# Machine Learning Tutorial for Beginners

I have already covered the higher level concepts in of Machine Learning in my blog [What is Machine Learning?](https://www.edureka.co/blog/what-is-machine-learning/) As promised here is a tutorial blog in the series, titled “Machine Learning Tutorial”. This tutorial blog will help you to understand about:

* [Understanding Machine Learning with an Analogy](https://www.edureka.co/blog/machine-learning-tutorial/#MachineLearning)
* [What is Machine Learning?](https://www.edureka.co/blog/machine-learning-tutorial/#WhatIsMachineLearning)
* [Biggest Confusion AI vs ML vs DeepLearning](https://www.edureka.co/blog/machine-learning-tutorial/#AIvsMLvsDL)
* [Understanding Supervised Learning](https://www.edureka.co/blog/machine-learning-tutorial/#SupervisedLearning)
* [Understanding Unsupervised Learning](https://www.edureka.co/blog/machine-learning-tutorial/#UnsupervisedLearning)
* [Understanding Reinforcement Learning](https://www.edureka.co/blog/machine-learning-tutorial/#ReinforcementLearning)
* [Machine Learning Using Python](https://www.edureka.co/blog/machine-learning-tutorial/#MachineLearningUsingPython)

Understanding Machine Learning with an Analogy

**As a Human:** Let’s suppose one day you went for shopping mangoes. The vendor had a cart full of mangoes from where you could handpick the mangoes, get them weighed and pay according to the rate fixed per Kg.

****

**Task: How will you choose the best mangoes?**

Given below is set of learning, human gains from his experience of shopping mangoes, you can drill it down to have a further look at it in detail. Go through it once, you will relate it to machine learning very easily.

Learning 1: Bright yellow mangoes are sweeter than pale yellow ones

Learning 2: The smaller and bright yellow mangoes are sweet only half the time

Learning 3: Small, pale yellow ones are the sweetest of all

Learning 4: Soft mangoes are jucier

Learning 5: Green mangoes are tastier than yellow ones

Learning 6: You don't need mangoes anymore

What if you have to write a code for it?

**As a Human Written Code:** Now, imagine you were asked to write a computer program to choose your mangoes (or oranges). You might write the following rules/algorithm:

if is bright yellow **and** size is big **and** sold by: mango is sweet.  
if (soft): mango is juicy

You would use these rules to choose the mangoes.

**Conclusion as a human:**

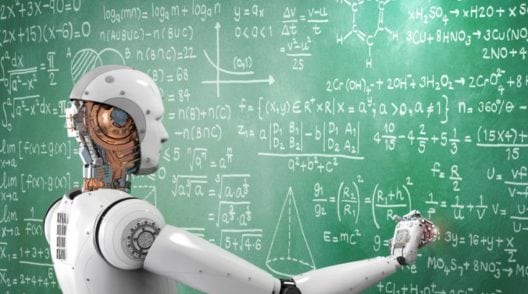
But every time you make a new observation from your experiments, you have to modify the list of rules manually.

You have to understand the details of all the factors affecting the quality of mangoes. If the problem gets complicated enough, it might get difficult for you to make accurate rules by hand that covers all possible types of mangoes. This will take a lot of research and effort and not everyone has this amount of time.

This is where Machine Learning comes into the picture

Machine Learning Tutorial: What is Machine Learning?

**Definition:**Machine Learning is a concept which allows the machine to learn from examples and experience, and that too without being explicitly programmed. So instead of you writing the code, what you do is you feed data to the generic algorithm, and the algorithm/machine builds the logic based on the given data.

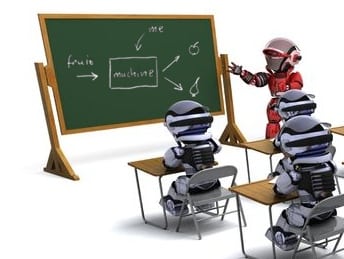


Machine Learning algorithms are an evolution of normal algorithms. They make your programs “smarter”, by allowing them to automatically learn from the data you provide. The algorithm is mainly divided into:

* Training Phase
* Testing phase

**Training Phase**

You take a randomly selected specimen of mangoes from the market (**training data**), make a table of all the physical characteristics of each mango, like color, size, shape, grown in which part of the country, sold by which vendor, etc (**features**), along with the sweetness, juiciness, ripeness of that mango (**output variables**). You feed this data to the machine learning algorithm (**classification/regression**), and it learns a model of the correlation between an average mango’s physical characteristics, and its quality.





**Testing Phase**

Next time when you go shopping, you will measure the characteristics of the mangoes which you are purchasing(**test data**)and feed it to the Machine Learning algorithm. It will use the model which was computed earlier to predict if the mangoes are sweet, ripe and/or juicy. The algorithm may internally use the rules, similar to the one you manually wrote earlier (for eg, a **decision tree**). Finally, you can now shop for mangoes with great confidence, without worrying about the details of how to choose the best mangoes.

**Conclusion as an Algorithm**

You know what! you can make your algorithm improve over time (**reinforcement learning**) so that it will improve its accuracy as it gets trained on more and more training dataset. In case it makes a wrong prediction it will update its rule by itself.

The best part of this is, you can use the same algorithm to train different models. You can create one each for predicting the quality of apples, grapes, bananas, or whatever you want.

For more detailed explanation on Machine Learning Tutorial feel free to go through this video:

**Machine Learning Tutorial | Machine Learning Algorithms | Edureka**

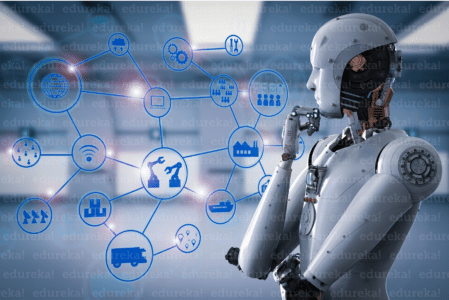
Subscribe to our YouTube channel to stay updated with our fresh content

Biggest Confusion AI vs ML vs Deep Learning

Let’s move ahead in this Machine Learning tutorial and discuss one of the biggest confusion. People think all three of them the AI, ML and the Deep Learning are same. But this is WRONG!, let me clarify things for you.

Artificial Intelligence

Artificial Intelligence is the broader concept of machines being able to carry out tasks in a smarter way. It covers anything which enables the computers to behave like humans.



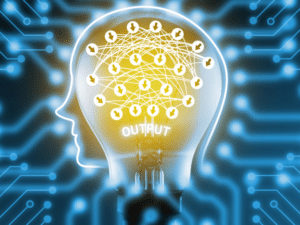


Machine Learning

Machine Learning is a subset of AI and is based on the idea that machines should be given the access to data, and should be left to learn and explore for themselves. It deals with the extraction of patterns from a large data sets.

Deep Learning

Deep Learning is a subset of Machine Learning where similar Machine Learning Algorithms are used to train Deep Neural Networks so as to achieve better accuracy in those cases where former was not performing up to the mark.



[Learn machine learning from THE experts](https://www.edureka.co/machine-learning-certification-training)

Let’s categorize Machine Learning into its subparts and see what each of them are, how they work, and how each one of them is used in the real life.

Starting with Supervised Learning, So what is it?

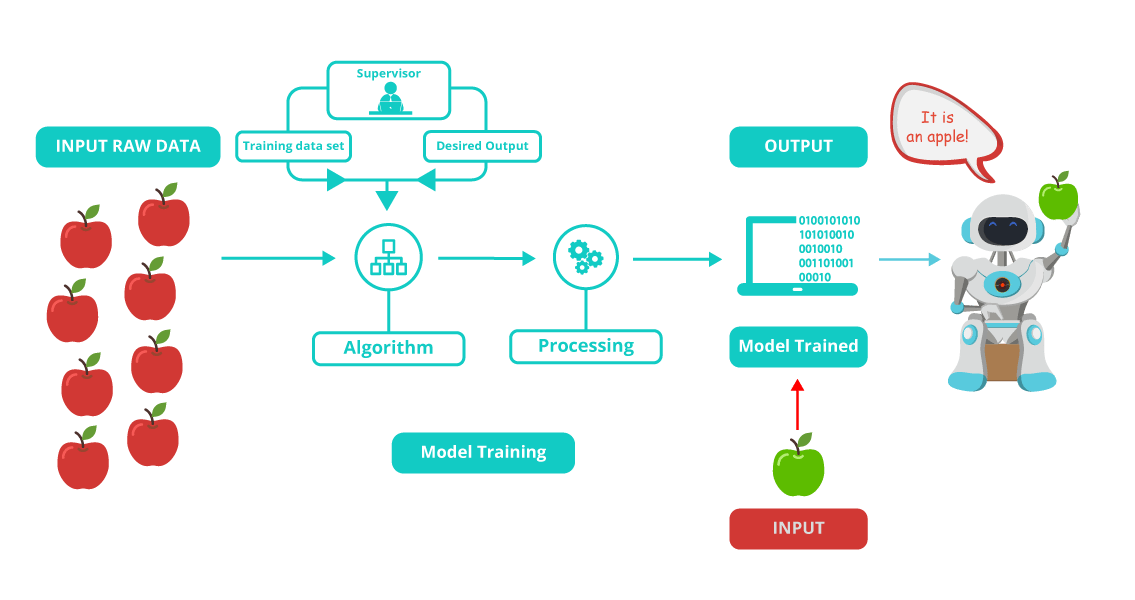
Understanding Supervised Learning

Let’s see the mathematical definition of Supervised Learning.

Supervised learning is the one where you have input variables (x) and an output variable (Y) and you use an algorithm to learn the mapping function from the input to the output. it,

Y = f(X)

The goal is to approximate the mapping function so well that whenever you get some new input data (x), the machine can easily predict the output variables (Y) for that data.

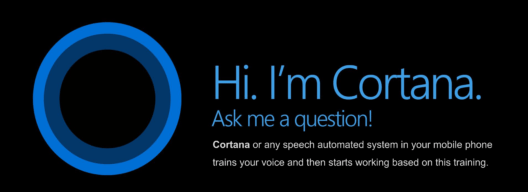
**Let me rephrase you this in simple terms:**

In Supervised machine learning algorithm, every instance of the training dataset consists of input attributes and expected output. The training dataset can take any kind of data as an input like**values of a database row, the pixels of an image, or even an audio frequency histogram.**

**Now let me tell you why this category of machine learning is termed as supervised learning?**

This category is termed as supervised learning because the process of an algorithm learning from the training dataset can be thought of as a **teacher teaching his students**. The algorithm continuously predicts the result on the basis of training data and is continuously corrected by the teacher. The learning continues until the algorithm achieves an acceptable level of performance.

Supervised Learning Usecases



Cortana

**Cortana** or any speech automated system in your mobile phone trains your voice and then starts working based on this training. This is an application of Supervised Learning

Weather Apps

Predicts the upcoming weather by analyzing the parameters for a given time on some prior knowledge (when its sunny, temperature is higher; when its cloudy, humidity is higher, etc.).





Biometric Attendance

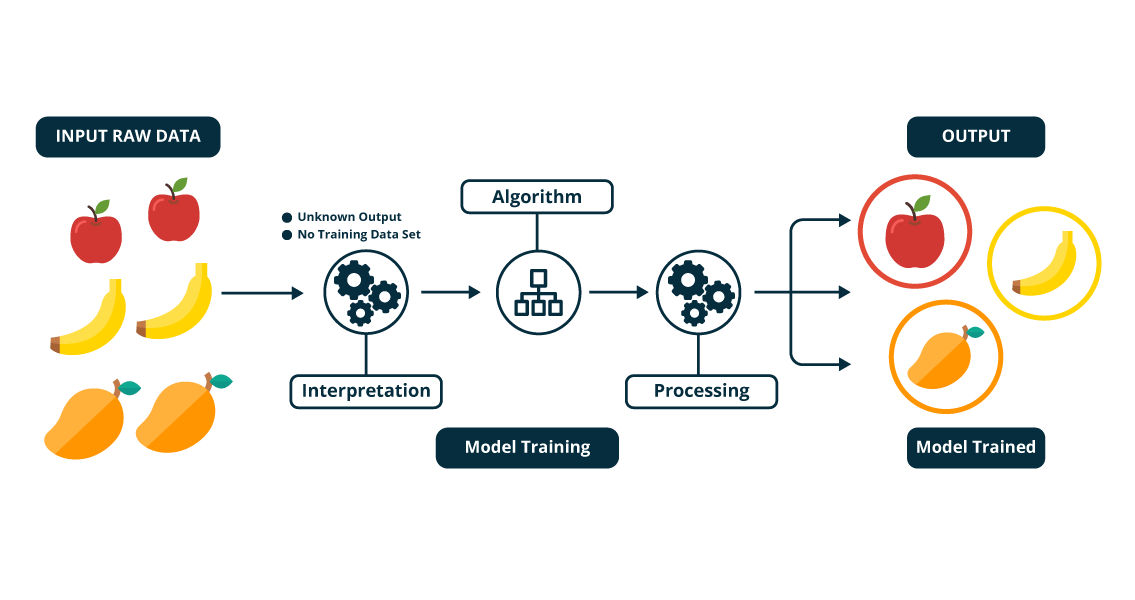
In **Biometric Attendance** you can train the machine with inputs of your biometric identity – it can be your thumb, iris or ear-lobe, etc. Once the machine is trained it can validate your future input and can easily identify you.

Understanding Unsupervised Learning

**So What is Unsupervised Learning?**

**Mathematically,** Unsupervised learning is where you only have input data (X) and no corresponding output variables.

The goal for unsupervised learning is to model the underlying structure or distribution in the data in order to learn more about the data.

**Let me rephrase it for you in simple terms:**

In the unsupervised learning approach, the sample of a training dataset does not have an expected output associated with them. Using the unsupervised learning algorithms you can detect patterns based on the typical characteristics of the input data. Clustering can be considered as an example of machine learning task that uses the unsupervised learning approach. The machine then groups similar data samples and identify different clusters within the data.

**Now let me tell you why this category of machine learning is known as unsupervised learning?**

Well, this category of machine learning is known as unsupervised because unlike supervised learning there is no teacher. Algorithms are left on their own to discover and return the interesting structure in the data.

Looking for a complete Machine Learning course?[Enroll Now](https://www.edureka.co/python)

Unsupervised Learning Usecases



A friend invites you to his party where you meet totally strangers. Now you will classify them using unsupervised learning (no prior knowledge) and this classification can be on the basis of gender, age group, dressing, educational qualification or whatever way you would like. Since you didn’t have any prior knowledge about people and so you just classified them “on-the-go”.

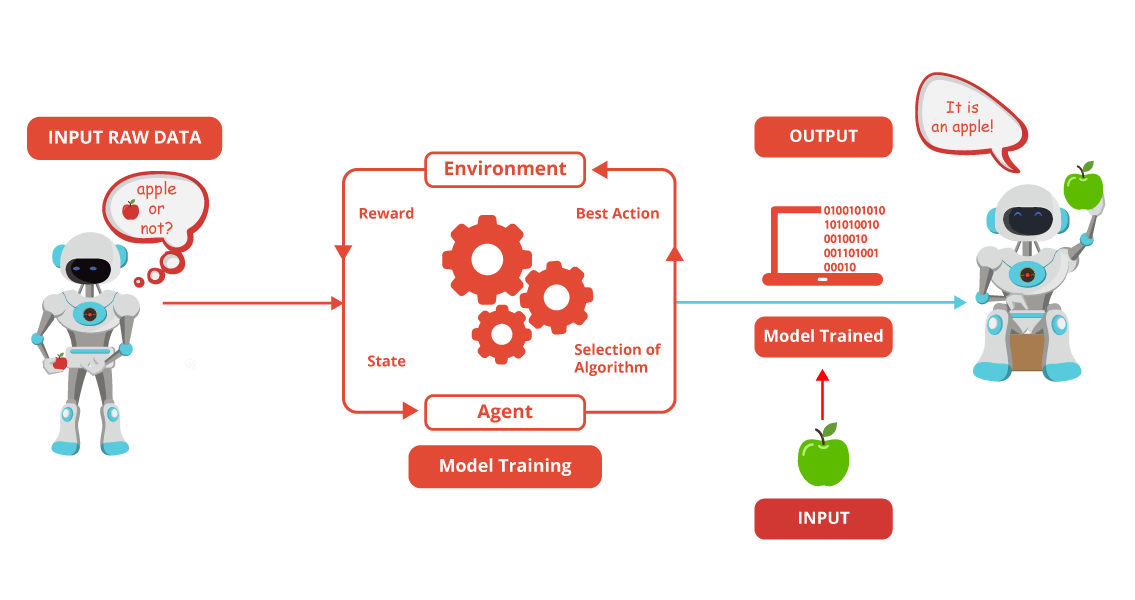
Let’s suppose you have never seen a Football match before and by chance watch a video on internet, now you can classify players on the basis of different criterion like Players wearing the same sort of kits are in one class, Players of one style are in one class (players, goalkeeper, referee), or on the basis of playing style(attacker or defender) or whatever way you would observe, you can classify it.



Understanding Reinforcement Learning

**So what is Reinforcement Learning?**

Reinforcement learning can be thought of as a hit and trial method of learning. The machine gets a Reward or Penalty point for each action it performs. If the option is correct, the machine gains the reward point or gets a penalty point in case of a wrong response.

The reinforcement learning algorithm is all about the interaction between the environment and the learning agent. The learning agent is based on exploration and exploitation.

Exploration is when the learning agent acts on trial and error and Exploitation is when it performs an action based on the knowledge gained from the environment. The environment rewards the agent for every correct action, which is the reinforcement signal. With the aim of collecting more rewards obtained, the agent improves its environment knowledge to choose or perform the next action.

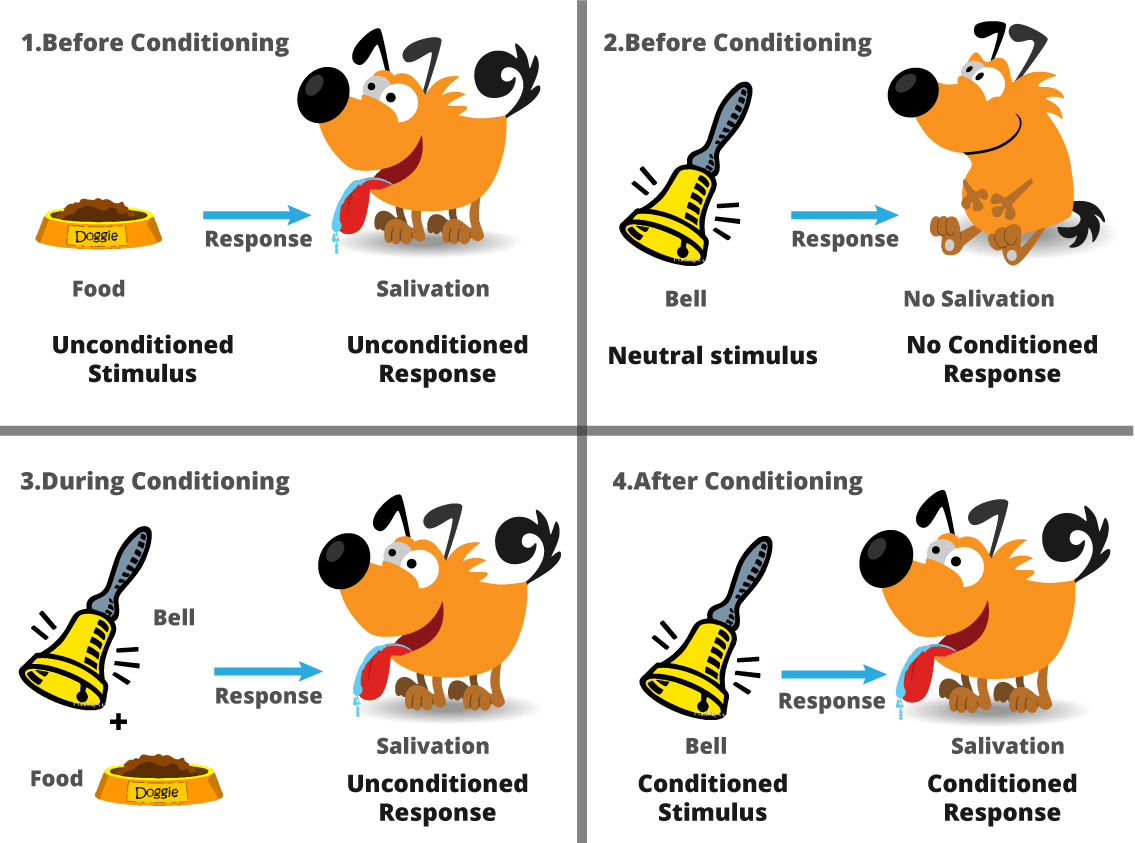
**Let see how Pavlov trained his dog using reinforcement training?**

Pavlov divided the training of his dog into four stages.

In the first part, Pavlov gave meat to the dog, and in response to the meat, the dog started salivating.

In the next stage he created a sound with a bell, but this time the dogs did not respond anything.

In the third stage, he tried to train his dog by using the bell and then giving them food. Seeing the food the dog started salivating.

Eventually, the dogs started salivating just after hearing the bell, even if the food was not given as the dog was reinforced that whenever the master will ring the bell, he will get the food.

Machine Learning Using Python

On a new tool, it is always good to start with a small project. For example, in this case, classification of iris flowers on [the iris dataset](https://archive.ics.uci.edu/ml/datasets/Iris).

It’s a good project and is really very easy to understand.

* All the attributes within the dataset are numeric, you just have to figure out how to load and handle data
* It is a multi-class classification problem thereby allowing you to practice the supervised learning algorithm
* 4 attributes and 150 rows, meaning it is small and easily fits into memory
* All of the numeric attributes are in the same units and the same scale, not requiring any special scaling or transforms to get started

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99  100  101  102  103  104  105  106  107  108  109  110  111  112  113  114 | # Check the versions of libraries  # Python version  import sys  print('Python: {}'.format(sys.version))  # scipy  import scipy  print('scipy: {}'.format(scipy.\_\_version\_\_))  # numpy  import numpy  print('numpy: {}'.format(numpy.\_\_version\_\_))  # matplotlib  import matplotlib  print('matplotlib: {}'.format(matplotlib.\_\_version\_\_))  # pandas  import pandas  print('pandas: {}'.format(pandas.\_\_version\_\_))  # scikit-learn  import sklearn  print('sklearn: {}'.format(sklearn.\_\_version\_\_))    # Load libraries  import pandas  from pandas.plotting import scatter\_matrix  import matplotlib.pyplot as plt  from sklearn import model\_selection  from sklearn.metrics import classification\_report  from sklearn.metrics import confusion\_matrix  from sklearn.metrics import accuracy\_score  from sklearn.linear\_model import LogisticRegression  from sklearn.tree import DecisionTreeClassifier  from sklearn.neighbors import KNeighborsClassifier  from sklearn.discriminant\_analysis import LinearDiscriminantAnalysis  from sklearn.naive\_bayes import GaussianNB  from sklearn.svm import SVC    # Load dataset  url = https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data  names = ['sepal-length', 'sepal-width', 'petal-length', 'petal-width', 'class']  dataset = pandas.read\_csv(url, names=names)    # shape  print(dataset.shape)    # head  print(dataset.head(20))    # head  print(dataset.head(20))    # descriptions  print(dataset.describe())    # class distribution  print(dataset.groupby('class').size())    # box and whisker plots  dataset.plot(kind='box', subplots=True, layout=(2,2), sharex=False, sharey=False)  plt.show()    # histograms  dataset.hist()  plt.show()    # scatter plot matrix  scatter\_matrix(dataset)  plt.show()    # Split-out validation dataset  array = dataset.values  X = array[:,0:4]  Y = array[:,4]  validation\_size = 0.20  seed = 7  X\_train, X\_validation, Y\_train, Y\_validation = model\_selection.train\_test\_split(X, Y, test\_size=validation\_size, random\_state=seed)  # Test options and evaluation metric  seed = 7  scoring = 'accuracy'    # Spot Check Algorithms  models = []  models.append(('LR', LogisticRegression()))  models.append(('LDA', LinearDiscriminantAnalysis()))  models.append(('KNN', KNeighborsClassifier()))  models.append(('CART', DecisionTreeClassifier()))  models.append(('NB', GaussianNB()))  models.append(('SVM', SVC()))    # Spot Check Algorithms  models = []  models.append(('LR', LogisticRegression()))  models.append(('LDA', LinearDiscriminantAnalysis()))  models.append(('KNN', KNeighborsClassifier()))  models.append(('CART', DecisionTreeClassifier()))  models.append(('NB', GaussianNB()))  models.append(('SVM', SVC()))    # evaluate each model in turn  results = []  names = []  for name, model in models:      kfold = model\_selection.KFold(n\_splits=10, random\_state=seed)      cv\_results = model\_selection.cross\_val\_score(model, X\_train, Y\_train, cv=kfold, scoring=scoring)      results.append(cv\_results)      names.append(name)      msg = %s: %f (%f) % (name, cv\_results.mean(), cv\_results.std())      print(msg)    # Make predictions on validation dataset  knn = KNeighborsClassifier()  knn.fit(X\_train, Y\_train)  predictions = knn.predict(X\_validation)  print(accuracy\_score(Y\_validation, predictions))  print(confusion\_matrix(Y\_validation, predictions))  print(classification\_report(Y\_validation, predictions)) |

I hope this Machine Learning Tutorial blog was useful, feel free to visit our edureka’s youtube page where you will find tons of technical videos on several topics. If you want to excel in your career you can take [Machine Learning Certification Training using Python](https://www.edureka.co/python) from [edureka](http://www.edureka.co/). Don’t just blindly trust someone’s word, here is the link to the [Course Curriculum](https://www.edureka.co/python), have a look at it and see if you find it really worth investing your time and effort. If you want to learn something more, add your suggestion in the comment box below, If the suggestion is good enough we might even plan to add it to our main course for you.