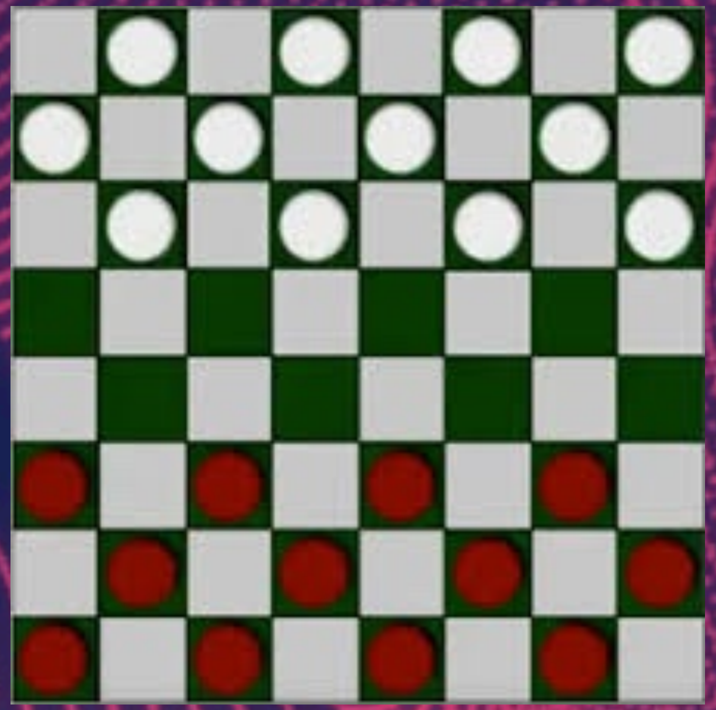
LINEAR REGRESSION

AI CLUB

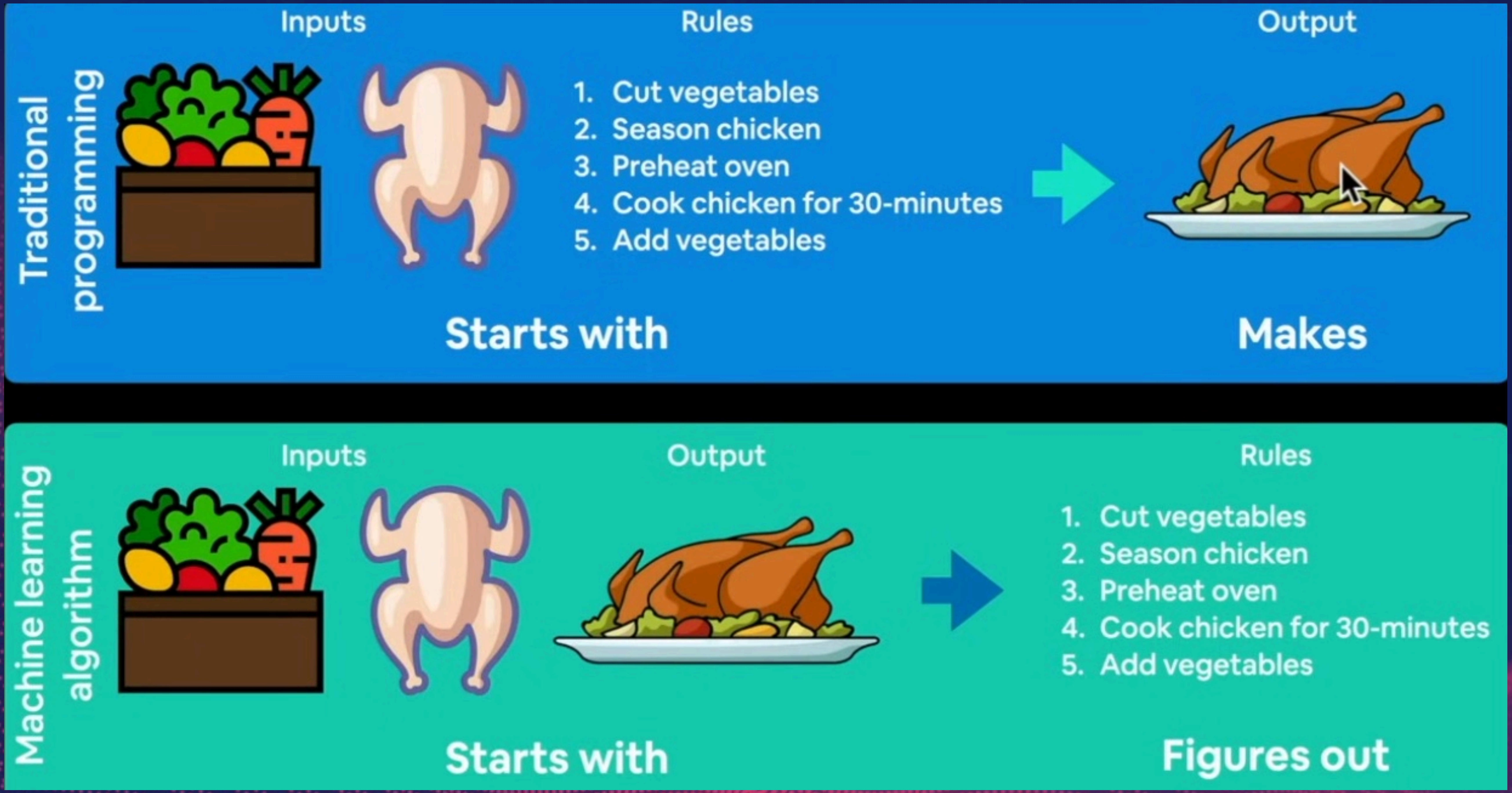
MACHINE LEARNING

Field of study that gives computers the ability to learn without explicitly being programmed
-Arthur Samuel



Samuel programmed the computer to play thousands of games against itself. Through this process, the computer learnt to identify good and bad positions, eventually becoming better than Samuel himself at playing checkers

TRADITIONAL PROGRAMMING VS MACHINE LEARNING



TYPES OF MACHINE LEARNING

Supervised

Algorithm learns from labeled data ie. input features (x) and their correct output labels (y). The goal for the model is to learn a mapping from inputs to outputs so that it can predict or classify the output for new, unseen data.

Unsupervised

Model is trained on unlabeled data. The goal of unsupervised learning is to discover interesting similarities, patterns or differences in the data without any predefined labels.

Reinforcement

Reinforcement learning is a machine learning algorithm that focuses on encouraging desired behaviors through rewards and discouraging undesired ones through penalties. It improves its performance by learning from the outcomes of its actions through trial and error.

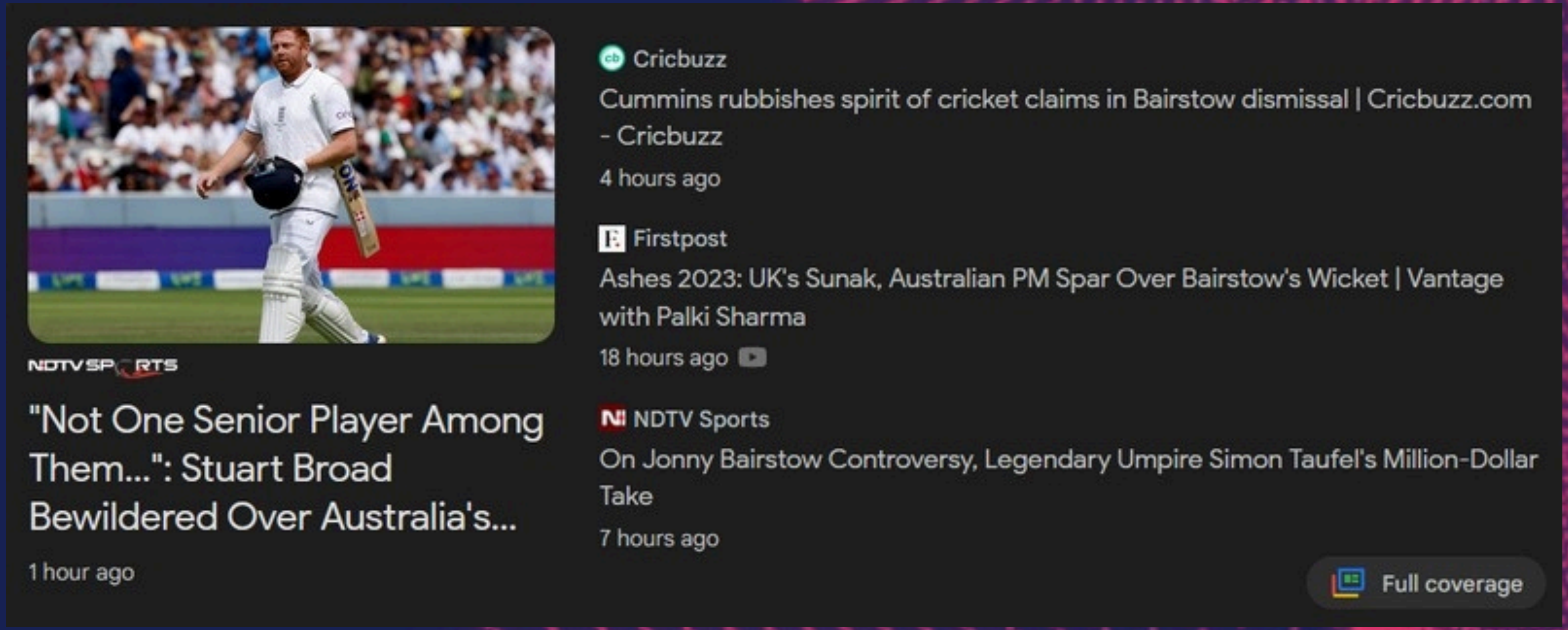
SUPERVISED LEARNING

Algorithm learns from labeled data
ie. input features (x) and their
correct output labels (y).

For example : After training on a dataset
with pictures of fruits and their labels,
the model is given a new fruit, such as a
banana, to identify.
The trained model examines the fruit's
shape and color, identifies it as a
banana.

UNSUPERVISED LEARNING

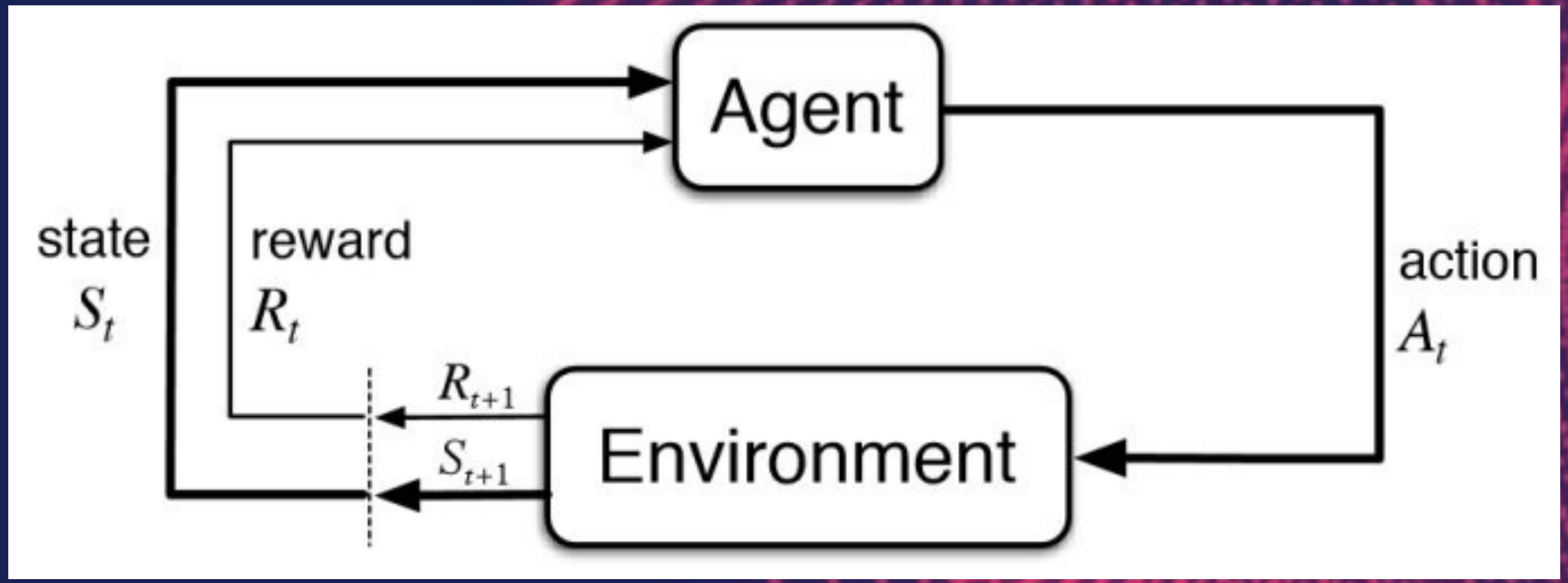
In unsupervised learning the algorithm learns from unlabelled data allowing the algorithm to act on that information without guidance.
Here the task of machine is to group unsorted information according to similarities, patterns and differences without any prior training of data.



Google news is a good example of Unsupervised learning.

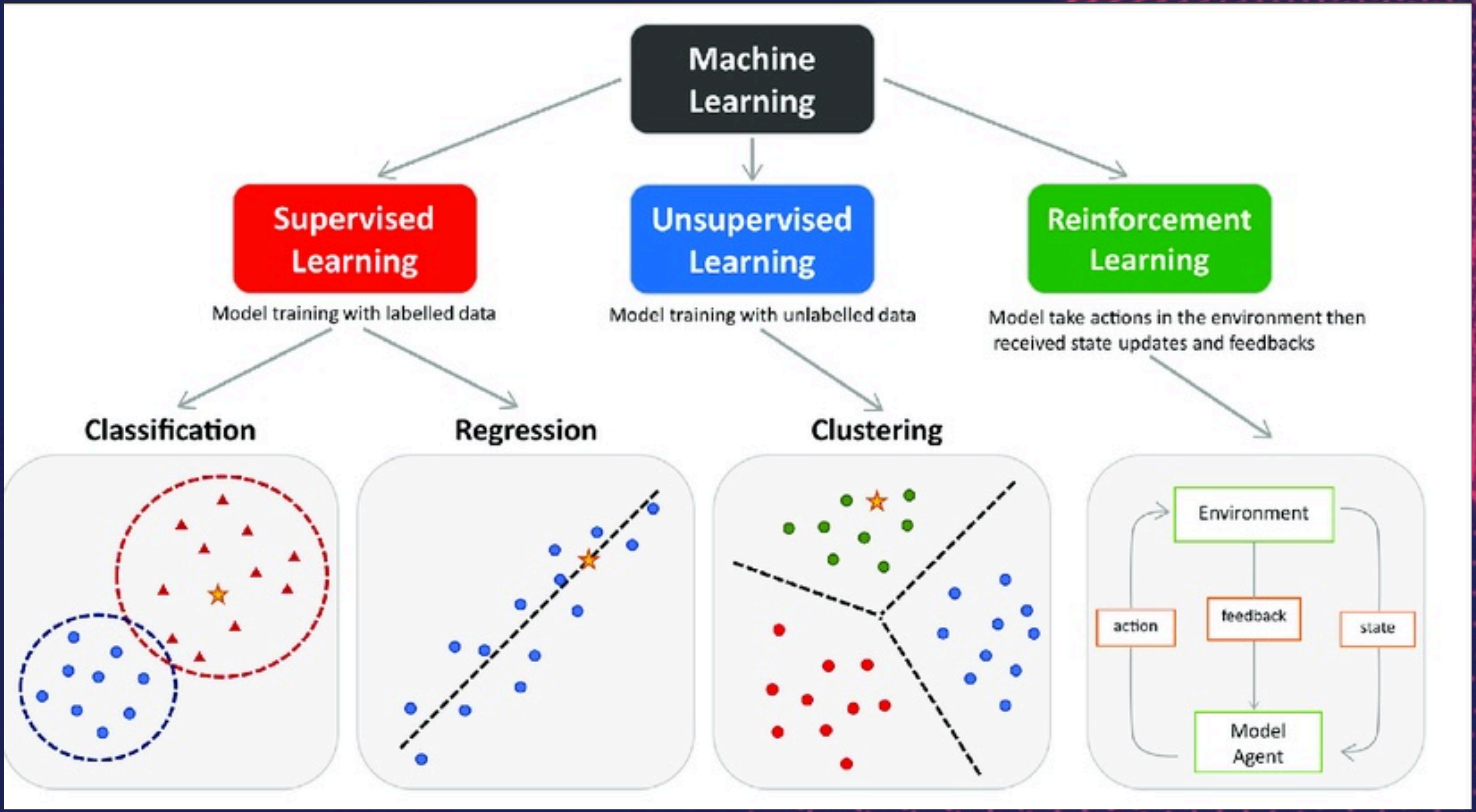
REINFORCEMENT LEARNING

Reinforcement learning is a machine learning training method based on rewarding desired behaviors and/or punishing undesired ones. In general, a reinforcement learning agent is able to perceive and interpret its environment, take actions and learn through trial and error.



AlphaZero, a chess engine developed by DeepMind is a great example of the application of Reinforcement learning

TYPES OF MACHINE LEARNING

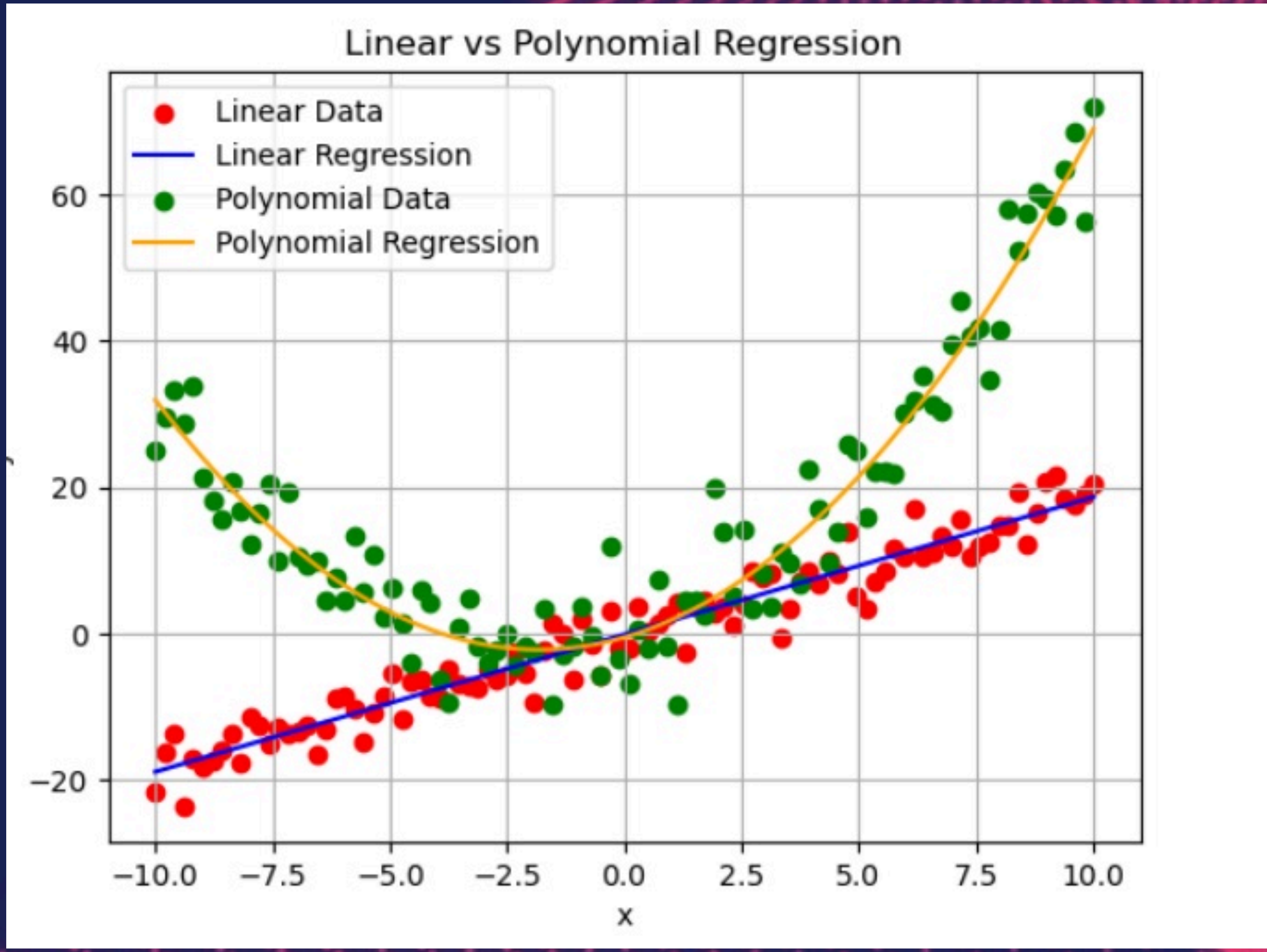


REGRESSION

“**Hello World**” of machine learning algorithms

Regression is a type of supervised learning technique that establishes a predictive relationship between labels and data points. It aims to predict a **continuous-valued output** by mapping input variables to a continuous function.

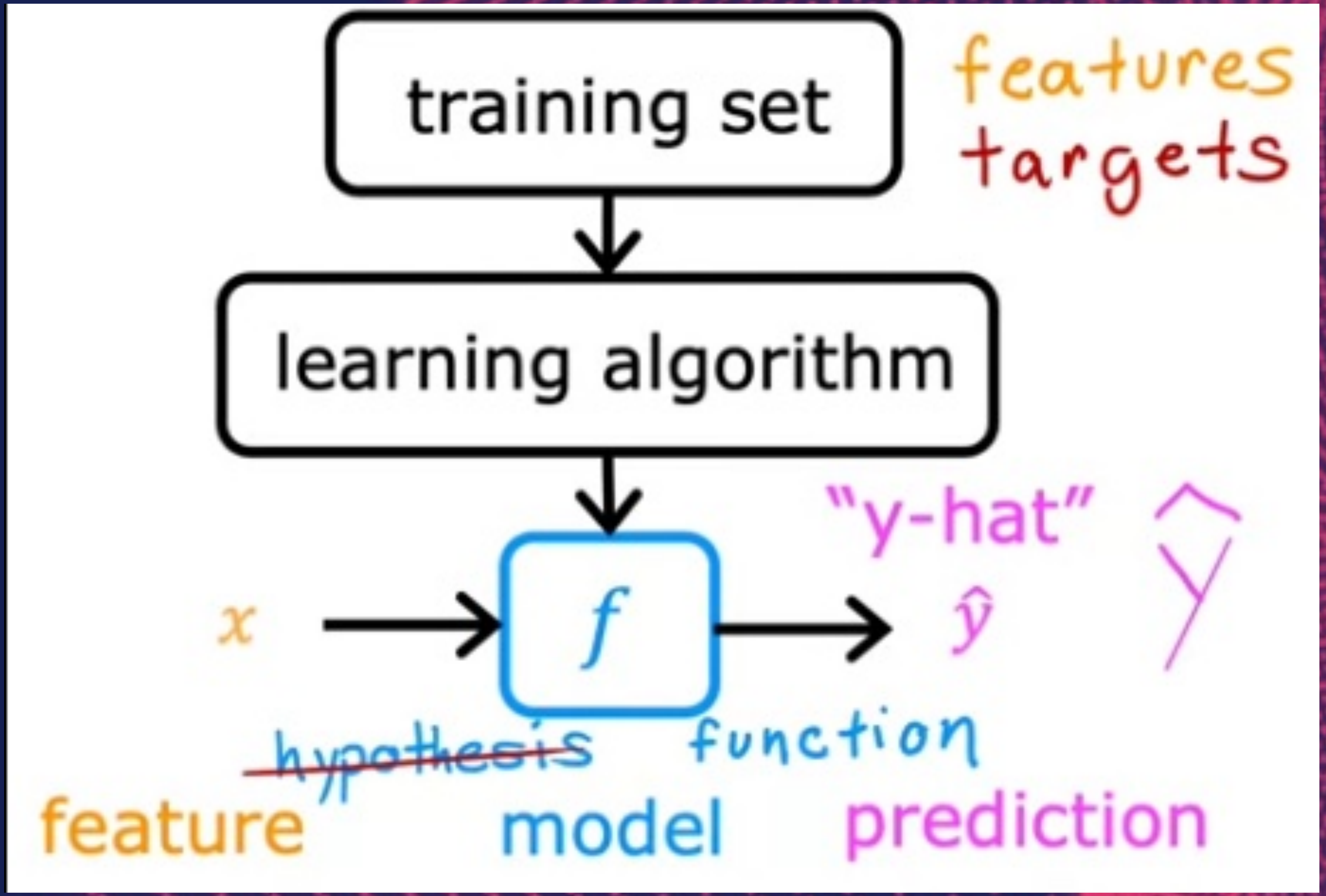
For example: Housing Price Prediction based on characteristics like size, number of rooms etc



REGRESSION

Training set refers to the data used to train our model. It contains input features and their output targets ie. the correct output values.

The algorithm learns from the training set and then comes up with a continuous function also called as **hypothesis** which gives the predicted output $\hat{y}$ for an input x

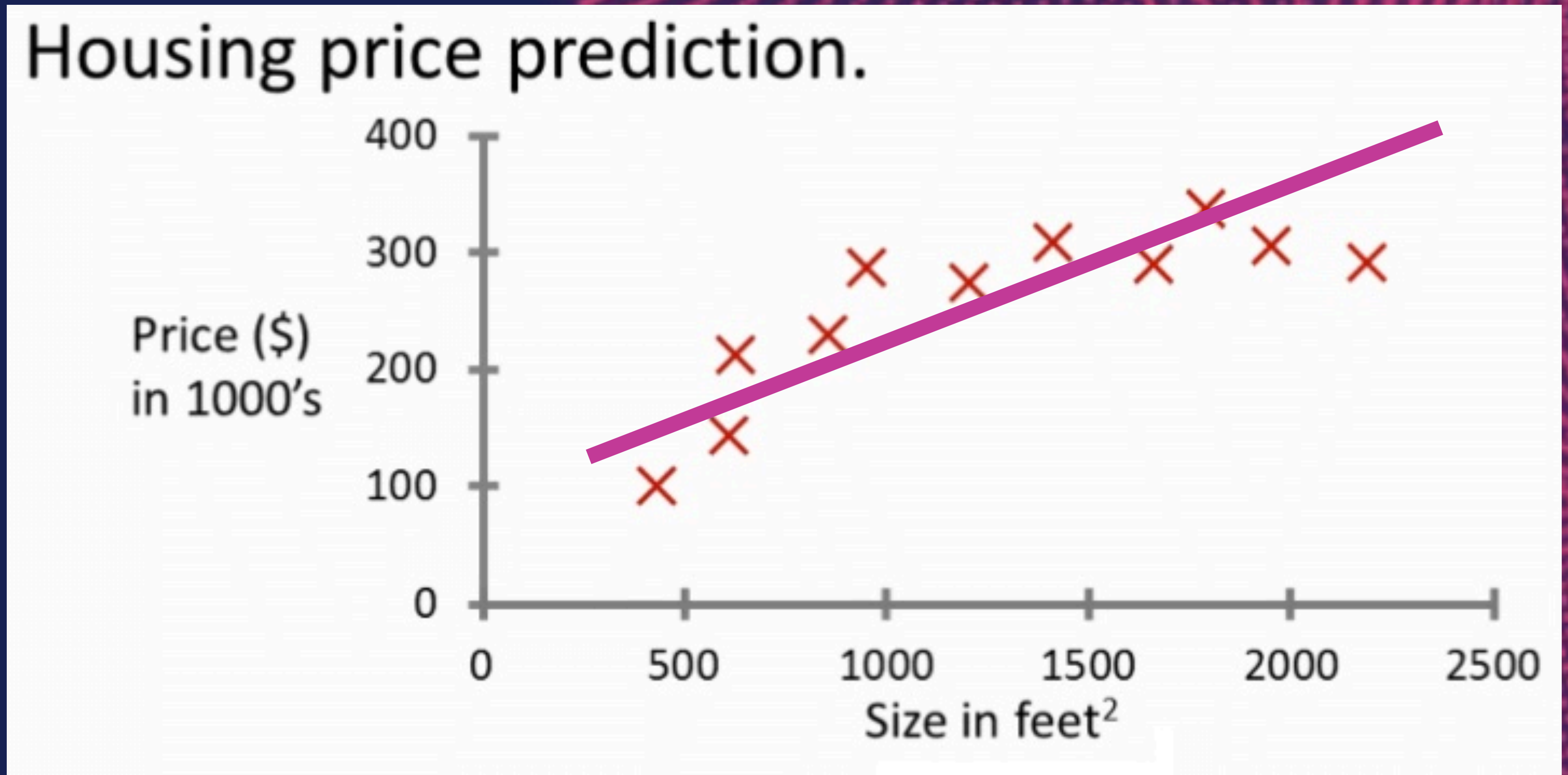


LINEAR REGRESSION

Linear regression is basically fitting a straight line to the given data. The hypothesis is of the form

$$\hat{y} = wx + b$$

- x refers to the input features
- w is the slope of the line also called weight
- b is the y intercept also called bias
- $\hat{y}$ is the value predicted by our model



How do we come up with the optimal parameters w and b to get the best-fit line for a given dataset ?

COST FUNCTION

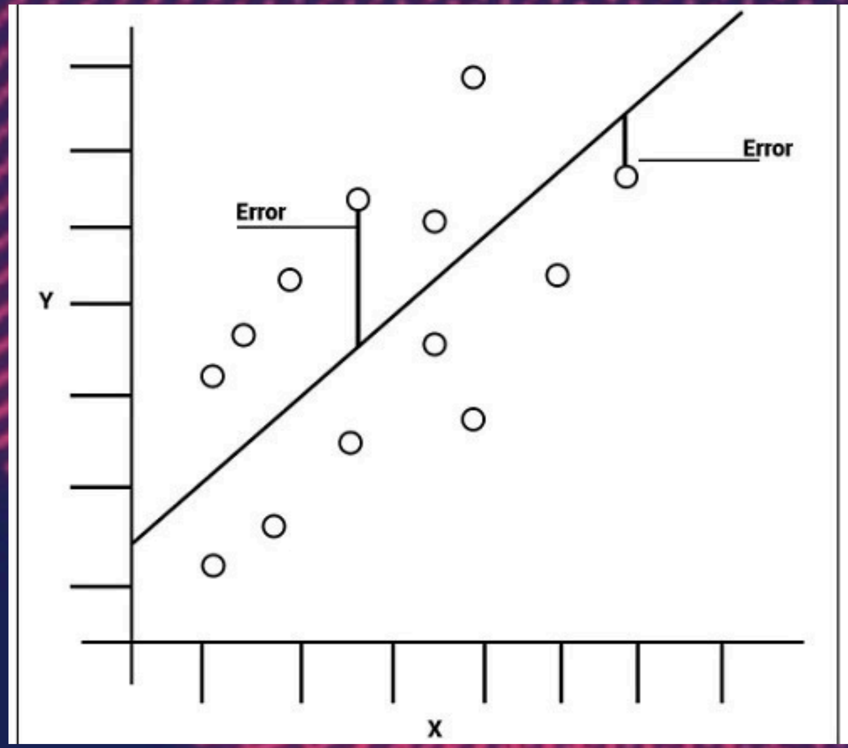
For this we need a quantity to determine how good or poor our model is at predicting output values for various inputs. This quantity is called the Cost function. Cost function quantifies the error between the value predicted by the model and the true output values.

Examples of cost function are

- Mean Square Error (MSE)
- Mean Absolute Error
- Binary Cross Entropy
- Categorical Cross Entropy

The cost function generally used for Linear Regression in MSE. The latter two are used for classification problems.

MEAN SQUARE ERROR



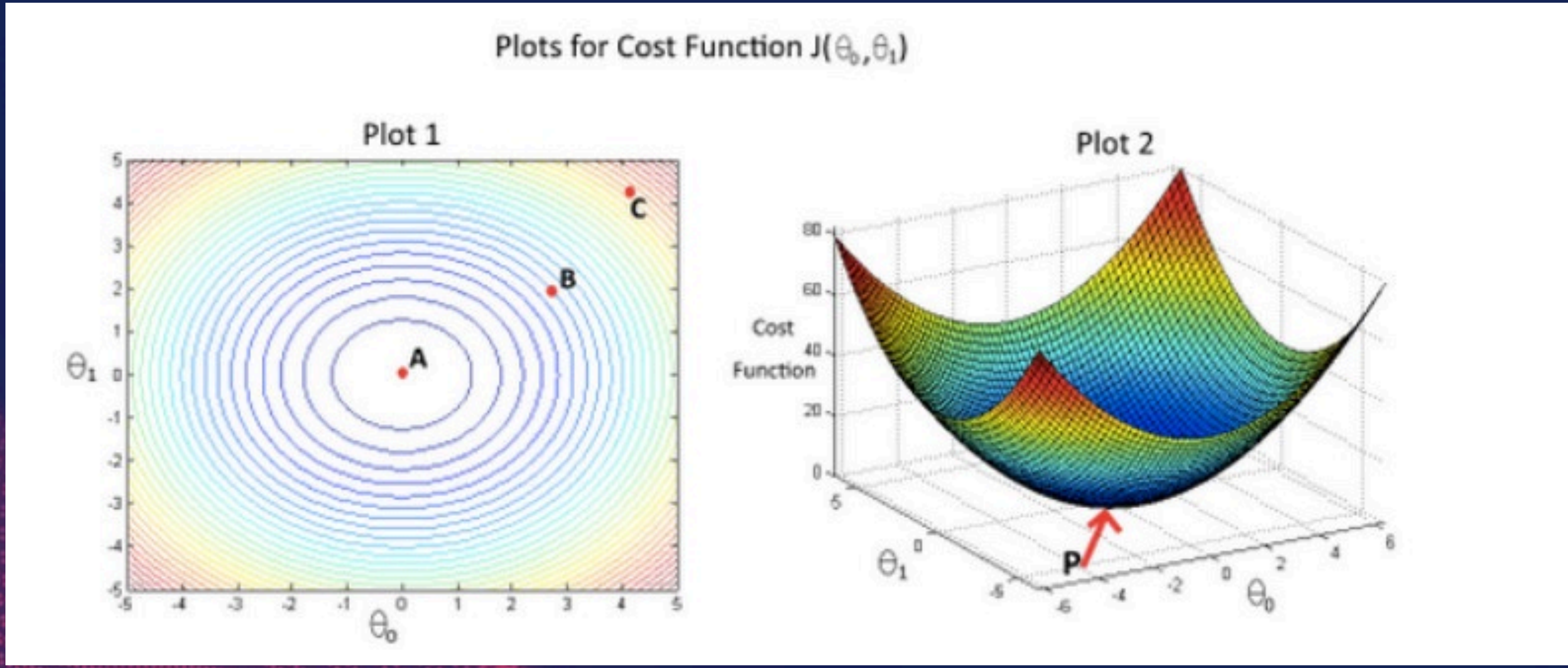
$$J(w,b) = \frac{1}{2m}\sum_{i=1}^{m}(\hat{y}_i - y_i)^2$$

$$\hat{y} = wx + b$$

- $\hat{y}_i$ is the predicted value for the i-th sample x_i in the training dataset
- y_i is the target value of the i-th sample x_i
- m is the number of training samples
- $J(w,b)$ is the Mean Sqaure Error cost function

Note that the cost function depends only on the parameters w and b for a given dataset

Plot of MSE as a function of the parameters w and b



The plot obtained is a 3-Dimensional Paraboloid surface with a single minima which is the global minima.

Now that we have Mean Square Error as our cost function for our linear regression algorithm, How do we proceed to minimize it ?

OPTIMIZATION

Optimization in the context of machine learning is about adjusting parameters in the model to minimize the cost function, thereby improving the accuracy and performance of the model

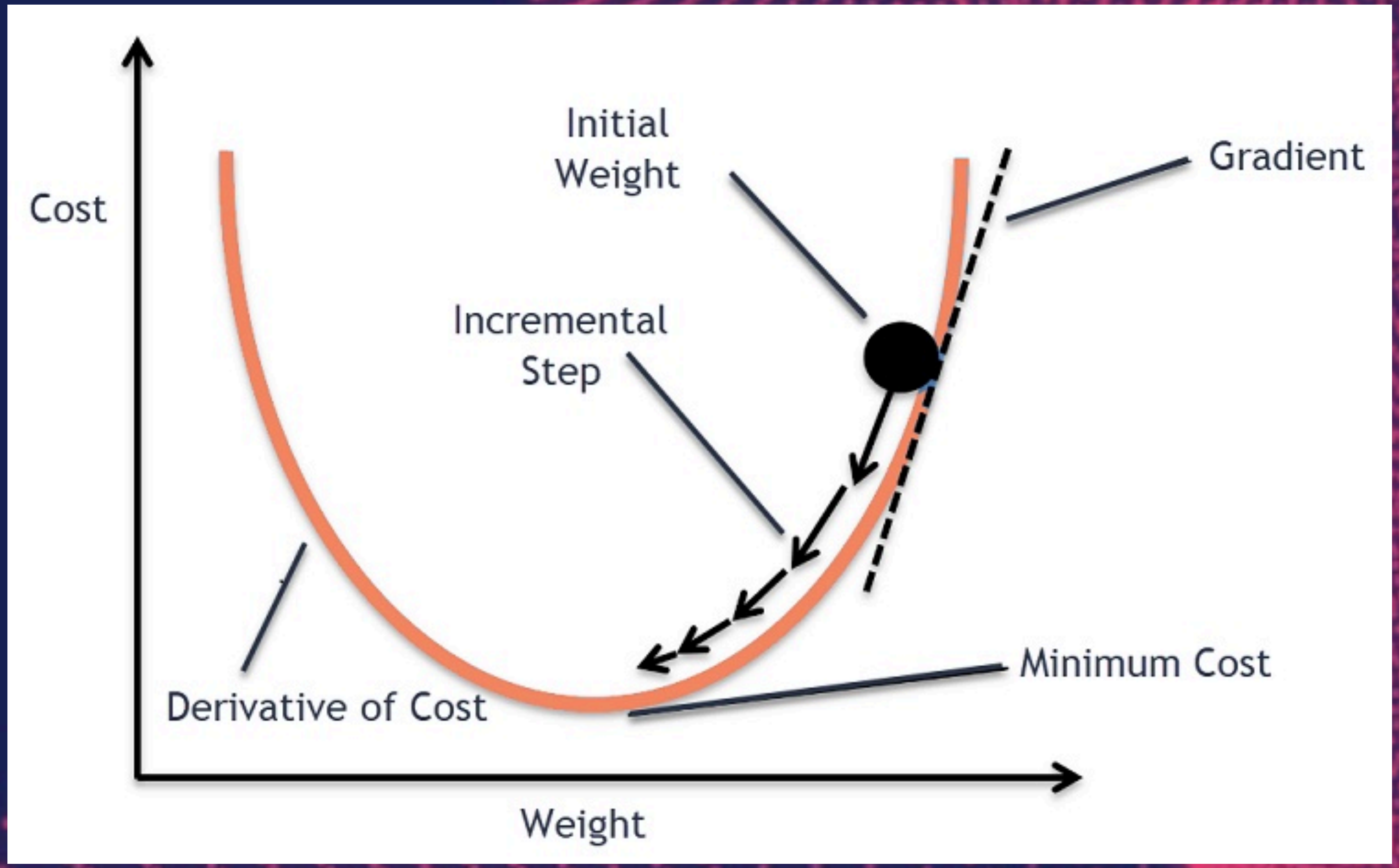
Examples of optimization algorithms are Gradient Descent, RMSProp, Adam etc.

For finding the optimal parameters w and b in our linear regression problem that minimizes the MSE we will be using the Gradient Descent Algorithm

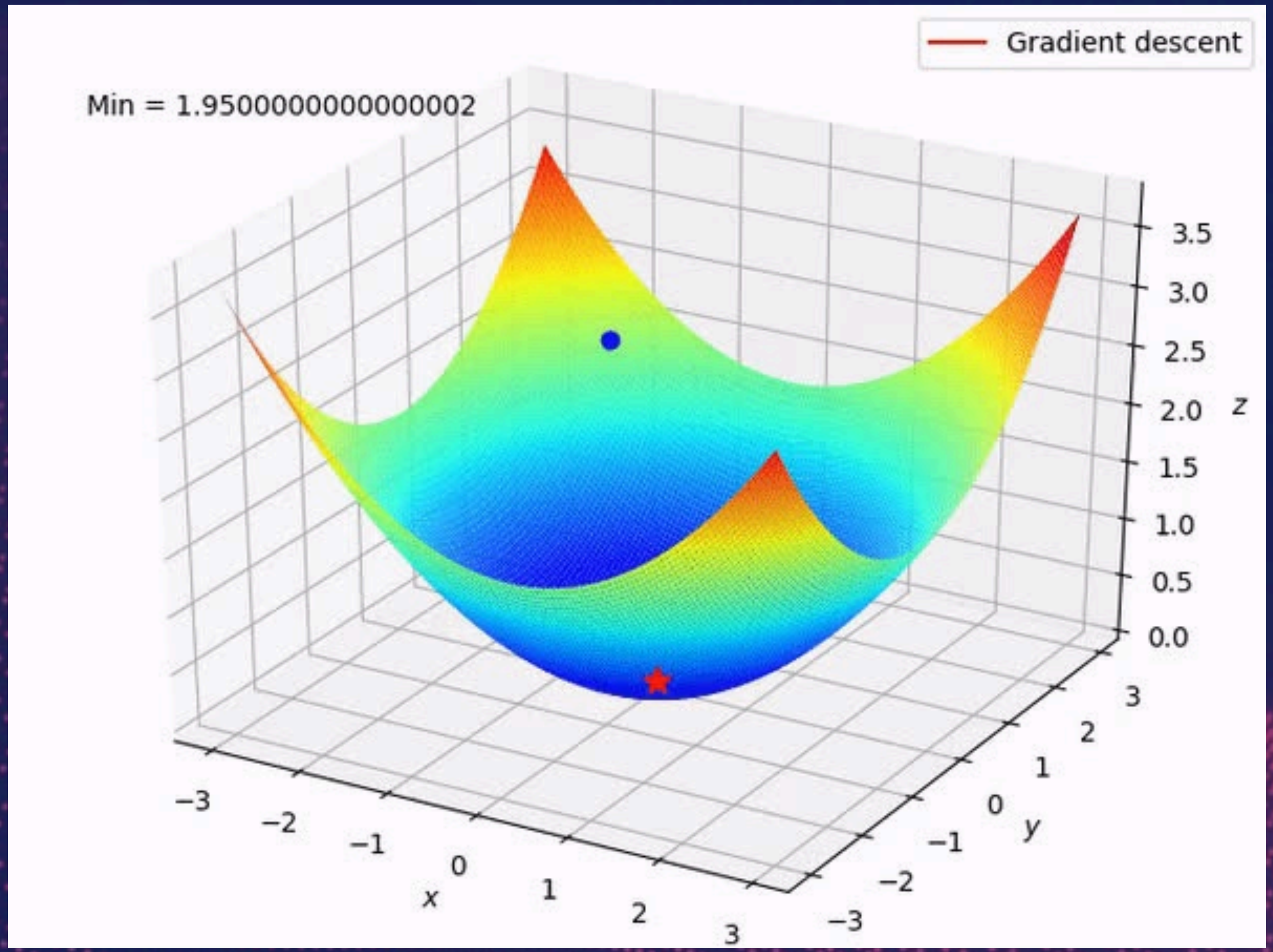
GRADIENT DESCENT ALGORITHM

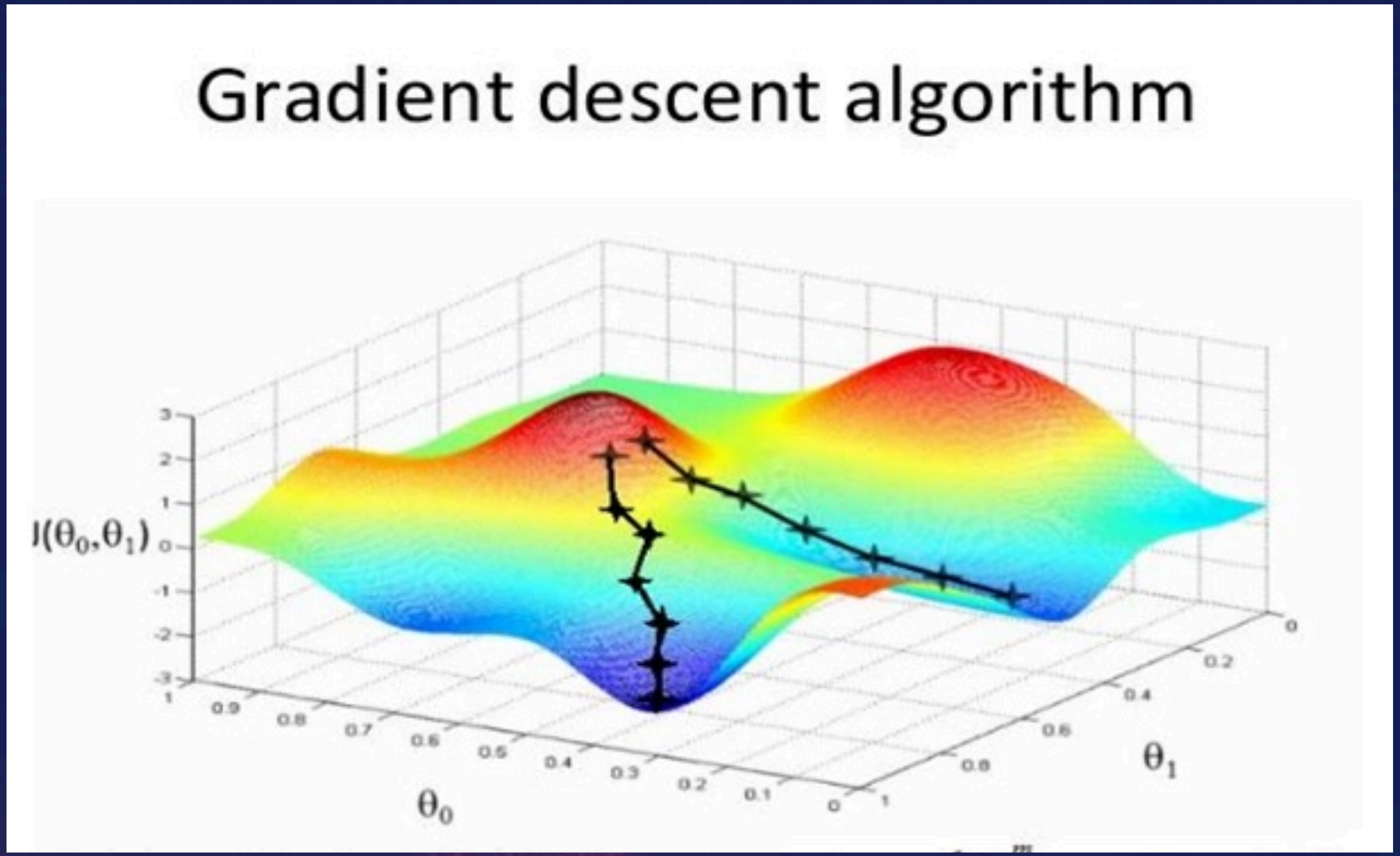
Gradient Descent is an optimization algorithm in which we try to reach the minima of the cost function by iteratively moving in the direction of steepest descent

During each iteration, we compute the gradient at the current point. Since the gradient gives the direction of steepest ascent, we move in the opposite direction with a step size α to reach the minimum. This process is repeated till we converge to the global minimum.



GRADIENT DESCENT





GRADIENT DESCENT IMPLEMENTATION

- The cost function MSE is given by

$$J(w,b) = \frac{1}{2m}\sum_{i=1}^{m}(\hat{y}_i - y_i)^2 \qquad \hat{y} = wx + b$$

- Computing the gradient at the current point

$$\frac{\partial J(w,b)}{\partial w} = \frac{1}{m}\sum_{i=1}^{m}(\hat{y}_i - y_i)x_i \qquad \frac{\partial J(w,b)}{\partial b} = \frac{1}{m}\sum_{i=1}^{m}(\hat{y}_i - y_i)$$

- Updating the weight and the bias

$$w = w - \alpha\frac{\partial J(w,b)}{\partial w} \qquad b = b - \alpha\frac{\partial J(w,b)}{\partial b}$$

$$w = w - \alpha\frac{1}{m}\sum_{i=1}^{m}((wx_i + b) - y_i)x_i$$

$$b = b - \alpha\frac{1}{m}\sum_{i=1}^{m}((wx_i + b) - y_i)$$

These steps are sequentially repeated till convergence is achieved

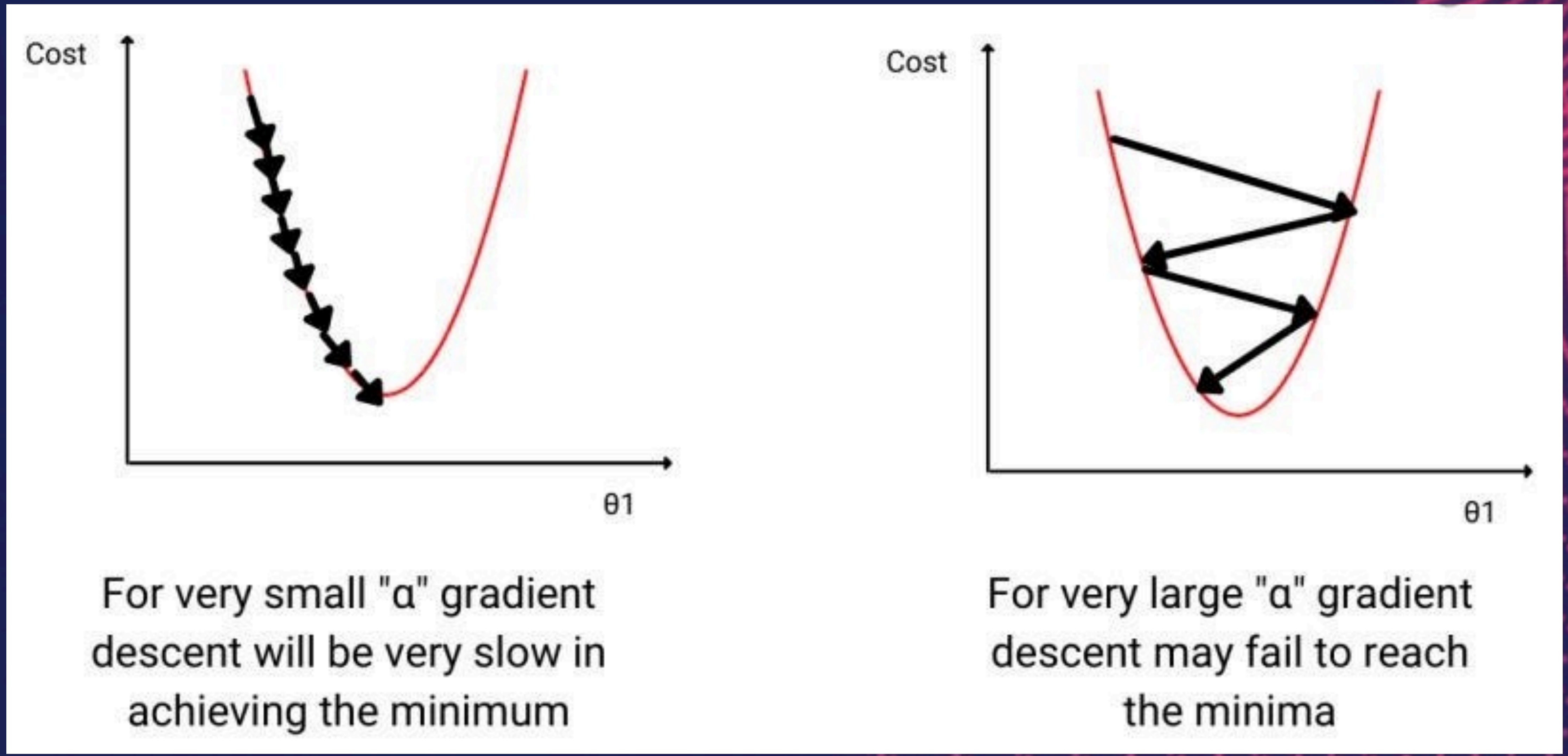
QUIZ TIME

Slido.com

4162721

LEARNING RATE α

It is a hyperparameter used in optimization algorithms that refer to the rate at which the model learns from the training data. In the context of gradient descent, It is the size of the steps taken to converge to the global minimum.



How do we make sure that learning rate α is optimal ?

Well, we plot the Cost function with the number of iterations. This plot is called the learning curve. If the chosen α is optimal then the cost function should decrease after every iteration. If cost function increases after a single iteration , it means the chosen α is high

MULTIPLE VARIABLE LINEAR REGRESSION

It is just an extension of simple linear regression using multiple independent variables which aims to model the relationship between multiple input features and an output target variable by fitting a linear equation to the training data.

$$\hat{y} = w_0 + w_1x_1 + w_2x_2 + \cdots + w_nx_n$$

Considering the same example of Housing price prediction,
Housing prices depend not just on size but also on many other factors.
Multiple variable linear regression helps in accomodating multiple input features and hence it is one of the most widely used machine learning algorithms even today.

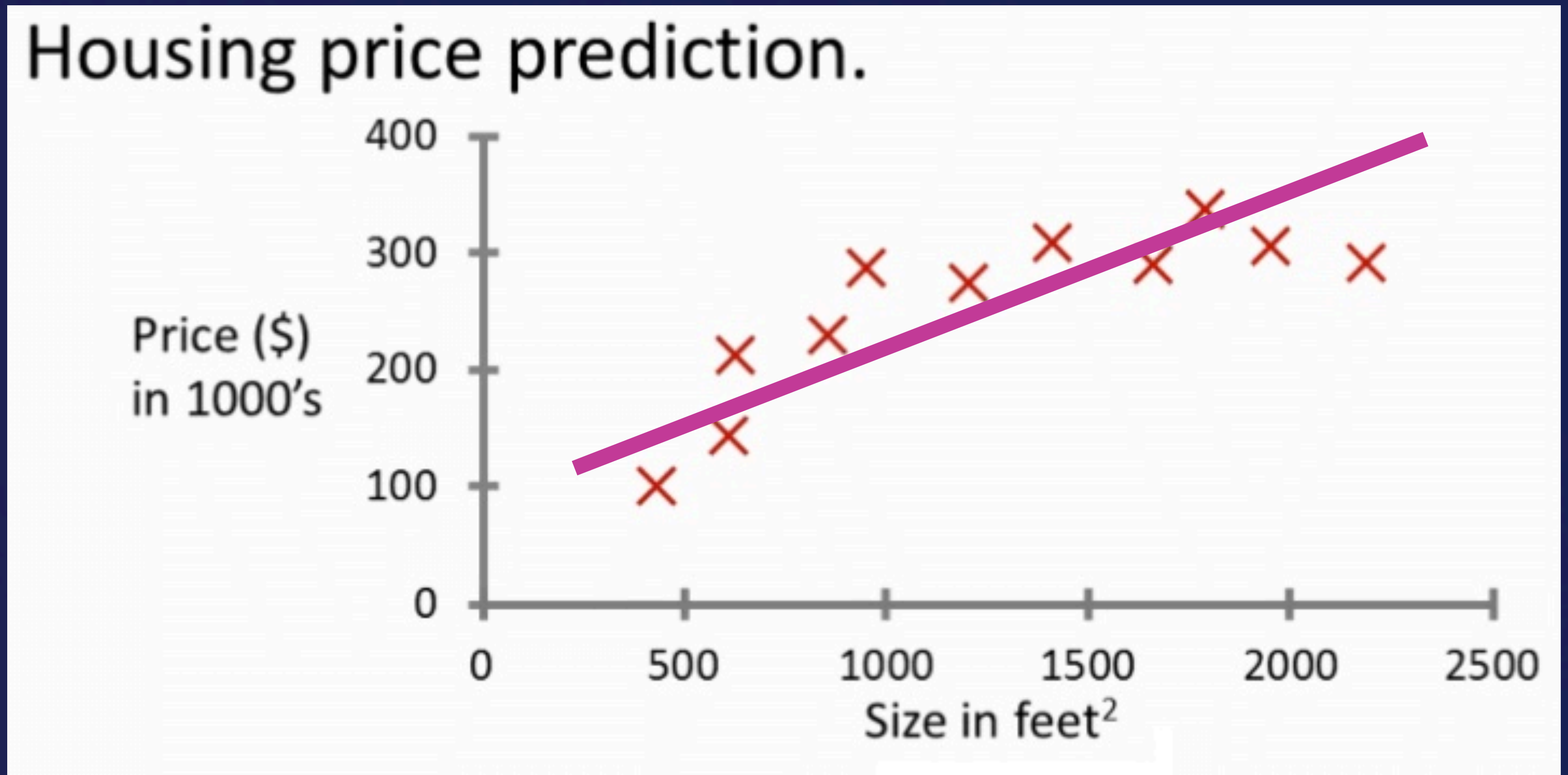
CODE IMPLEMENTATION

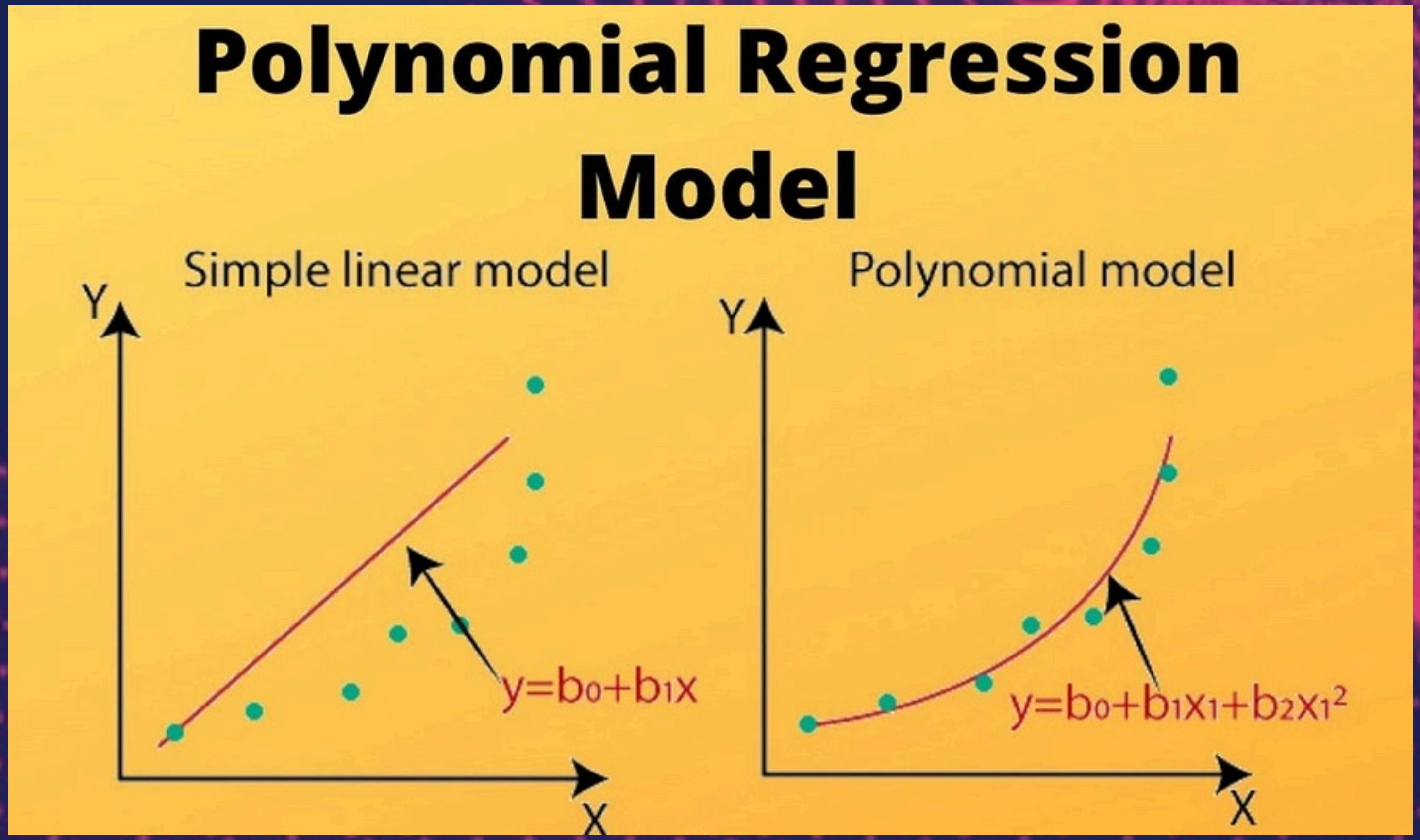
POLYNOMIAL REGRESSION

Polynomial Regression is a regression algorithm that models the relationship between output $\hat{y}$ and input features x as nth degree polynomial

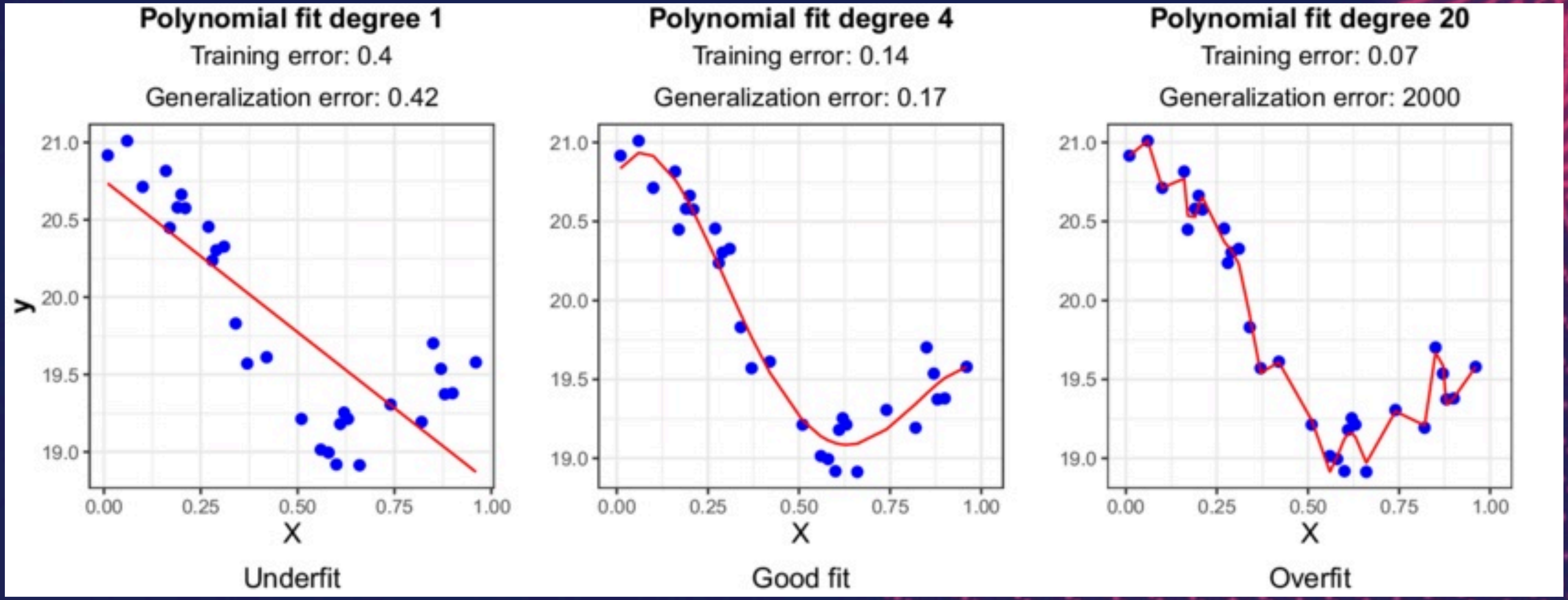
$$\hat{y} = w_0 + w_1 x + w_2 x^2 + w_3 x^3 + \cdots + w_n x^n$$

It can be considered as a special case Multiple variable Regression with the higher order terms representing various input features. Polynomial regression helps in capturing non linear relationships in the data which linear regression fails to do.

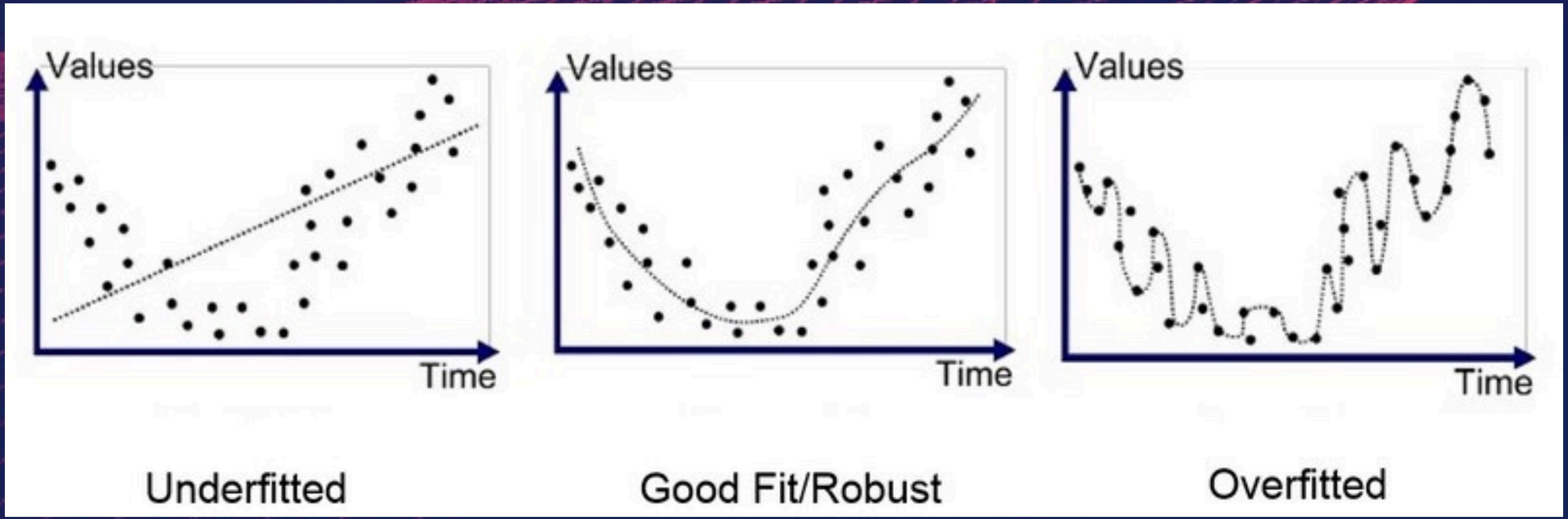




Polynomial regression helps to bring in more features into consideration ie. the higher order terms, but this gives rise to some problems like overfitting and underfitting



OVERFITTING AND UNDERFITTING



As it can be observed from the leftmost graph, straight line is clearly not the best fit for the given data. On adding a quadratic feature, we get the second graph which is the robust fit for the given data. But adding too many features can be dangerous. In such a case our model might fit the training data extremely well but would fail in predicting output for the testing data as evident from the right most graph.

OVERFITTING AND UNDERFITTING

Underfitting

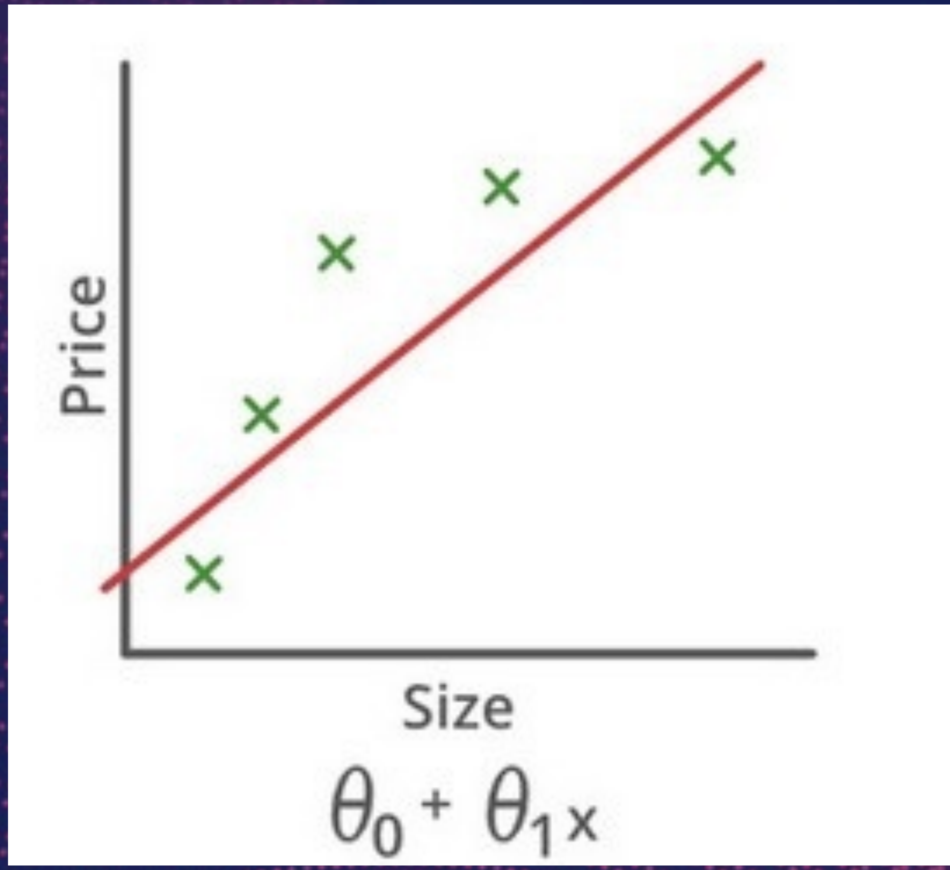
Underfitting occurs when the model is too simple to capture the underlying pattern in the data. This usually happens when the degree of the polynomial is too low.

Overfitting

Overfitting occurs when the model is too complex and captures not only the underlying pattern but also the noise in the data. This happens when the degree of the polynomial is too high.

BIAS

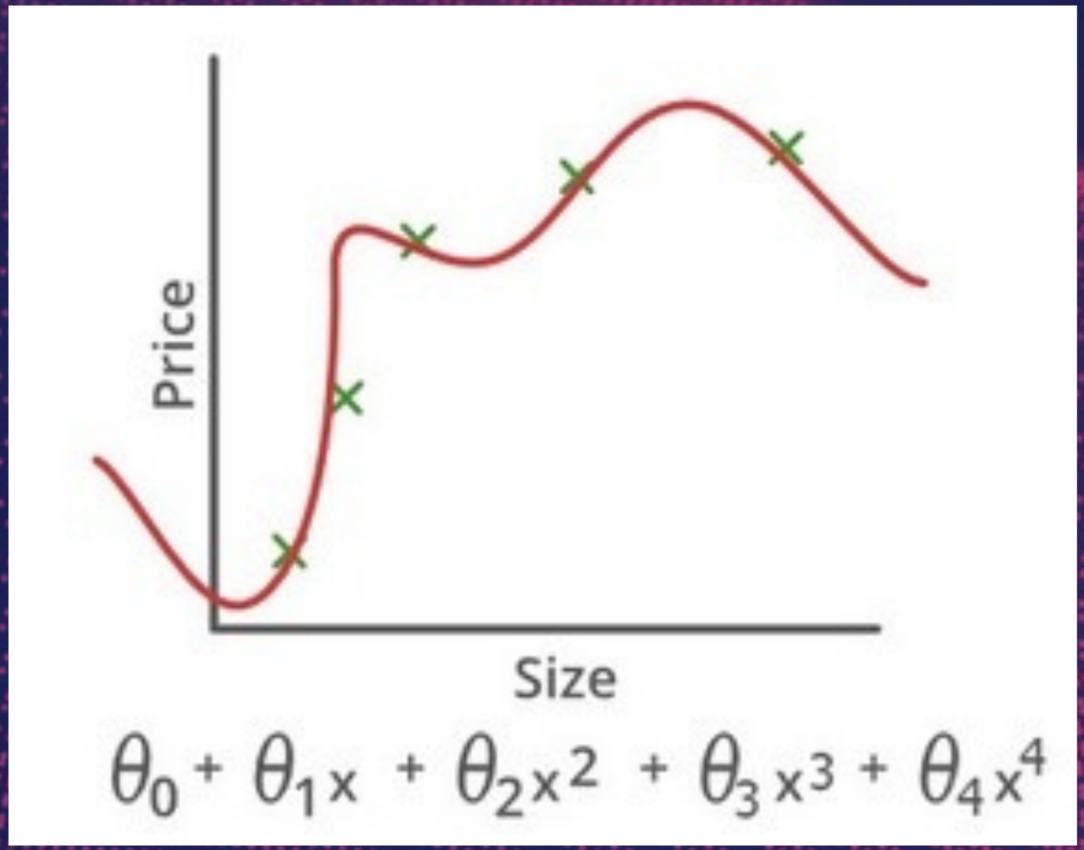
Bias is the error that arises when the chosen model or algorithm is too simple to handle the complexity of a problem.



A high bias means that the model is too simple , hence it is not able to capture important features or patterns from the dataset. This leads to under-fitting.
For example: when we apply linear regression to a non linear dataset as shown in the figure

VARIANCE

Variance refers to the error that occurs when a complex model which attempts to incorporate too many features is applied to a dataset. This complexity makes the model highly sensitive to fluctuations in the training data.



A high variance means that the model passes through most of the data points and it results in over-fitting. The model in this case learns the training data too well but performs poorly on testing data.

QUIZ TIME

Slido.com

2056392

BIAS - VARIANCE TRADEOFF

High bias and low variance leads to underfitting.
High variance and low bias leads to overfitting.

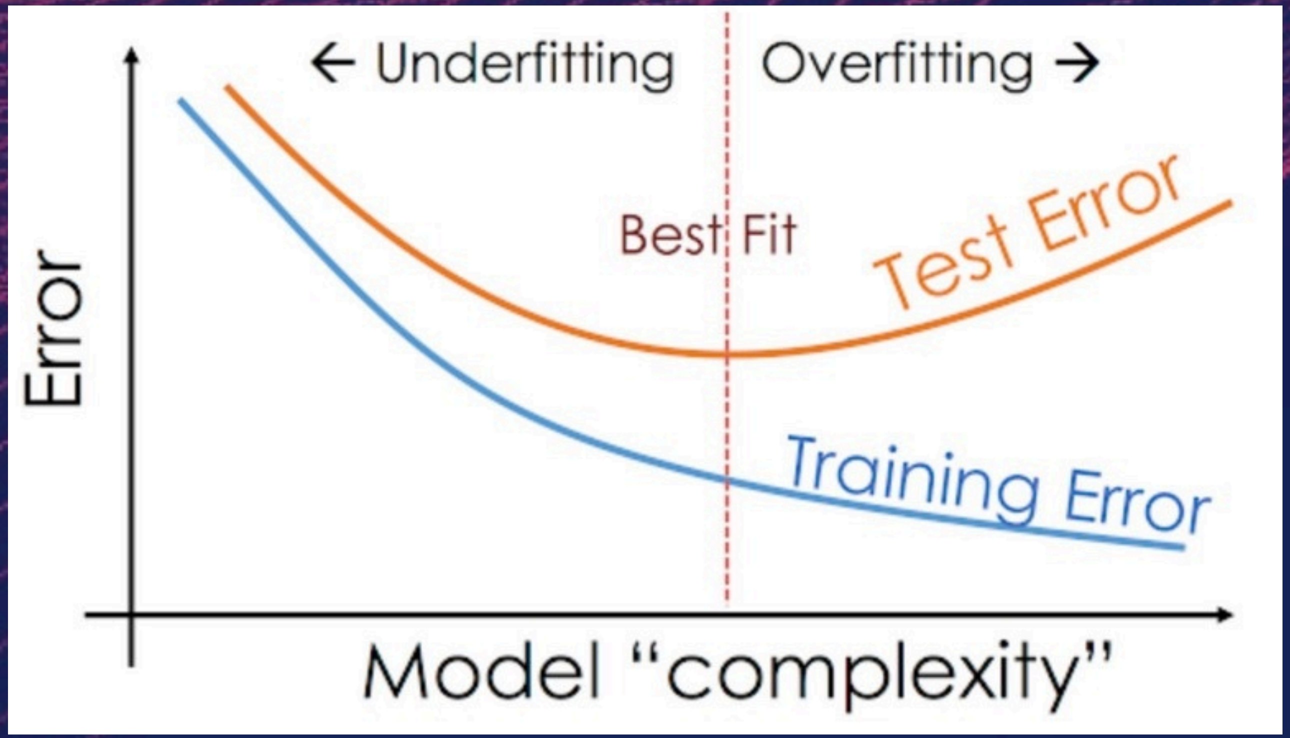
So what’s the ideal scenario?

Low bias and low variance

This is when the model is successful in capturing the features and patterns in the data avoiding overfitting as well as underfitting.

This brings us to the necessity of optimizing bias and variance

OPTIMIZING BIAS AND VARIANCE



The idea is to plot the cost function for every degree of x for the testing data . The minima thus found is the optimal order of the polynomial in order to balance bias and variance.

Attendance QR

CODE IMPLEMENTATION

THANK YOU