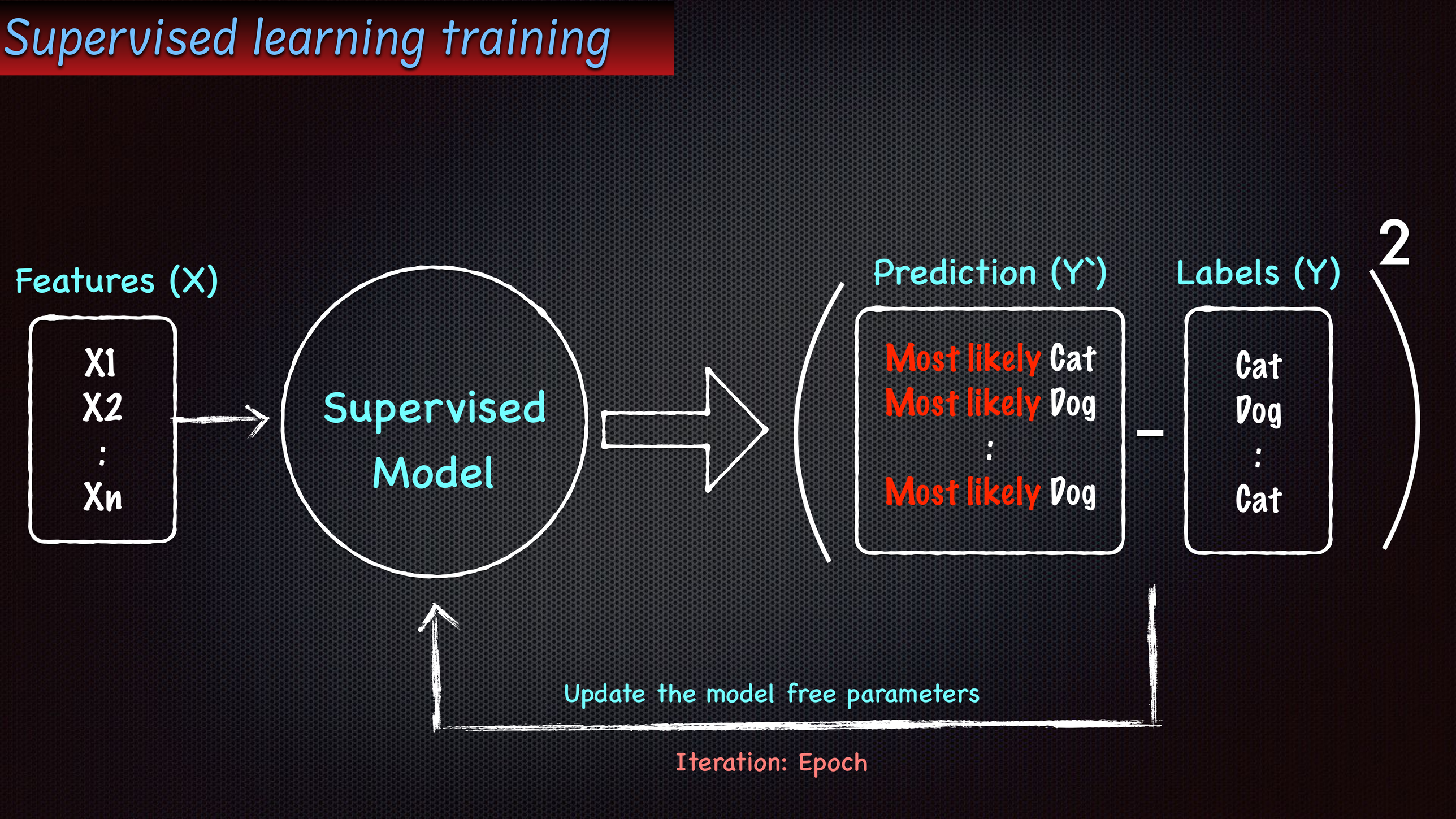
Supervised Learning vs Unsupervised Learning



Supervised learning training

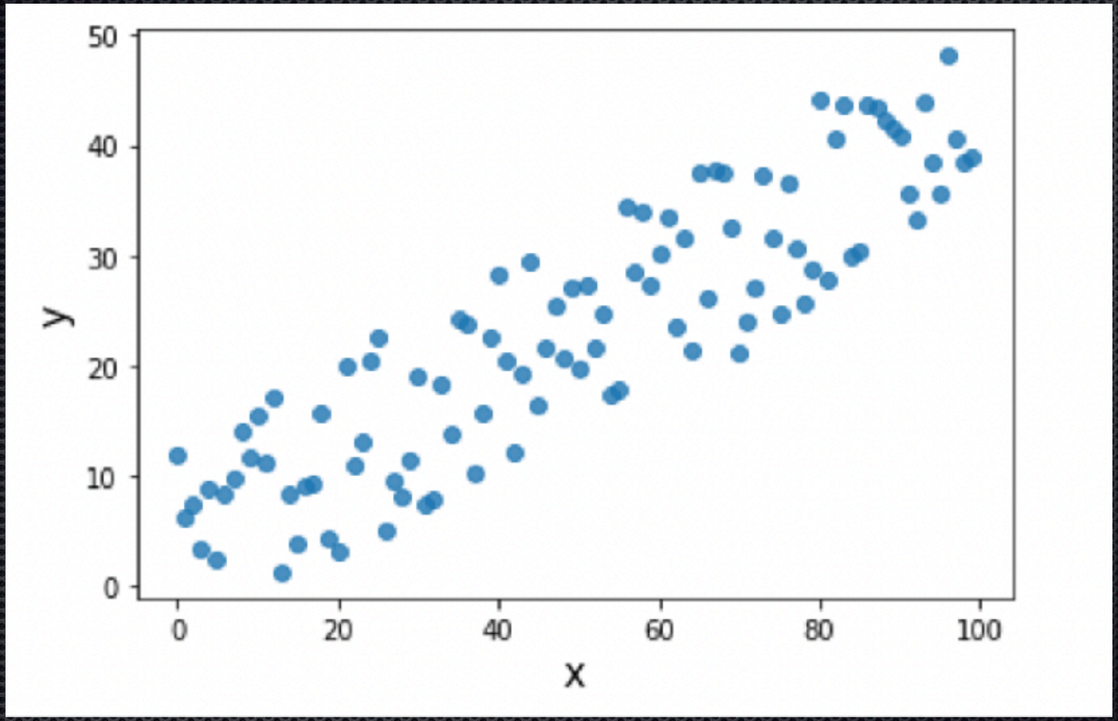


Supervised learning training



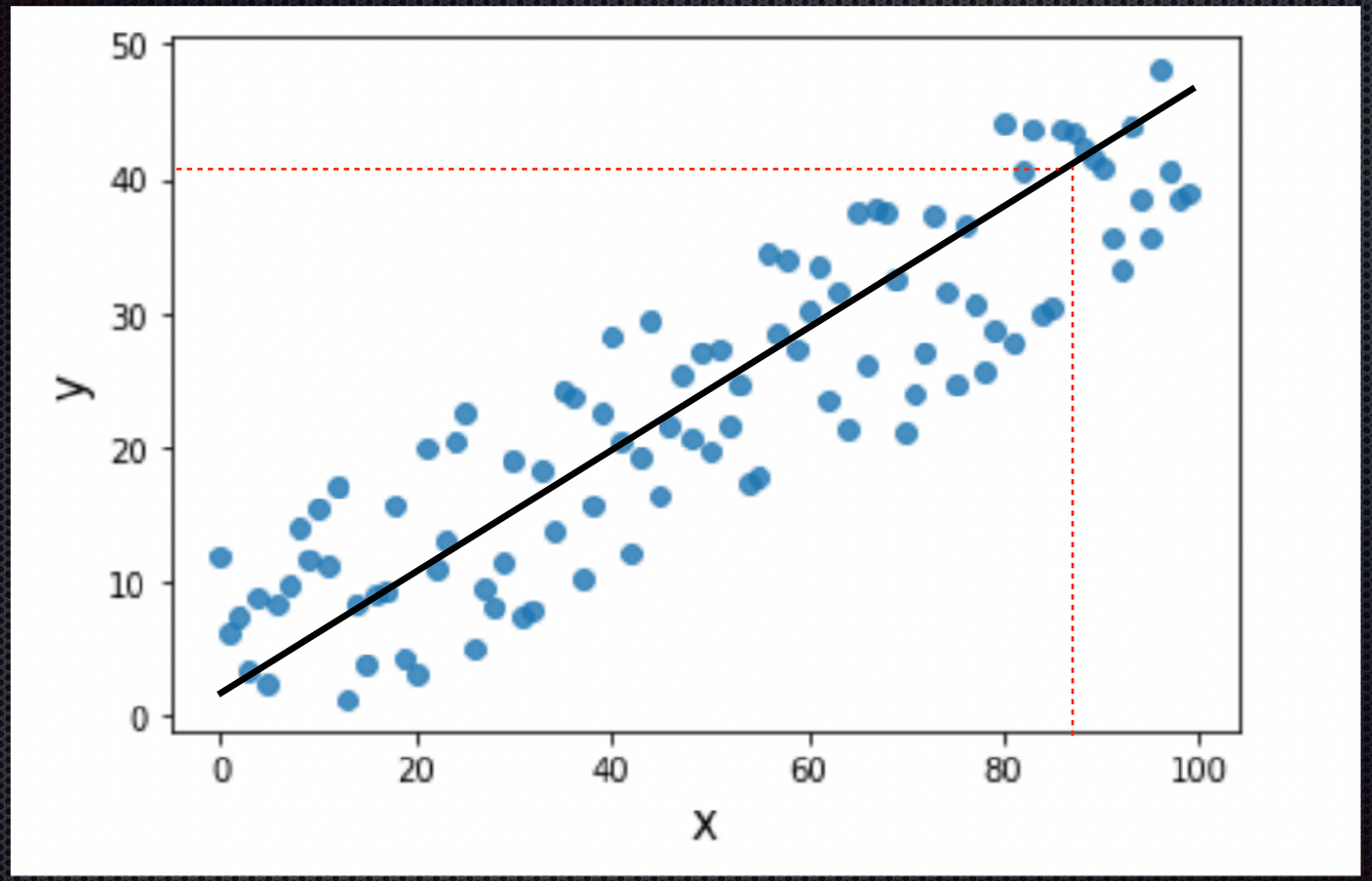
Linear regression

For a continuous data set X



Given new (unseen) x point, can you predict the corresponding Y ?

Linear regression

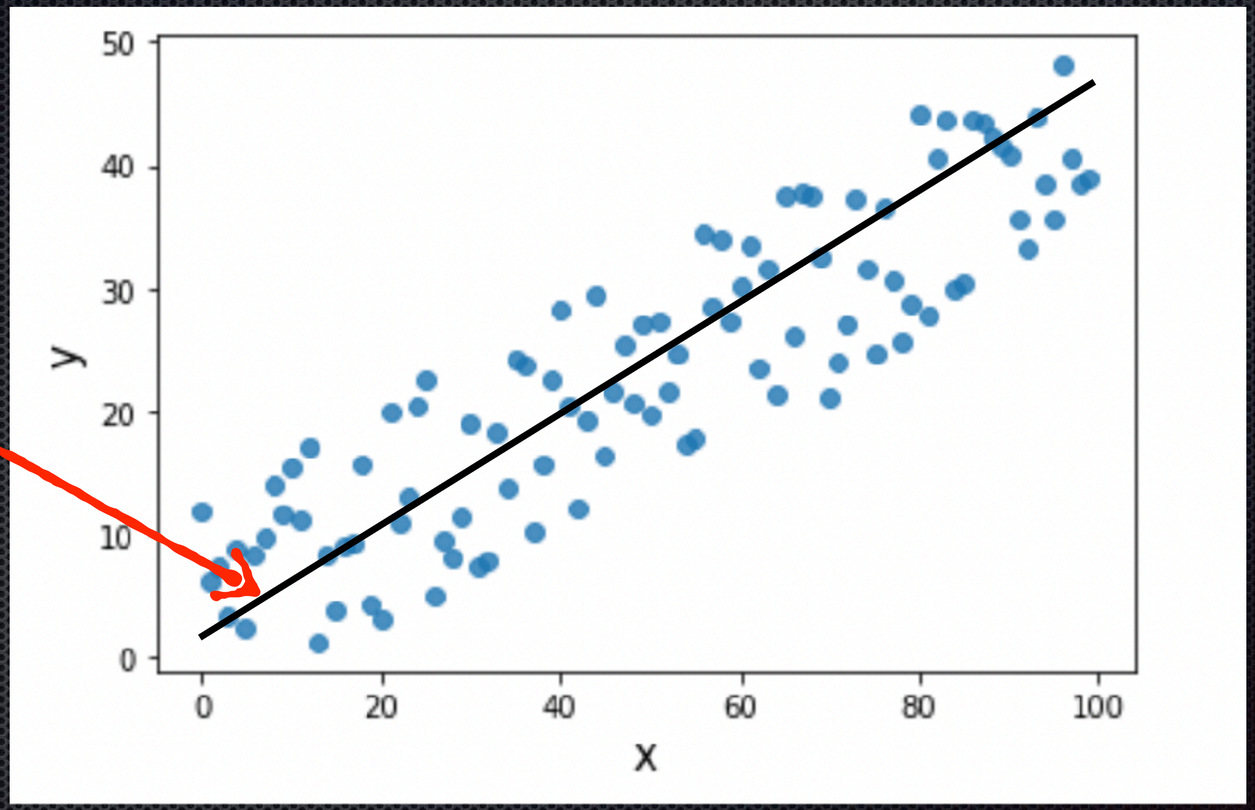


Once we found the line that best fit the data we can interpret the Y for new x points

Linear regression

How to find the line that fit the given data ?

$$\hat{Y} = B0 + B1x$$



With given Y and x we try to find B0 and B1 that fit the line to the data

Linear regression

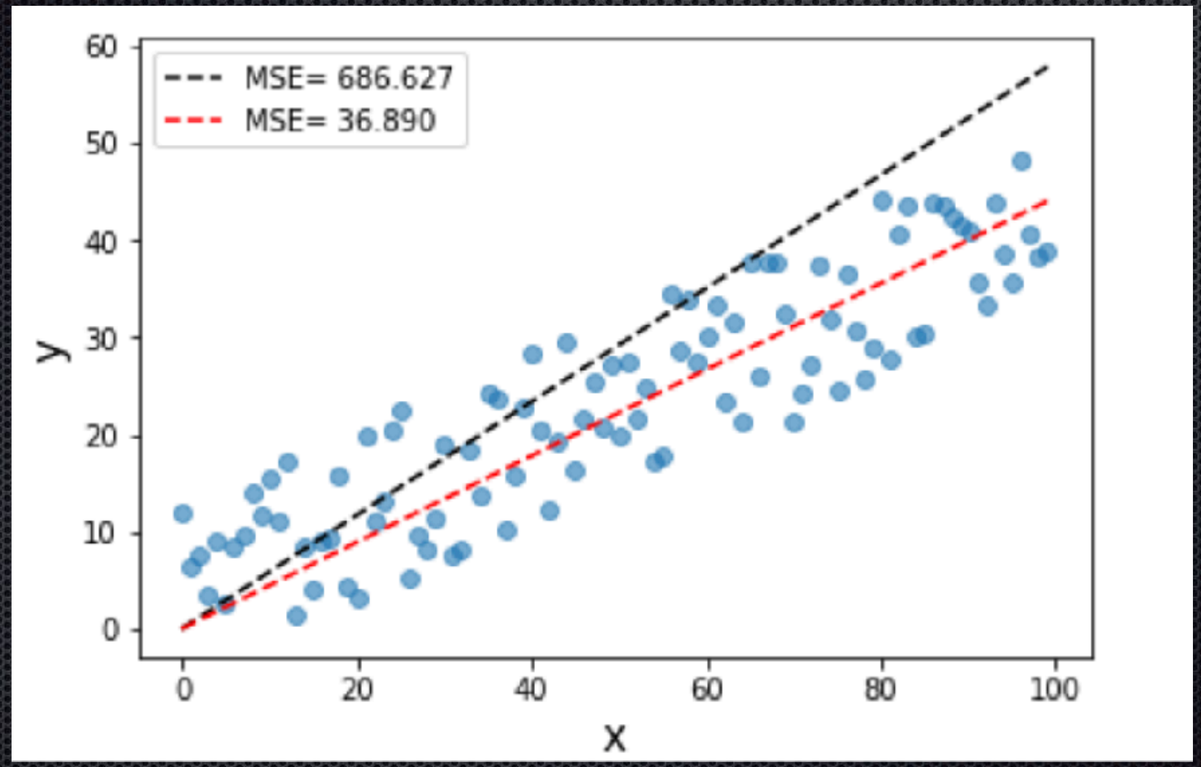
Question:

We can draw infinite number of lines that can fit the data!!
Which line shall we consider ?

Linear regression

Question:

We can draw infinite number of lines that can fit the data!!
Which line shall we consider ?

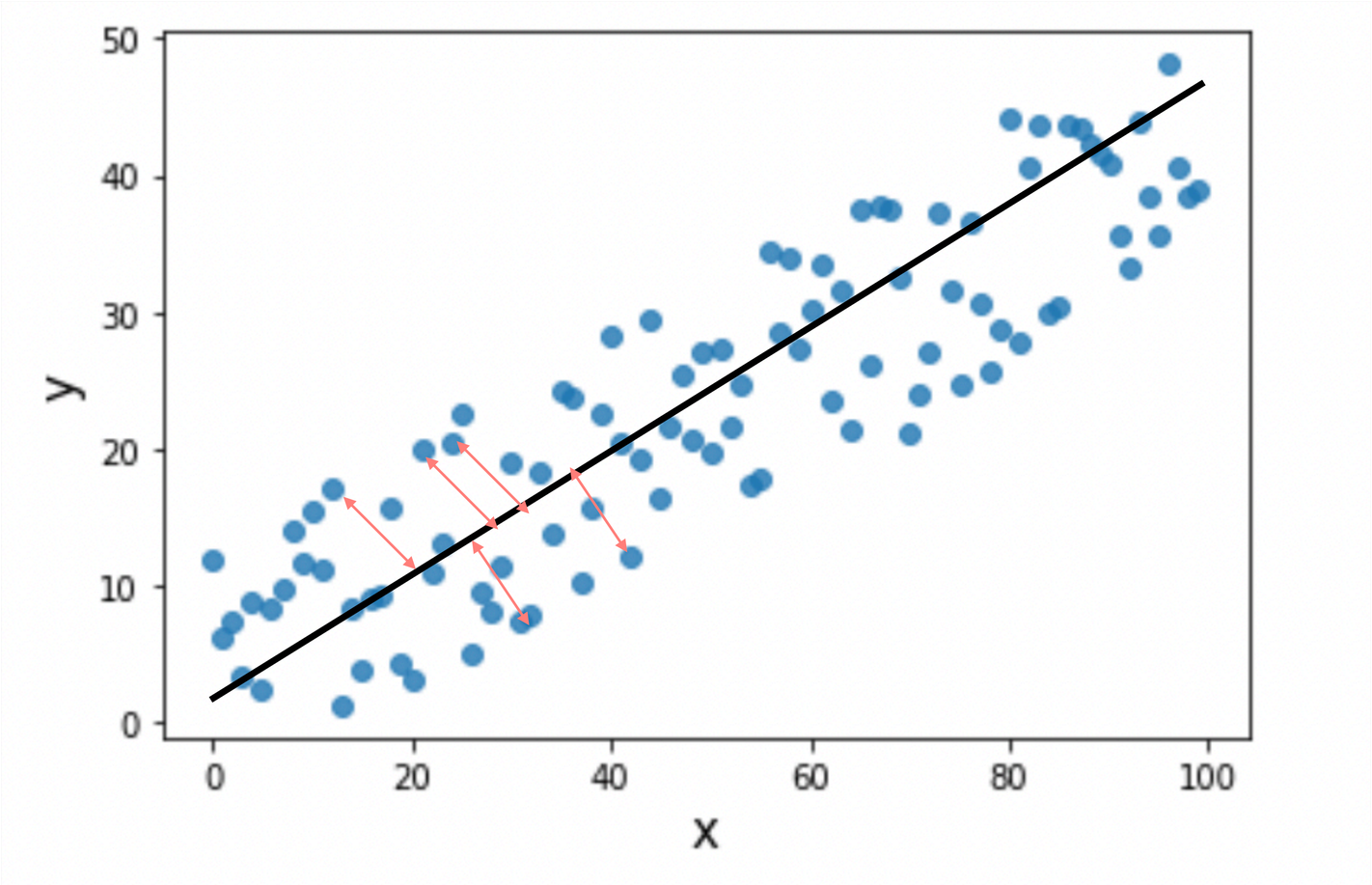


We have to consider the line that minimizes the error function

Linear regression-loss function

Loss function:

It mostly quantifies the difference between the model predictions and the true values



Linear regression-loss function

Types of linear regression loss functions

Absolute mean error:

$$\mathrm{AME} = \frac{1}{m}\sum^{m}\left|Y - \hat{Y}\right| = \frac{1}{m}\sum^{m}|Y - (B0 + B1x)|$$

Mean square error:

$$\mathrm{MSE} = \frac{1}{m}\sum^{m}\left|Y - \hat{Y}\right|^2 = \frac{1}{m}\sum^{m}|Y - (B0 + B1x)|^2$$

Root mean square error:

$$\mathrm{RMSE} = \sqrt{\frac{1}{m}\sum^{m}\left|Y - \hat{Y}\right|^2} = \sqrt{\frac{1}{m}\sum^{m}|Y - (B0 + B1x)|^2}$$

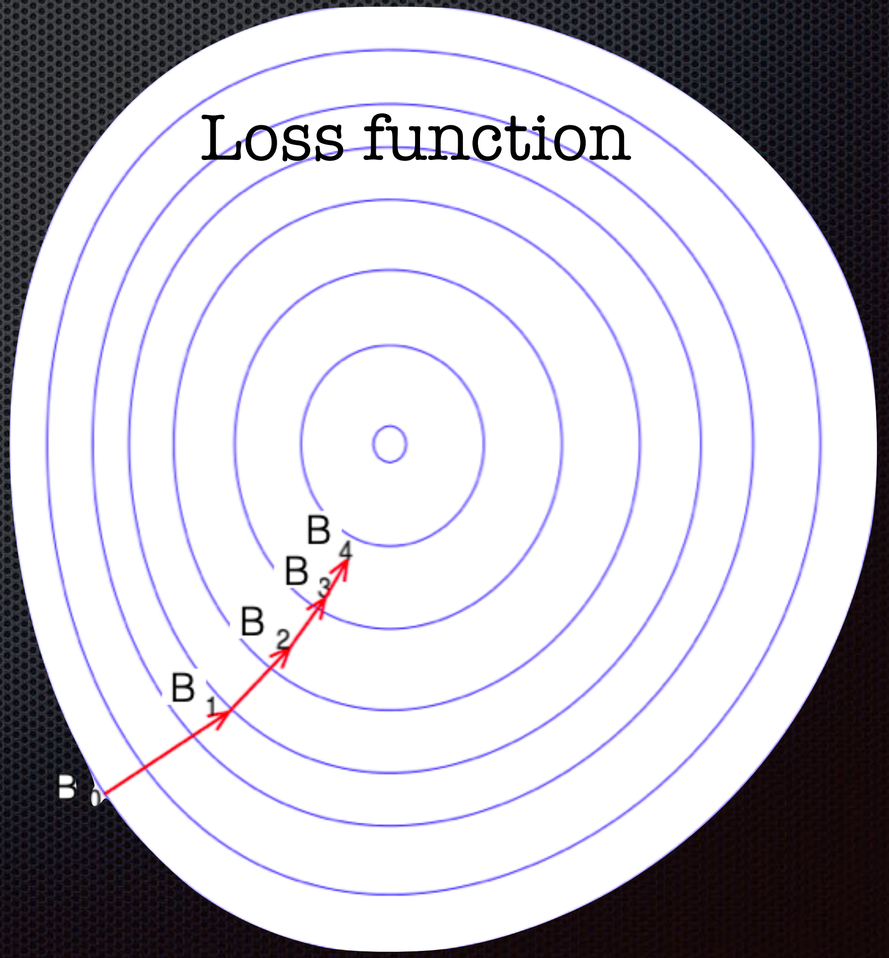
Gradient Descent method

Question:

Well, now we know that to find the best fit line we need to minimize the loss function, but how ??

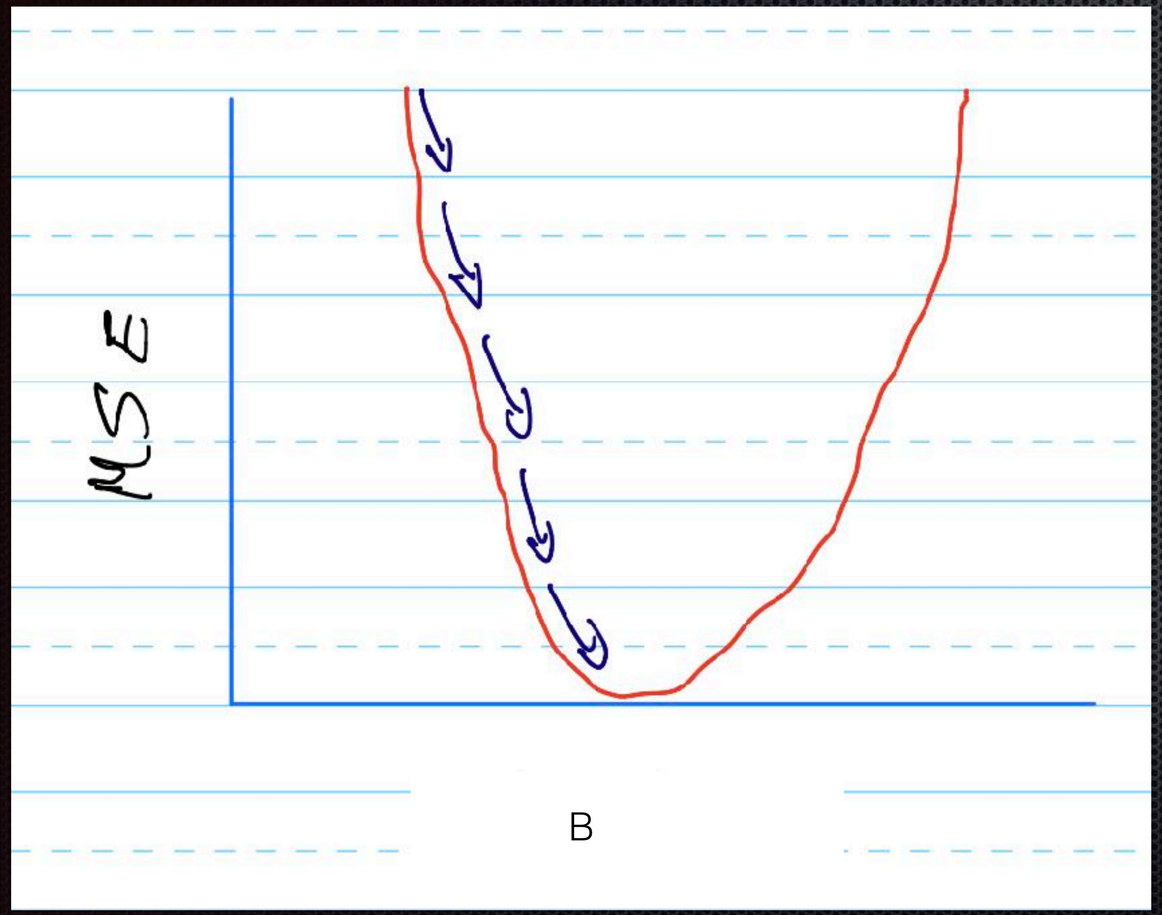
Find the best fit parameters using the gradient descent method

$$B^i_{\text{new}} = B^i_{\text{old}} - \eta \nabla \text{Loss}(B^i_{\text{old}})$$



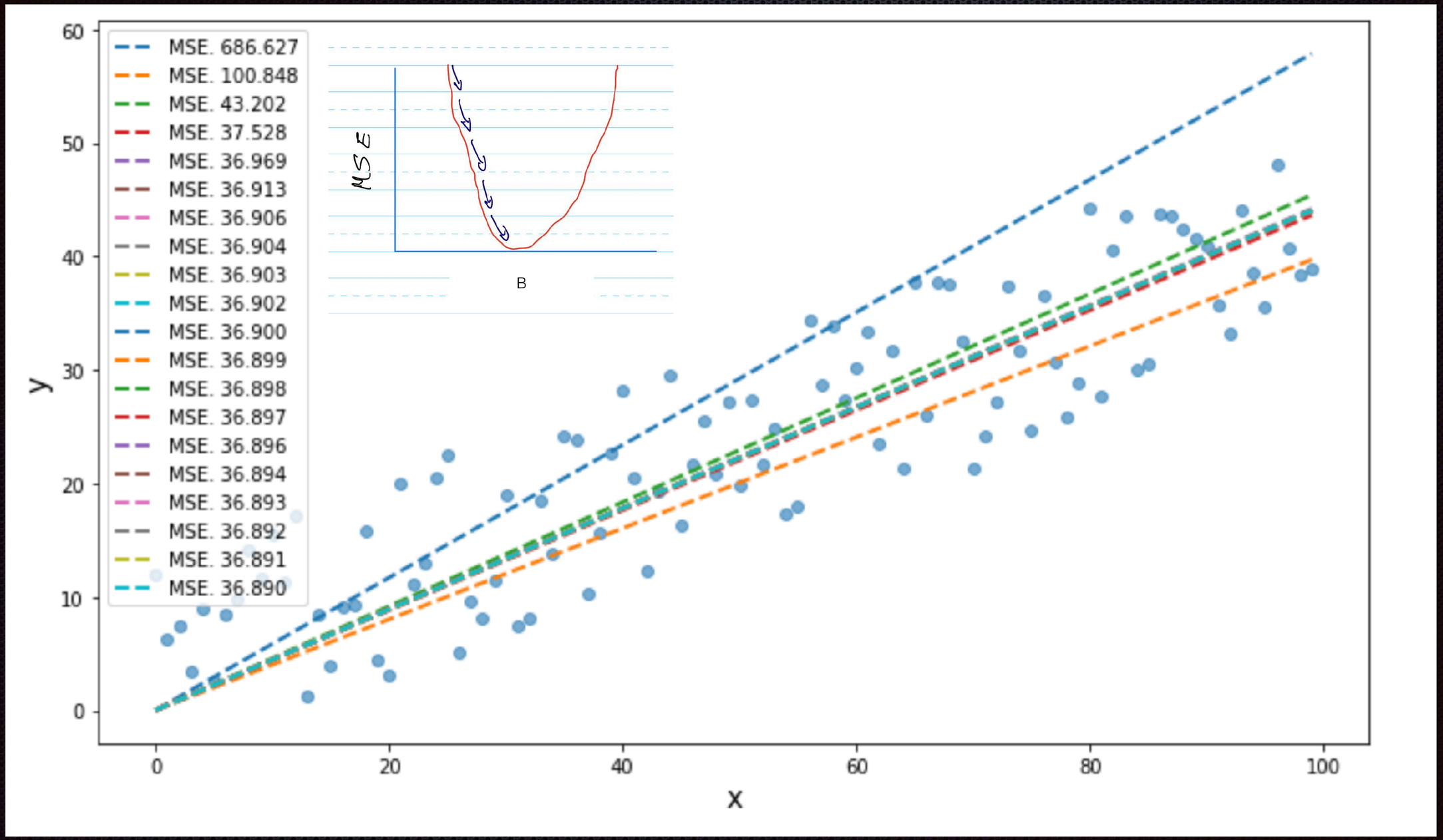
Gradient Descent method

Given a number of iteration, every iteration we update the parameters and calculate the error function until we hit the global minimum



$$B^i_{\text{new}} = B^i_{\text{old}} - \eta \nabla \text{MSE}(B^i_{\text{old}})$$

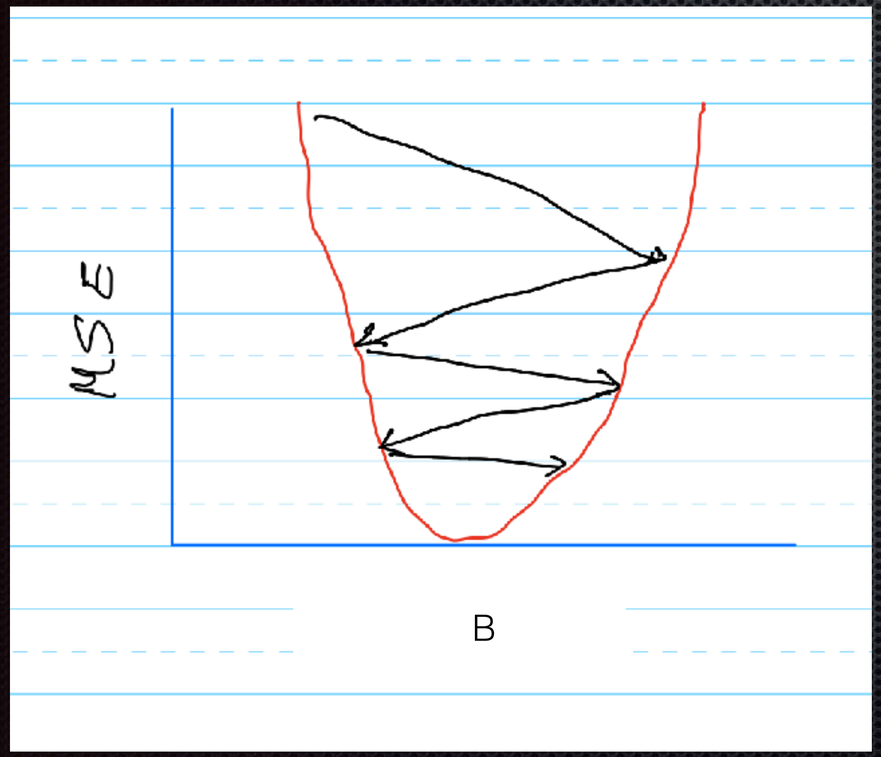
Gradient Descent method



Learning rate

$$B^i_{\text{new}} = B^i_{\text{old}} - \eta \nabla \text{Loss}(B^i_{\text{old}})$$

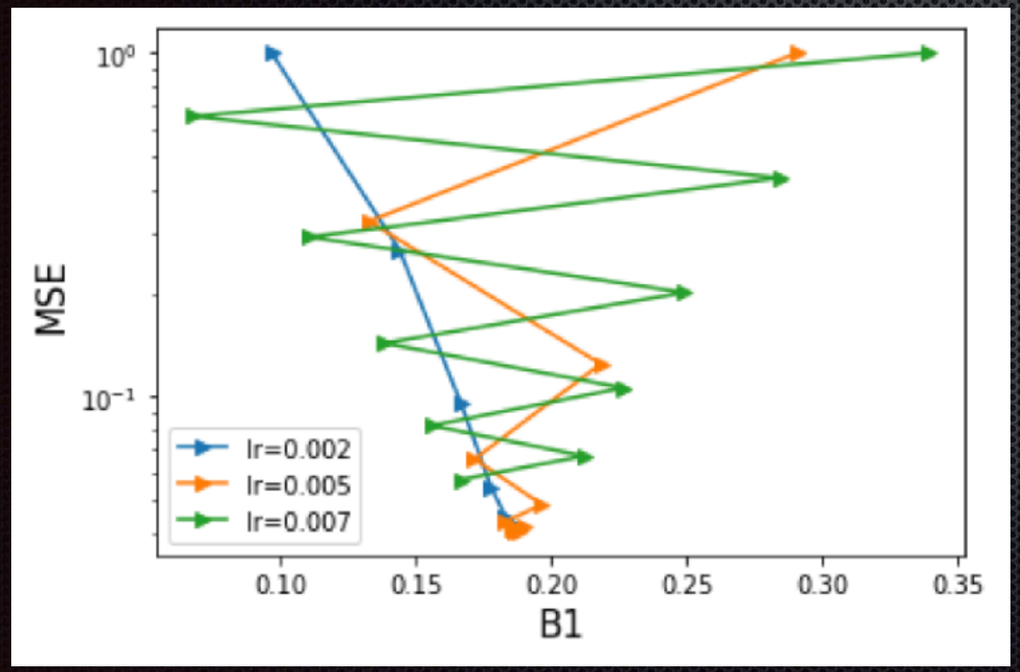
η Is called the learning rate which controls the descent rate of the loss function

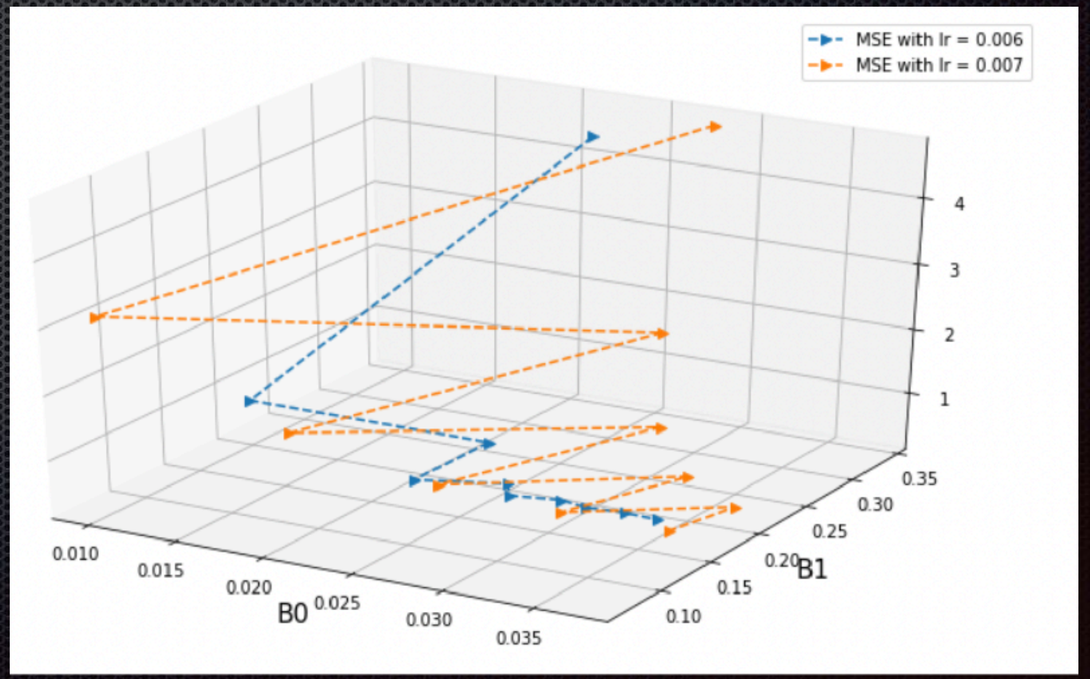


For large learning rate the function oscillates and we will not able to reach the minimum

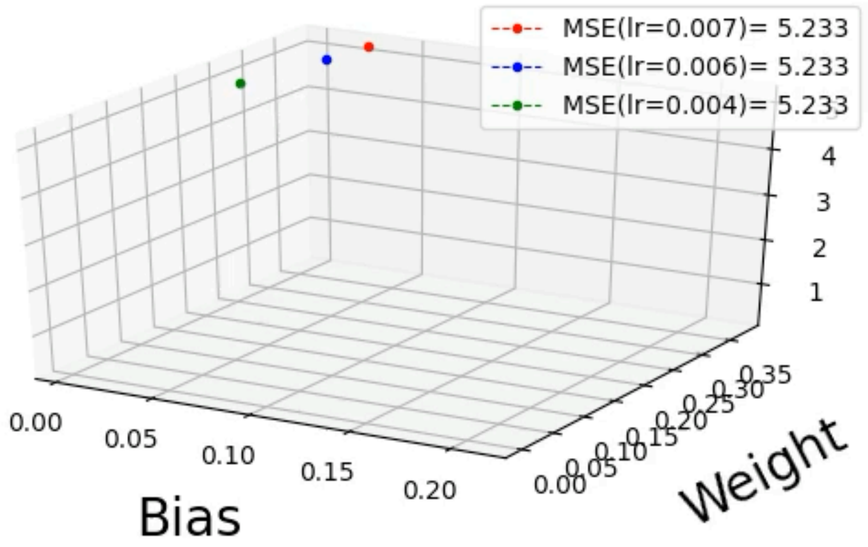
Learning rate

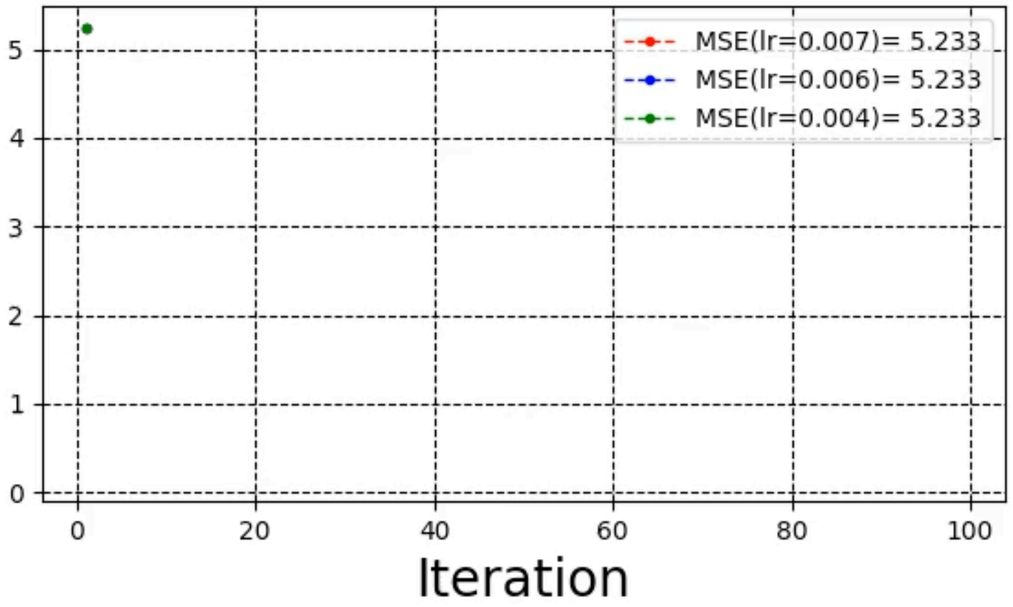
Effect of different learning rates

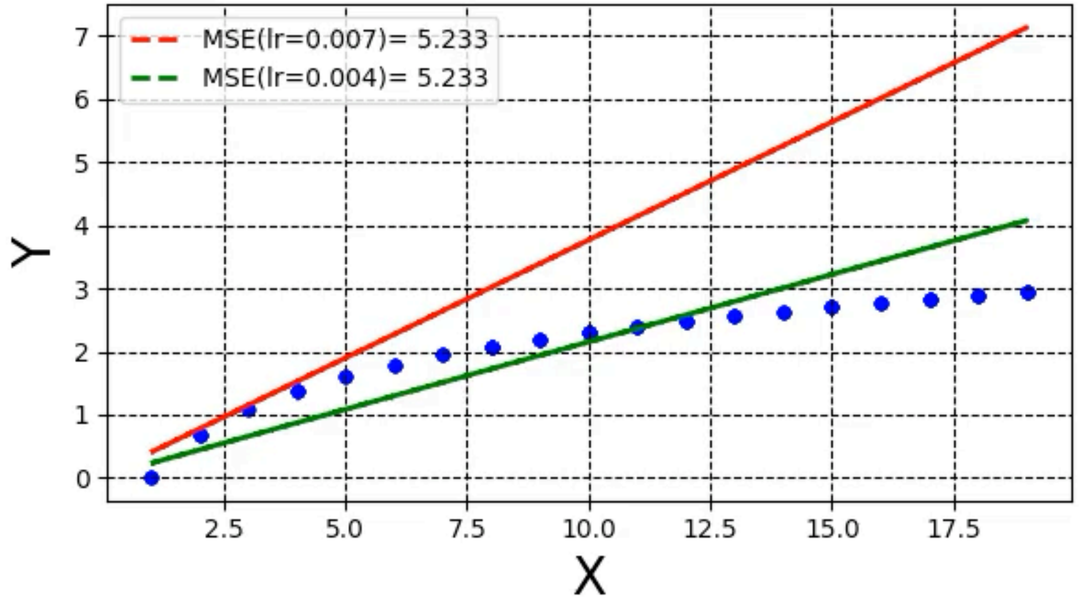


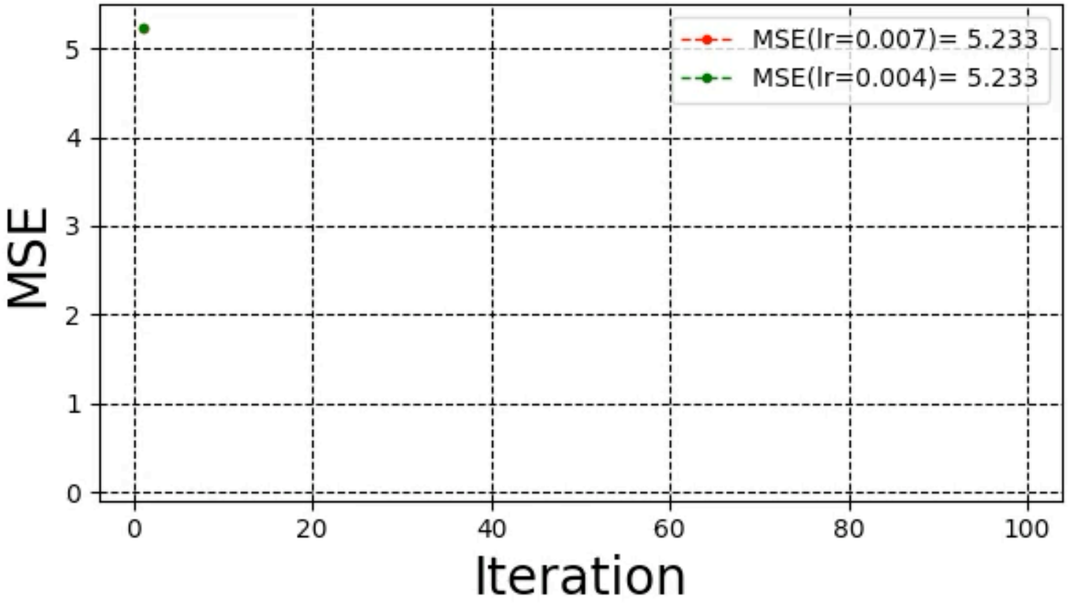


Learning rate





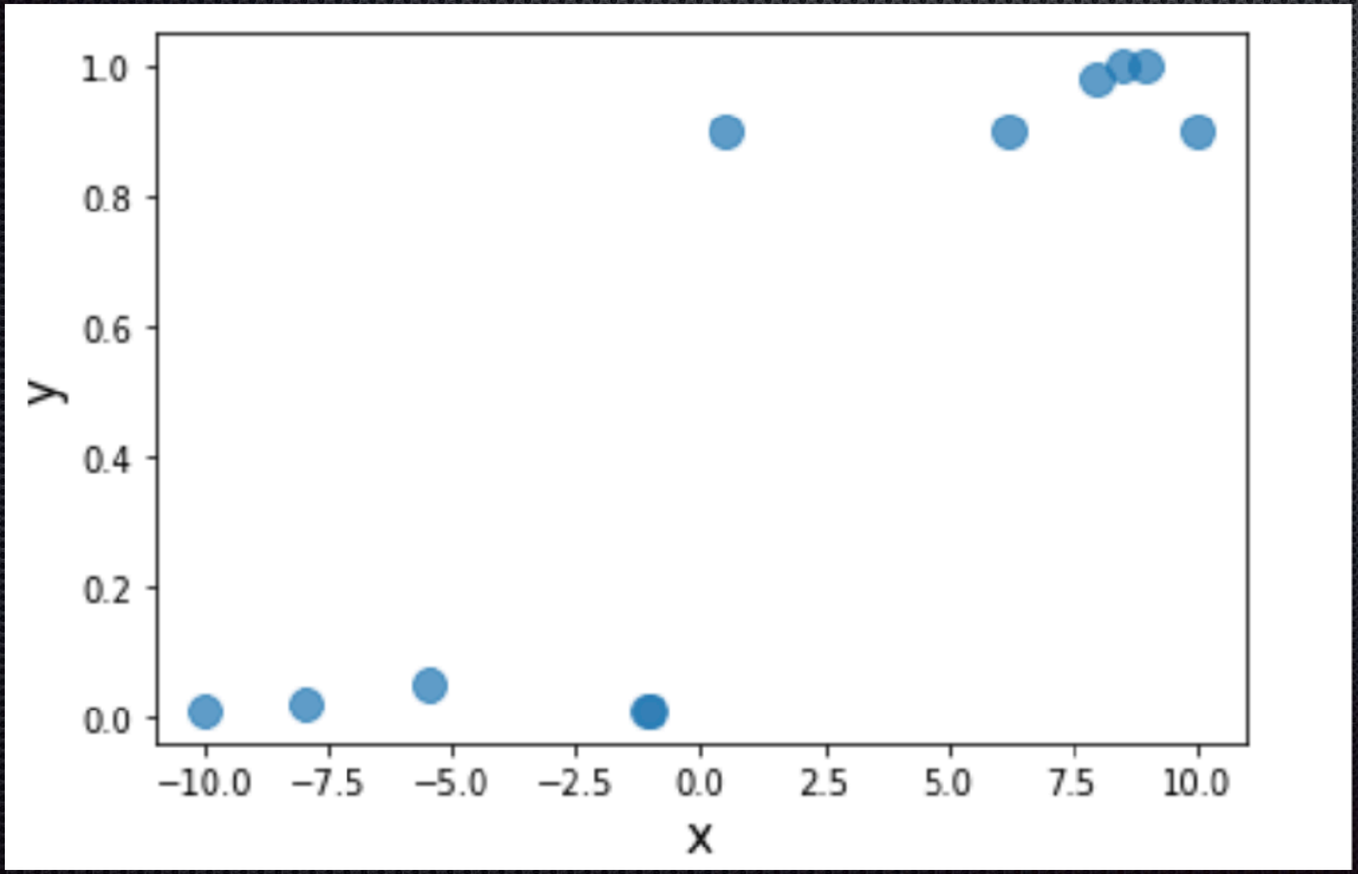




Categorical data

Question:

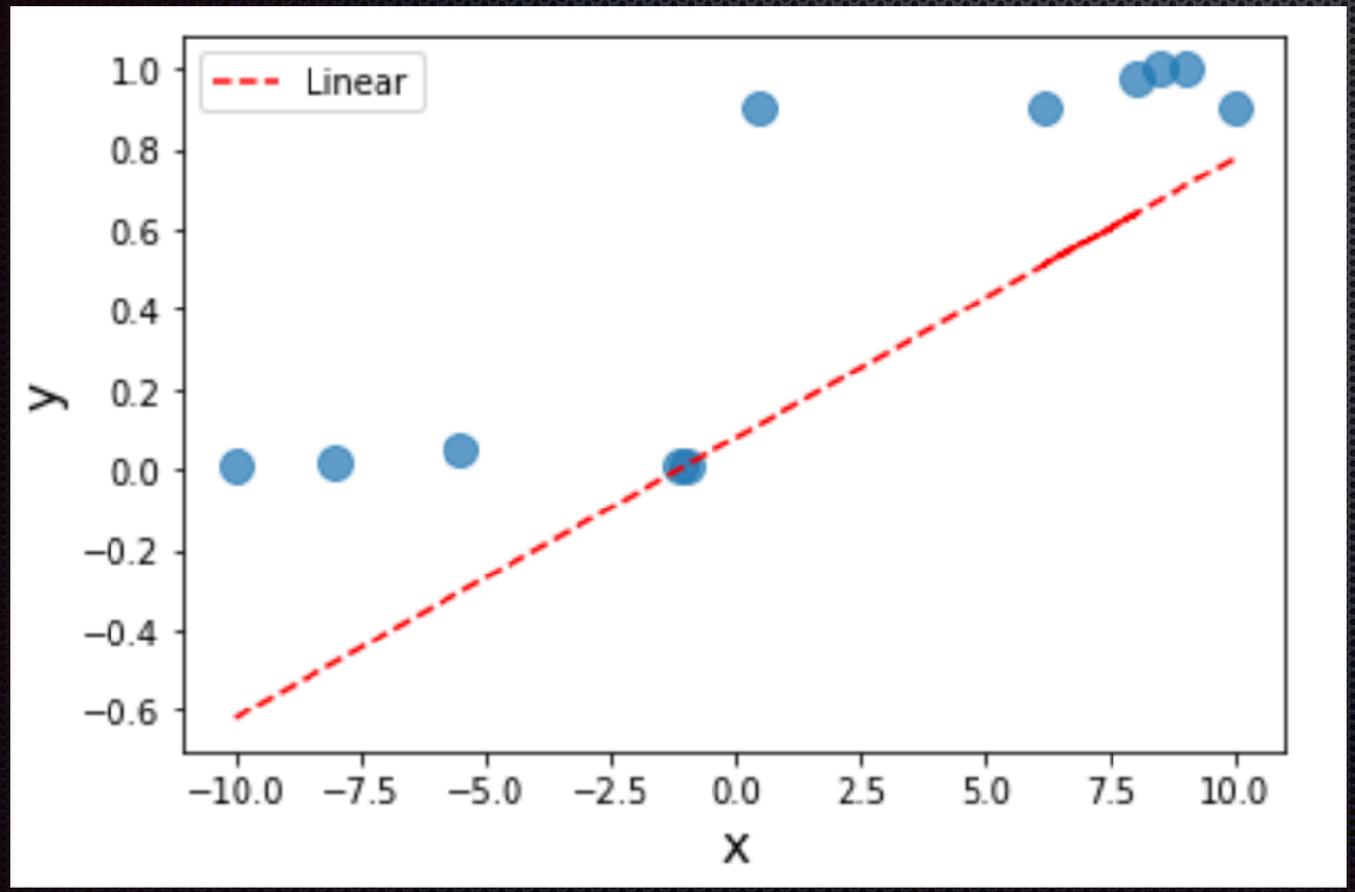
Can we repeat the previous steps for categorical data ?



Categorical data

Question:

Can we repeat the previous steps for categorical data ?

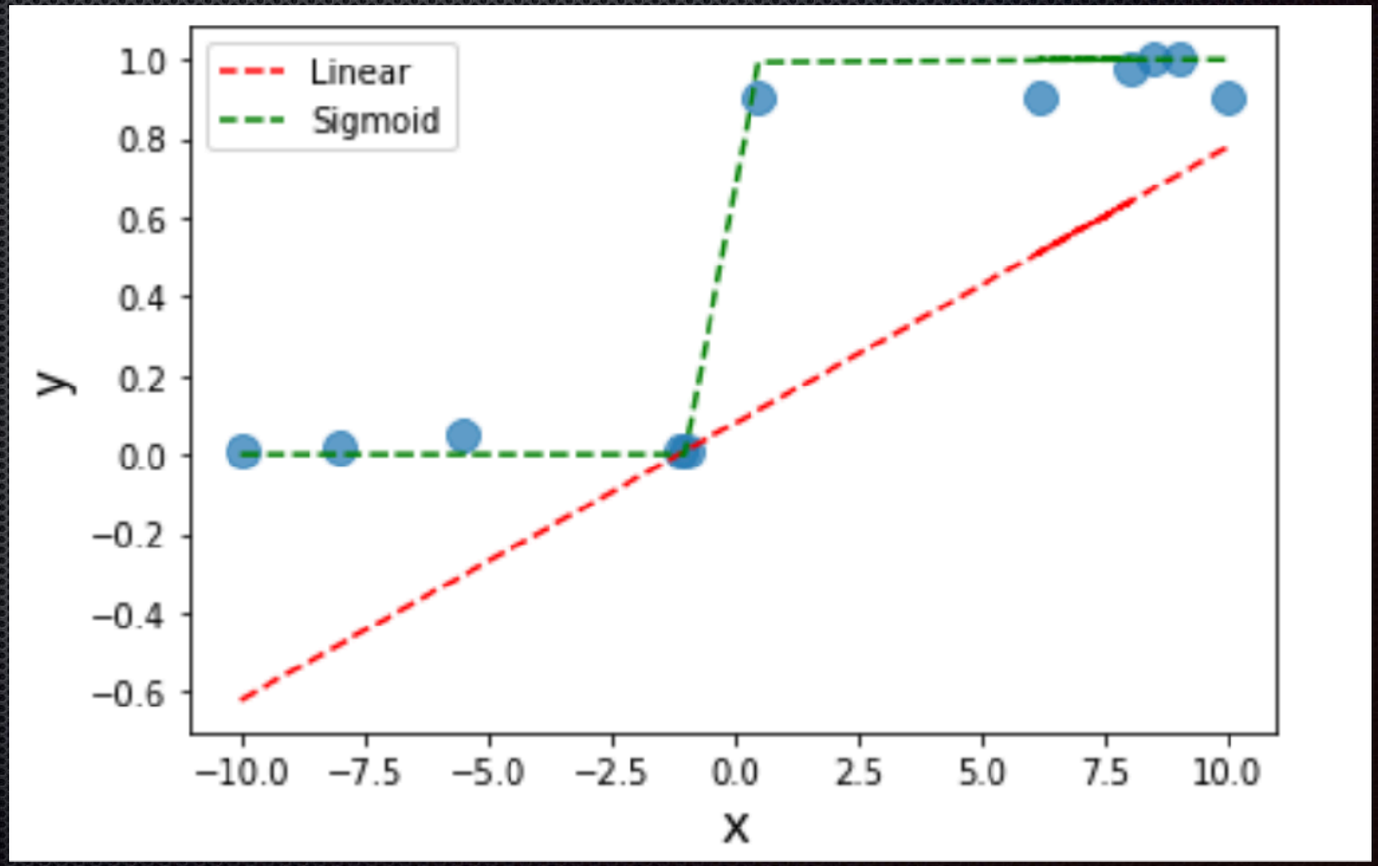


It seems not!! But why ?

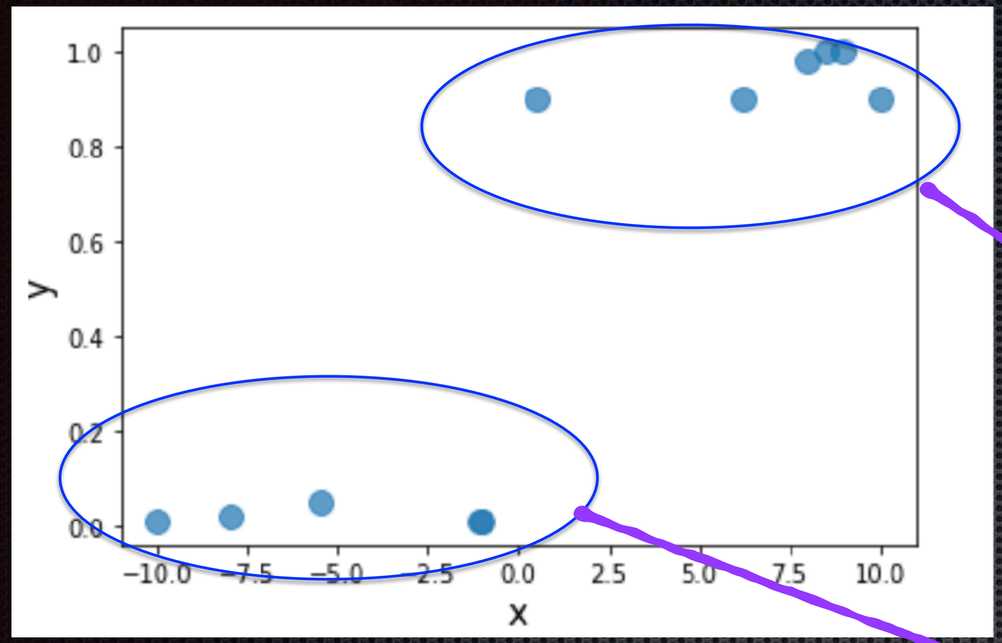
Categorical data

To fit categorical data we need a non linear fitting function

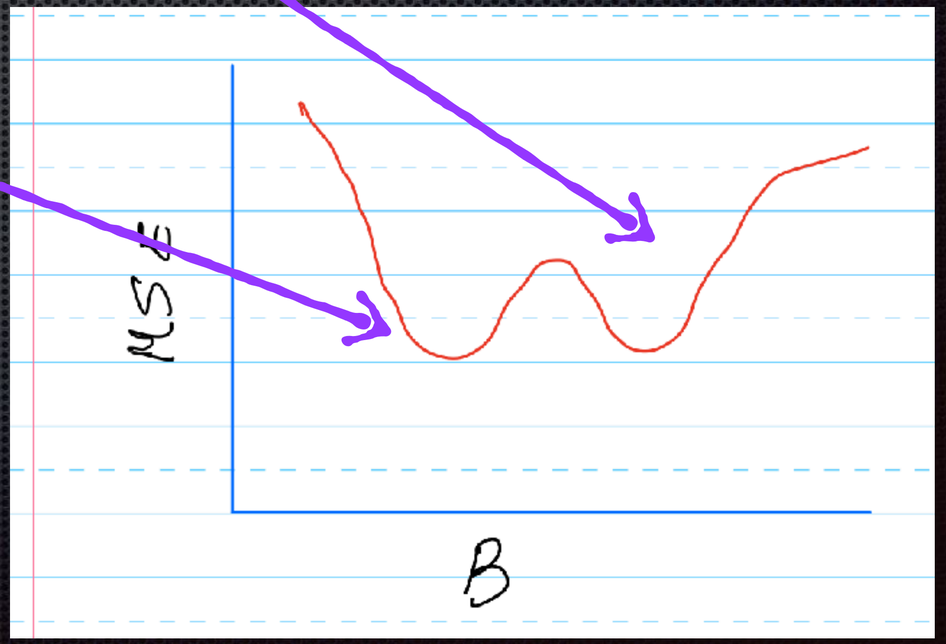
$$\text{Sigmoid} = \frac{1}{1+e^{-\hat{y}}} = \frac{1}{1+e^{-(B0+B1x)}}$$



Categorical data



In the case of categorical data the MSE will develop many local minima and will be hard to converge



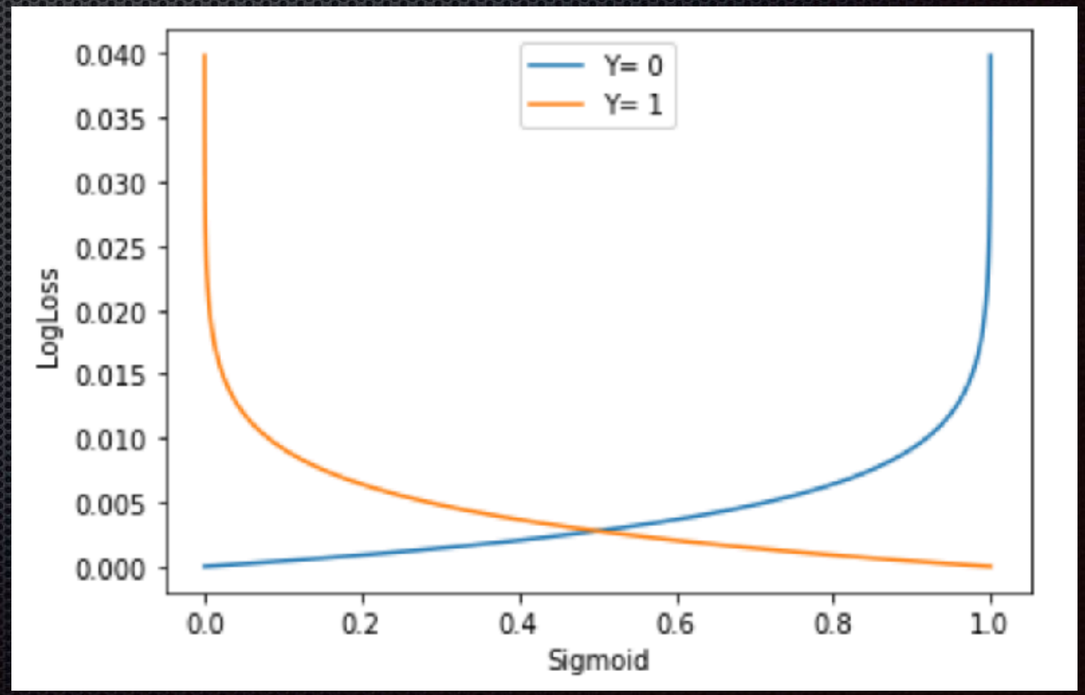
Categorical data-Loss function

Entropy Loss for Categorical data

From Information theory

$$\text{Entropy} = \log(\frac{1}{P})$$

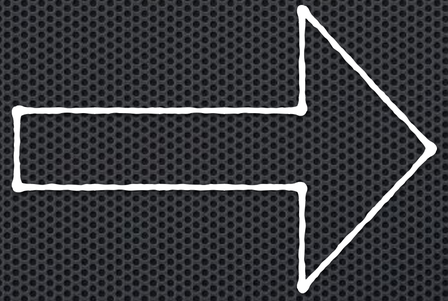
$$\text{Entropy Loss} = -\frac{1}{m}\sum^{m}\left[Y\log(\hat{Y}) + (1-Y)\log(1-\hat{Y})\right]$$



The model can converge to a global minimum

Loss functions

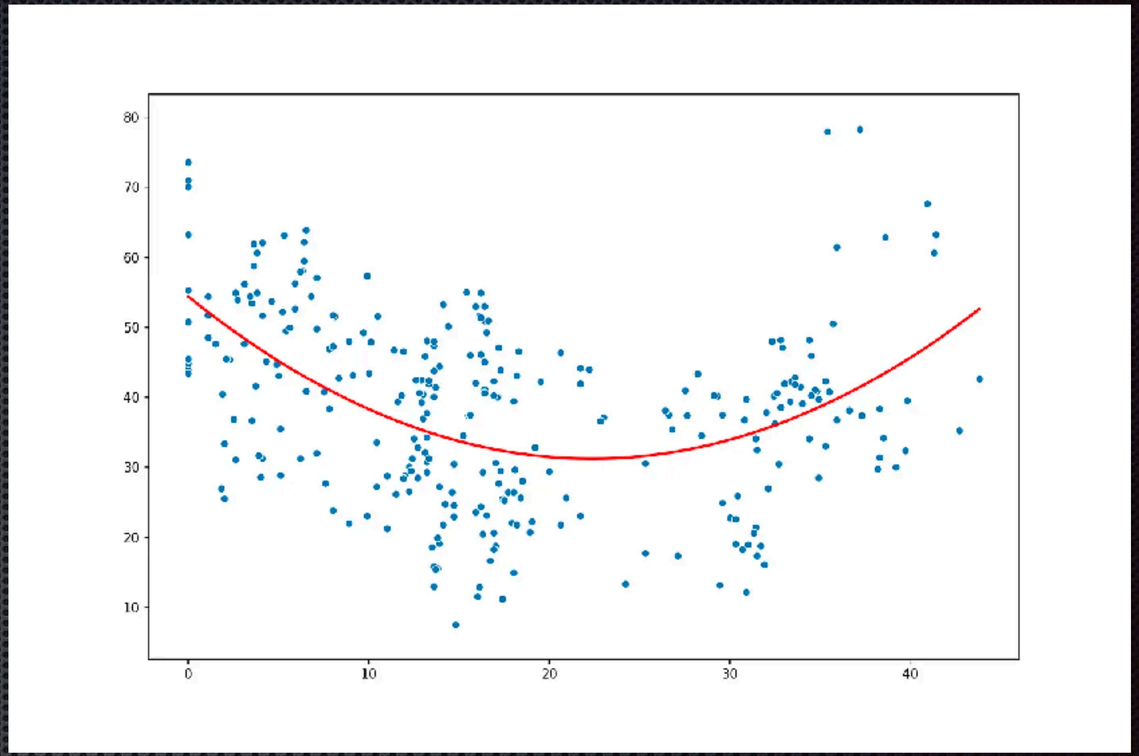
Continuous data  Mean error functions

Categorical data  Entropy loss functions

Polynomial regression model

We can fit the continuous data with polynomial model of power 2

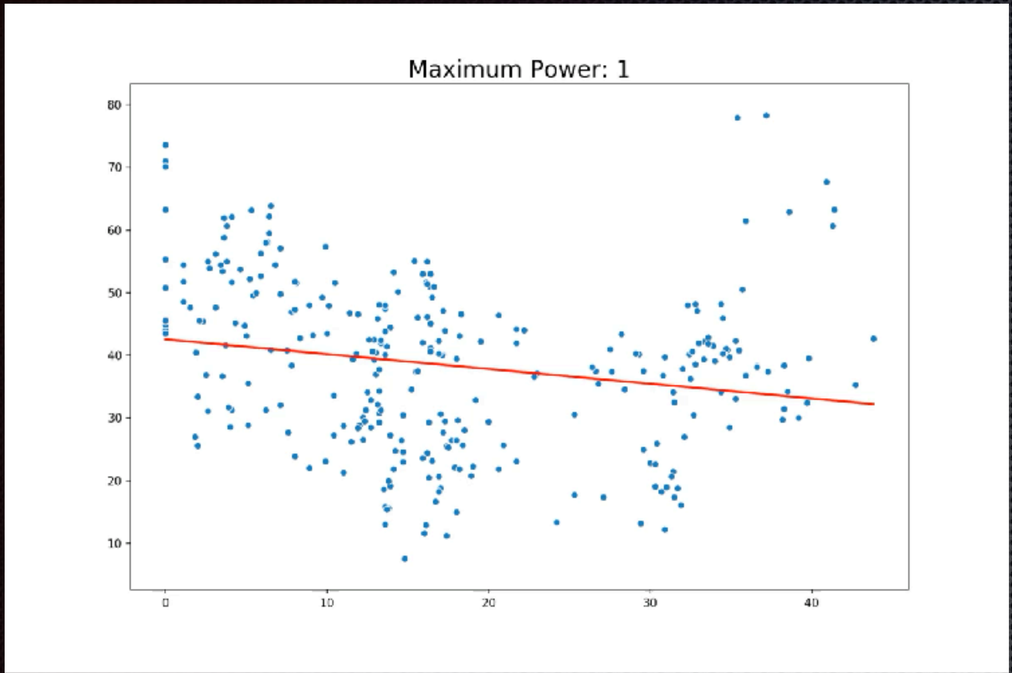
$$\hat{Y} = \omega_1 x + \omega_2 x^2 + B0$$

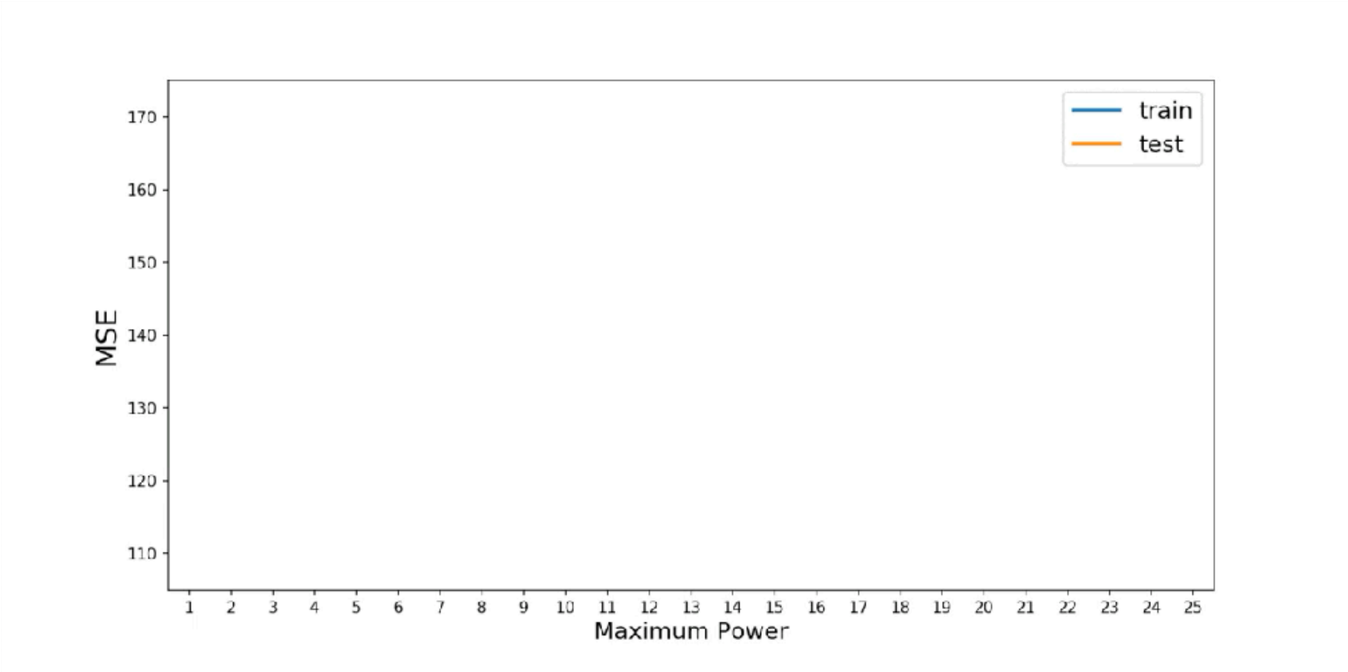


Adding more powers to the polynomial model adds more non-linarlity to The fitted function. How many powers fit the data best?

Polynomial regression model

$$\hat{Y} = \omega_1 x + \omega_2 x^2 + \ldots + \omega_{25}\, x^{25} + B0$$



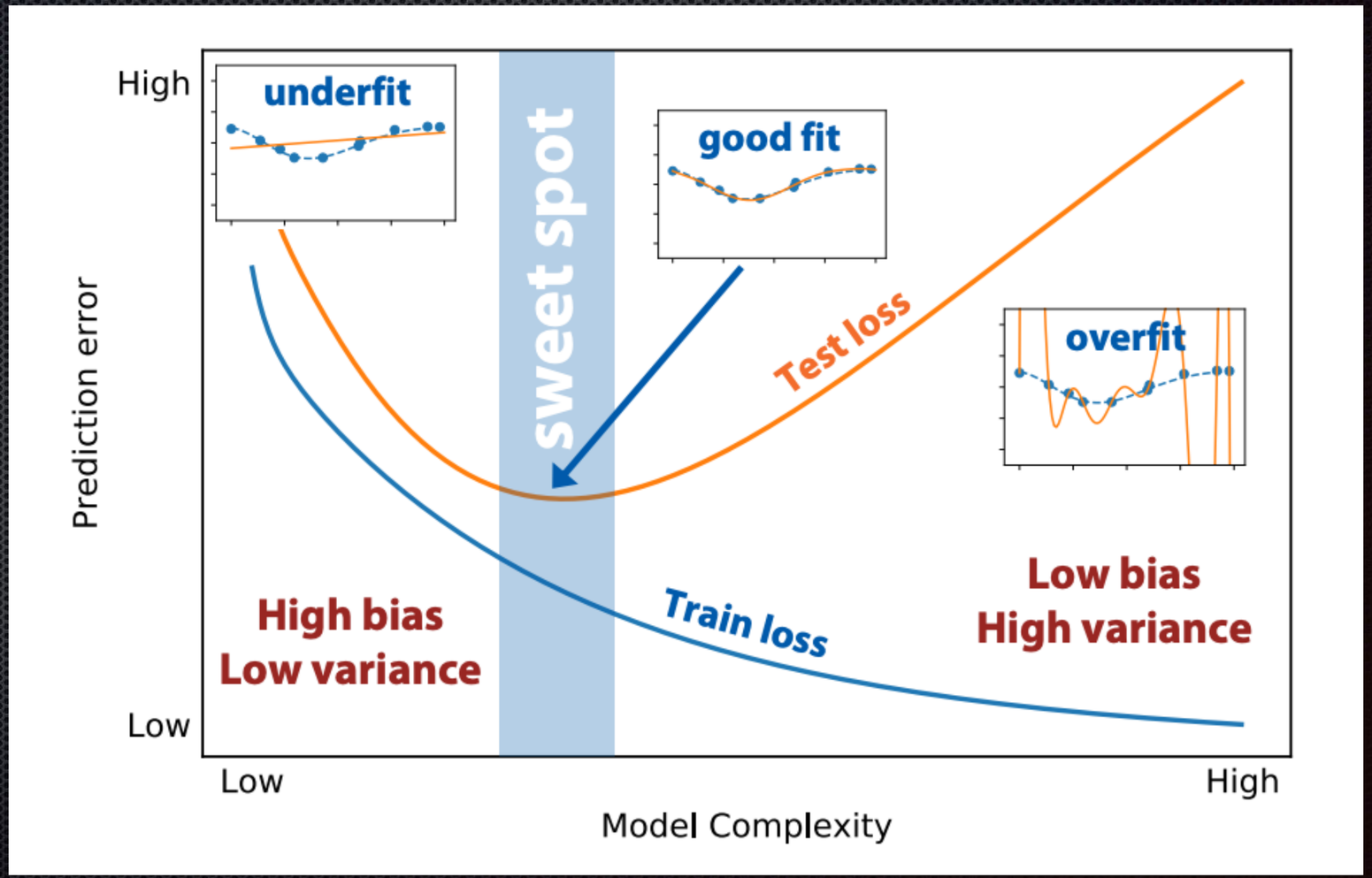


Why the training loss is much better than the test loss function ??

Underfitting & Overfitting

Adding more powers the model has the freedom to learn not only the pattern into the data but it learns the noise in the data as well.

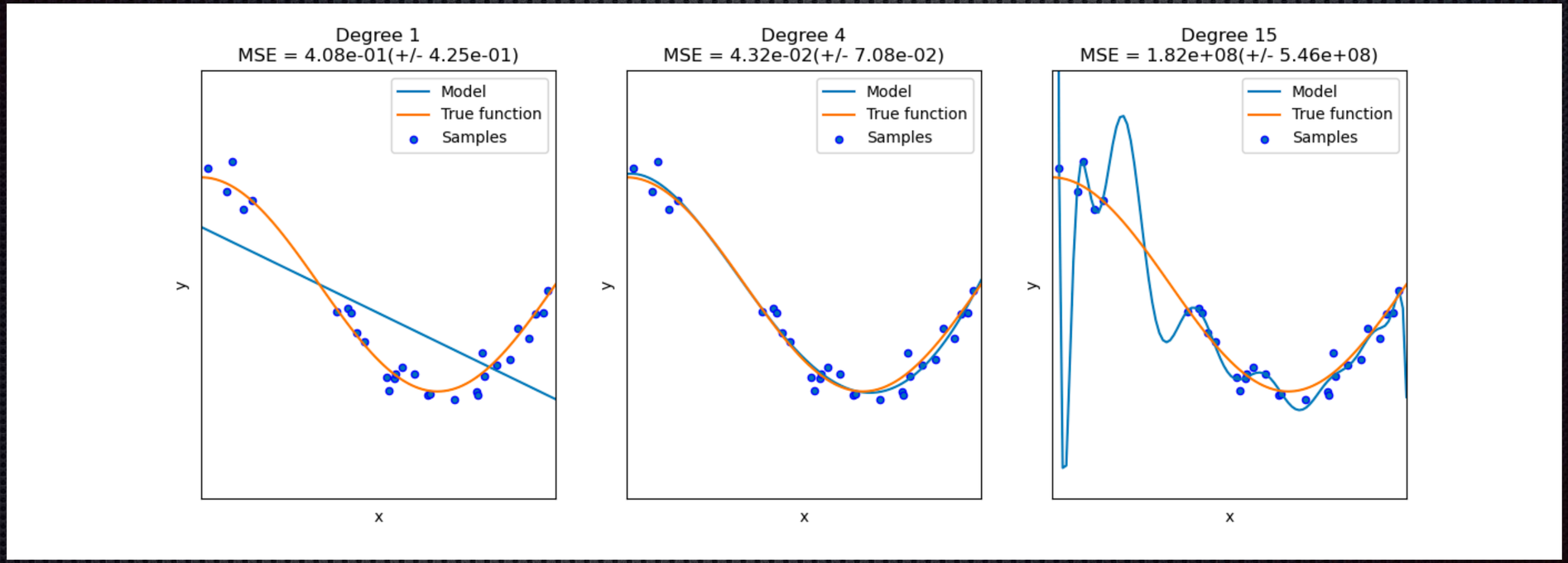
To avoid the overfitting problem we need to regularize the increasing number of powers



Overfitting danger

Regularized regression

The idea is to reconstruct a loss function that penalize the model
If it contains more degrees of freedom than required



We want to construct a loss function that forces the model
to remove the powers above the best fit value

Regularized regression

Error function with regularization

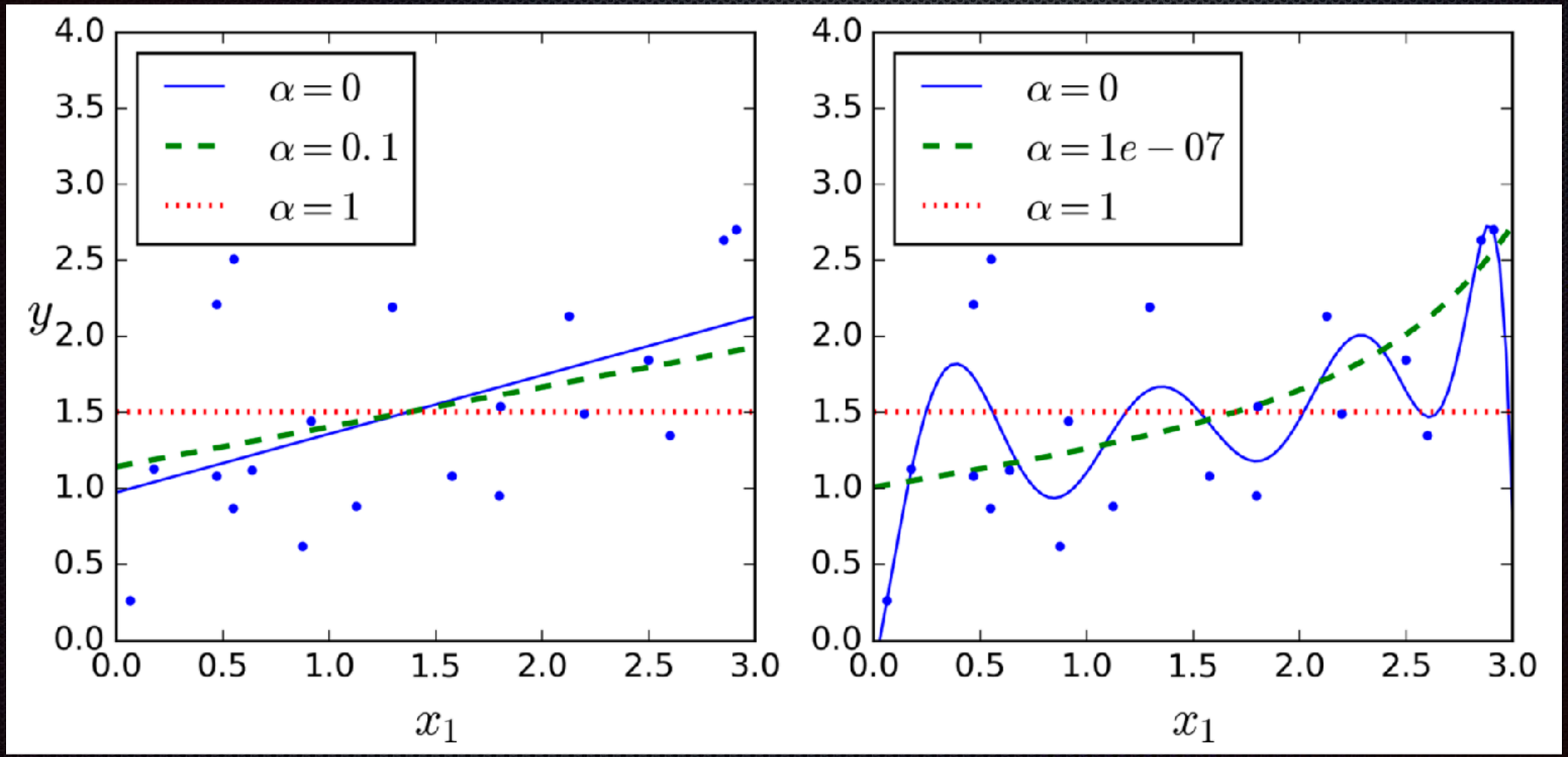
$$\hat{Y} = \omega_1 x + \omega_2 x^2 + \ldots + \omega_n \, x^n + B0$$

Lasso Model: $$L1 = \frac{1}{m} \sum^{m} \left(Y - \hat{Y} \right) + \lambda \sum^{n} \omega_n$$

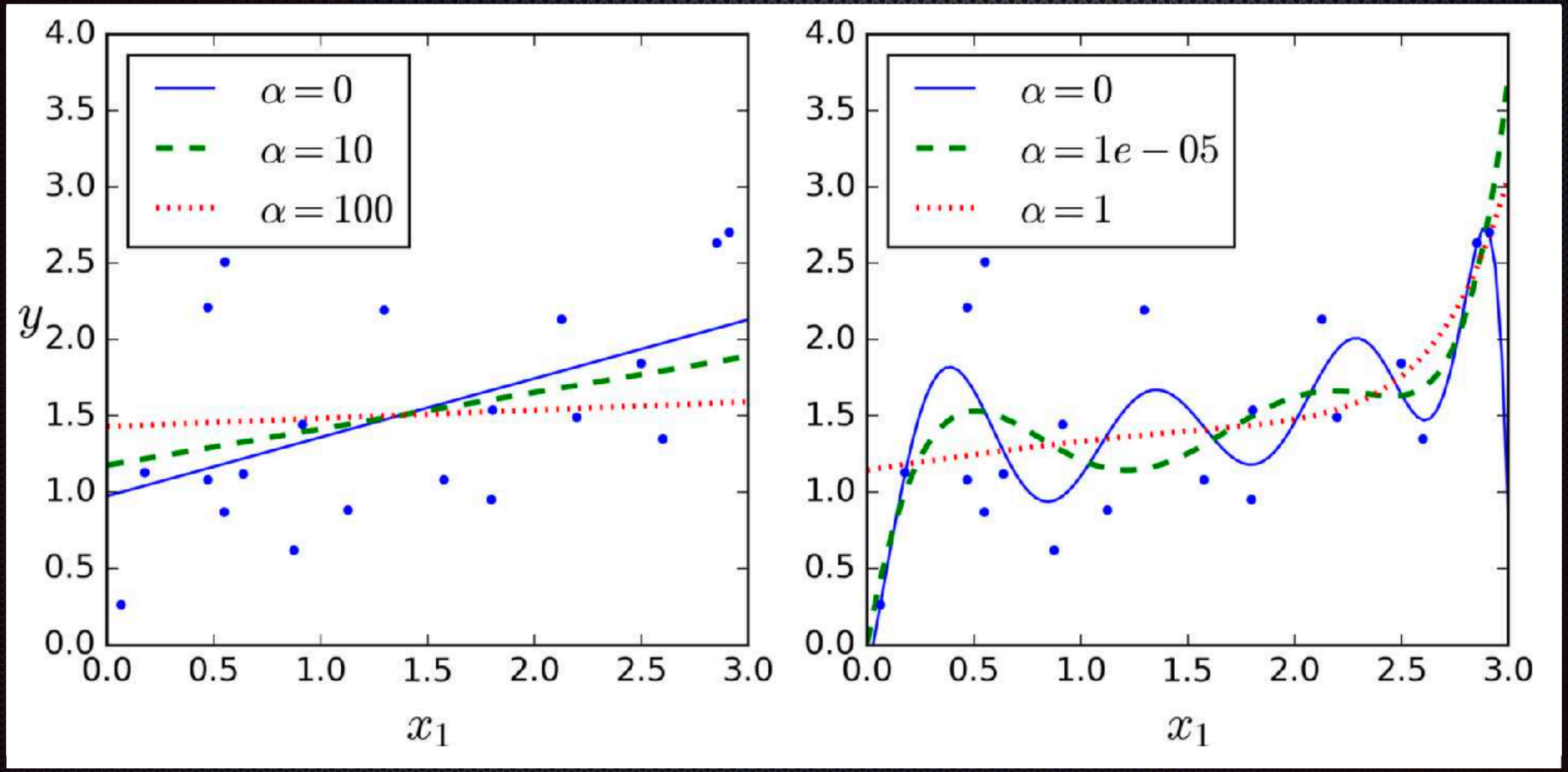
Ridge Model: $$L2 = \frac{1}{m} \sum^{m} \left(Y - \hat{Y} \right) + \lambda \sum^{n} (\omega_n)^2$$

Hyper-parameter to be optimized

Lasso Model



Ridge Model



Question

In which analysis problems we need to use Lasso over Ridge regression ?

To be continued…