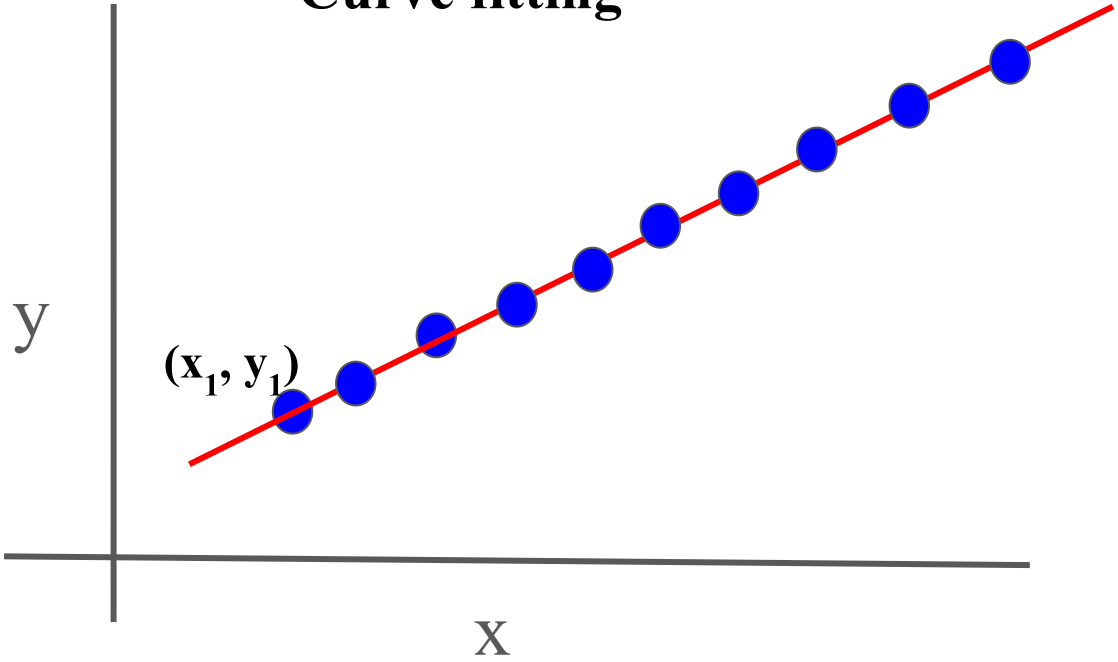
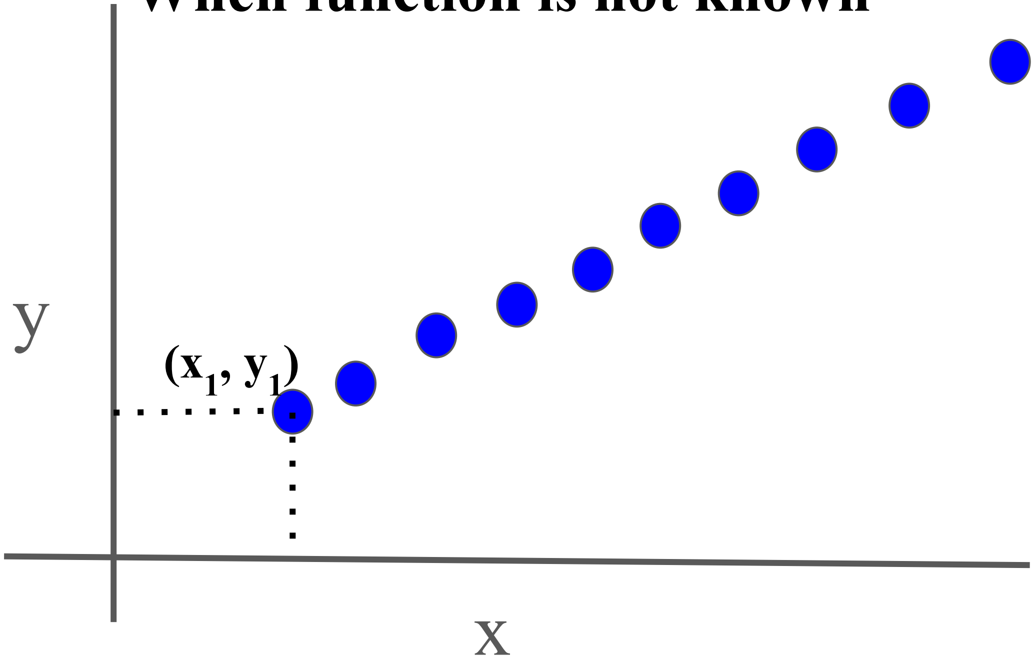
Linear Regression

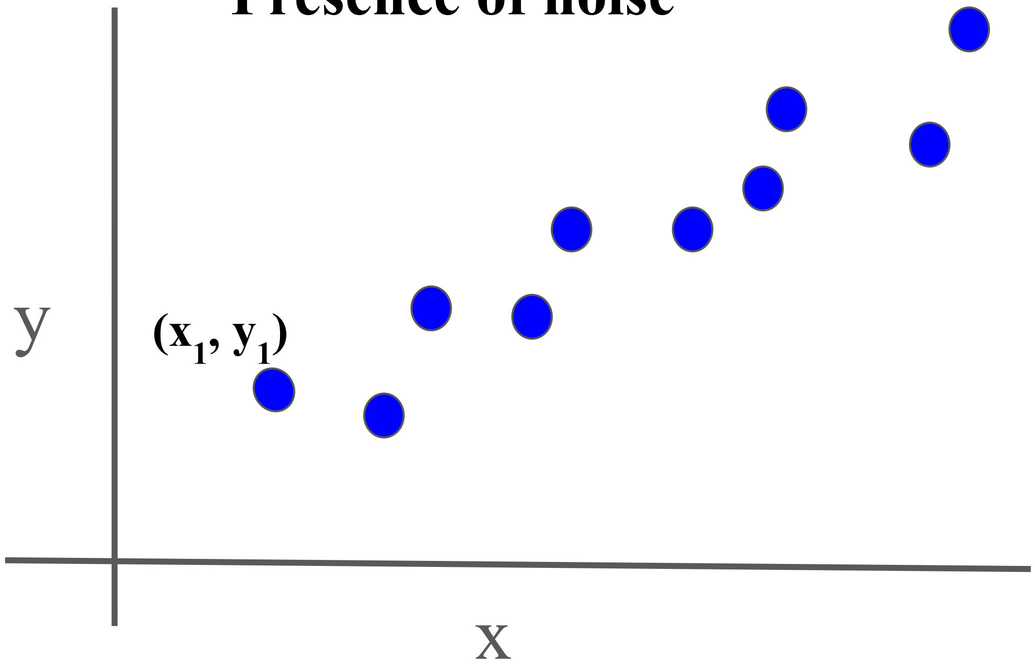
Curve fitting



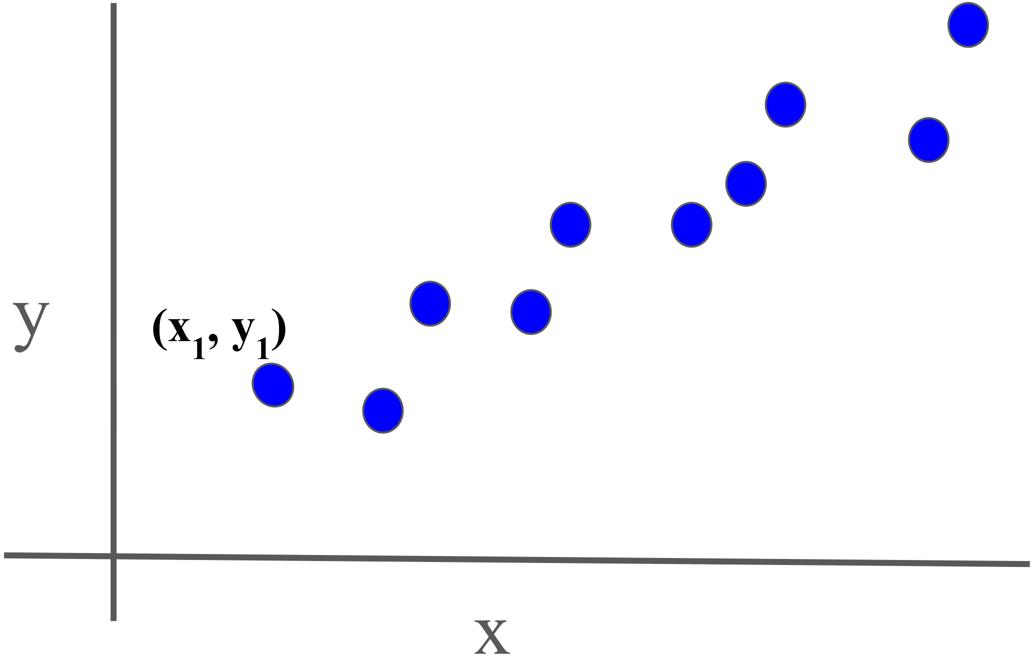
When function is not known



Presence of noise



Can we recover the true function?



Let’s try

First, we need to choose the function in general

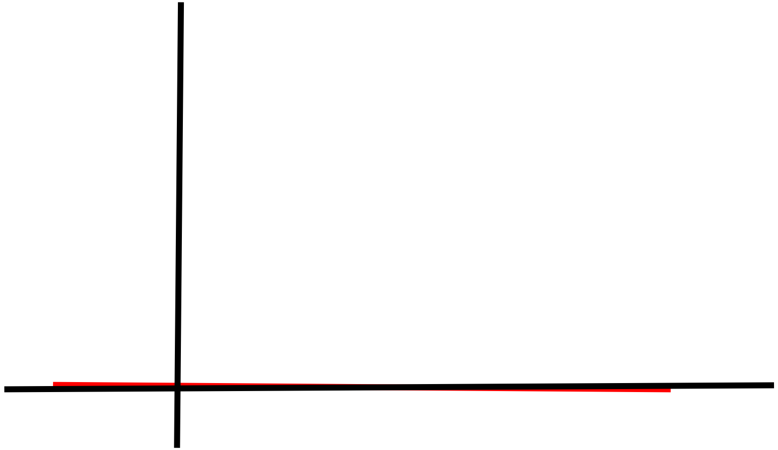
What does that mean?

Equation of line

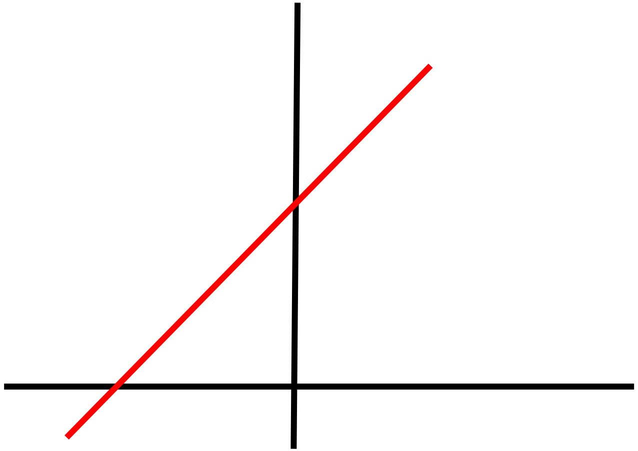
$$y=mx+c$$

For different values of $\boldsymbol{m}$ *and* $\mathbf{c}$*, we have different lines*

For example:

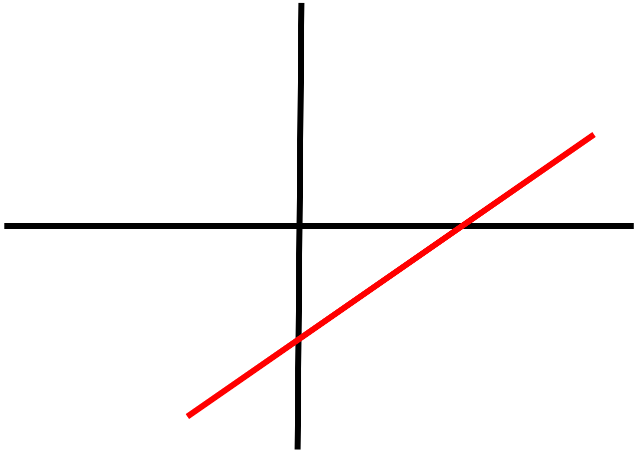


m=0; c=0

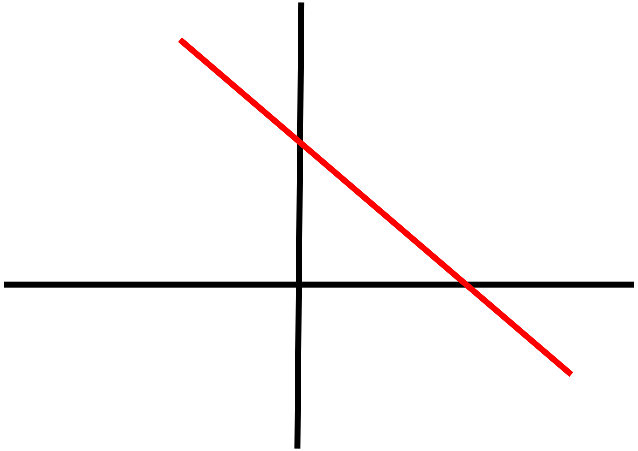


m=1; c=1

For example:



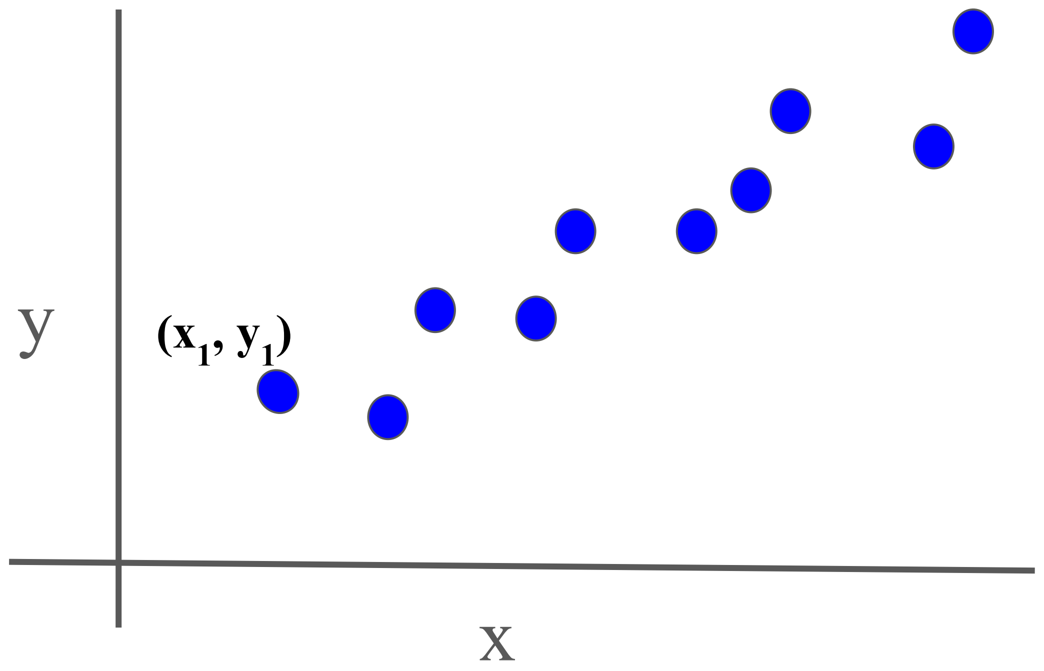
m=1; c=-1



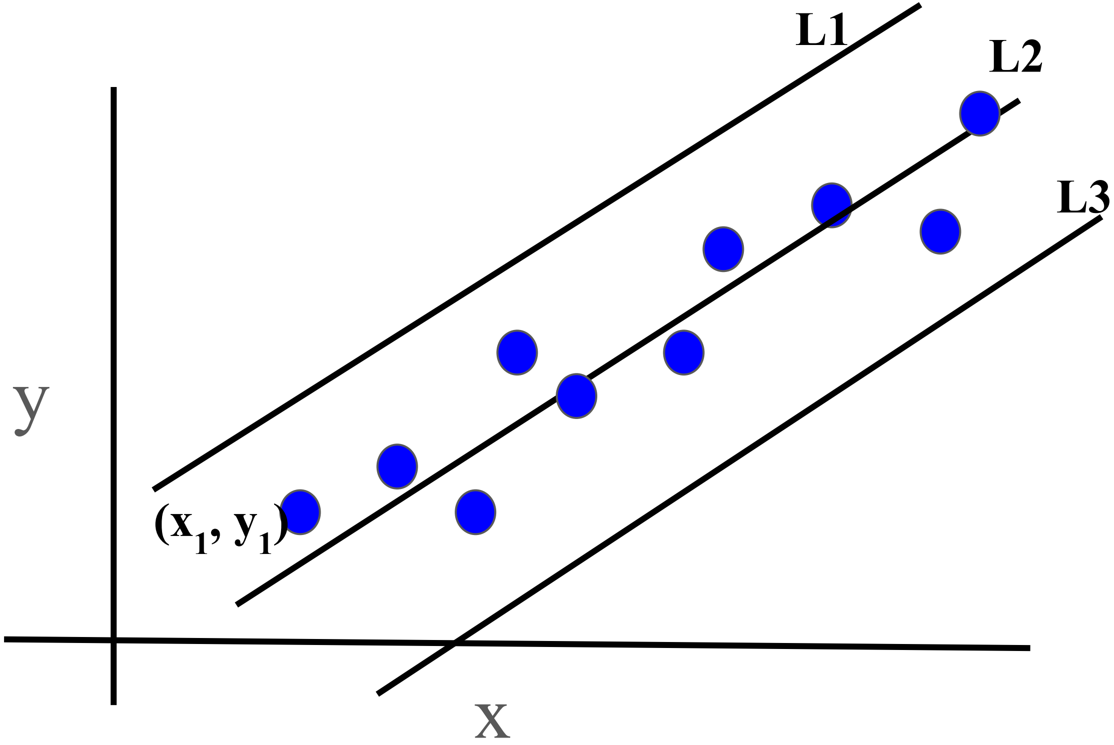
m=-1; c=1

Let's come back to the problem

Can we recover the true function?



Let's try to fit the line first



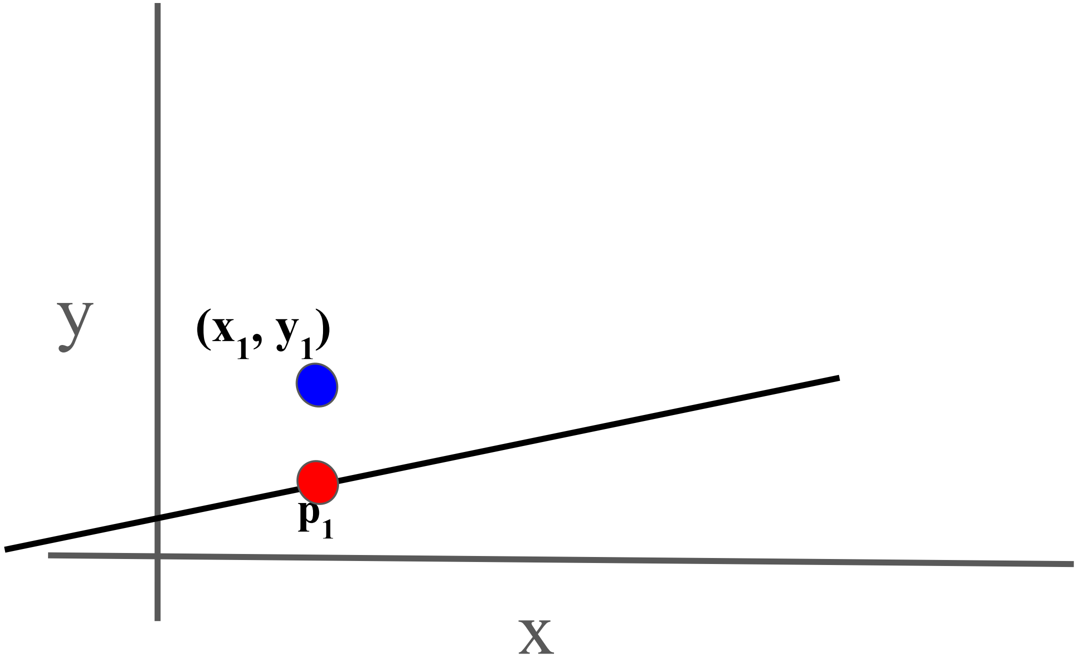
Which line is best L1, L2, and L3?

How to do it in computer?

We need to choose the optimal value of ***m*** *&* ***c***

Let's write it as follows:

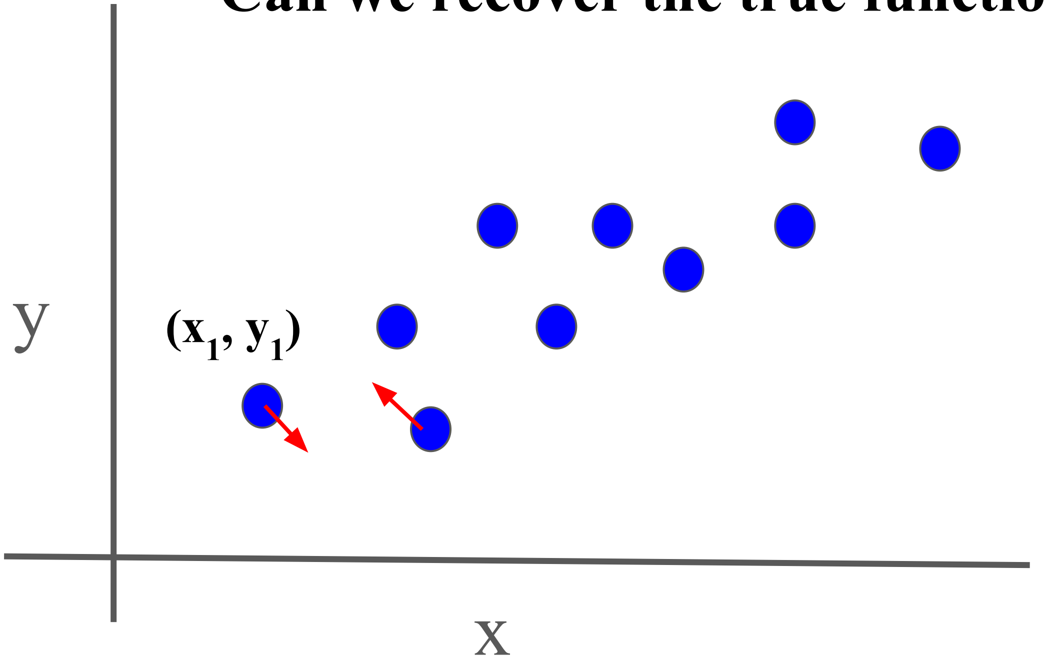
$$\boldsymbol{m * x \; + \; c \; = \; p}$$



*where, **m** & **c** is unknown.*

Our purpose is to make the value of ***p*** *as close to the value of* ***y*** *in the data*

Can we recover the true function?



We have 10 points:

	Actual	Target	Error
$mx_1 + c =$	p_1	y_1	$(y_1 - p_1)$
$mx_2 + c =$	p_2	y_2	$(y_2 - p_2)$
	.	.	.
	.	.	.
	.	.	.
$mx_{10} + c =$	p_{10}	y_{10}	$(y_{10} - p_{10})$

Now, we take the square of the error.

Why square of the error?

Because +ve and -ve errors can cancel out each other.

Therefore;

Actual	Target	Error
$mx_1 + c = p_1$	y_1	$(y_1 - p_1)$
$mx_2 + c = p_2$	y_2	$(y_2 - p_2)$
.	.	.
.	.	.
.	.	.
$mx_{10} + c = p_{10}$	y_{10}	$(y_{10} - p_{10})$

$$\text{Total error} = (\mathbf{y_1 - p_1})^2 + (\mathbf{y_2 - p_2})^2 + \cdots\cdots\cdots + (\mathbf{y_{10} - p_{10}})^2$$

- Alternatively

$$\text{Total error} = (\mathbf{y_1}\text{-}\mathbf{p_1})^2 + (\mathbf{y_2}\text{-}\mathbf{p_2})^2 + \cdots\cdots\cdots + (\mathbf{y_{10}}\text{-}\mathbf{p_{10}})^2$$

$$\text{Total error} = \sum_{\{i=1\}}^{n} (y_i - p_i)^2$$

Our objective is to minimize the error

- # Simple trick from Calculus

Minimization of function:

$$\nabla \mathbf{f} = \mathbf{0}$$

Take the derivative of the function and put it equal to zero

Hence, we get the optimal value of m & c

$$y = mx + c$$

We can use this (m,c) for predicting the output for the new values of x

CODE

https://github.com/aksiitbhu/Machine-Learning-2024/blob/main/LinearRegressionLab1GEU2024.ipynb

Github link for course materials

https://github.com/aksiitbhu/Machine-Learning-2024/tree/main

Thanks