

# Machine Learning with Python

Luis Carlos Rojas Torres

September 2021



# Chapter 1

## Week 1

### 1.1 What is Machine Learning?

#### 1.1.1 Welcome

Hello and welcome to Machine Learning with Python. In this course you'll learn how Machine Learning is used in many key fields and industries. For example, in the healthcare industry, data scientists use Machine Learning to predict whether a human cell that is believed to be at risk of developing cancer is either benign or malignant. As such, Machine Learning can play a key role in determining a person's health and welfare. You'll also learn about the value of decision trees and how building a good decision tree from historical data helps doctors to prescribe the proper medicine for each of their patients. You'll learn how bankers use Machine Learning to make decisions on whether to approve loan applications; and you will learn how to use Machine Learning to do bank customer segmentation, where it is not usually easy to run for huge volumes of varied data. In this course you'll see how machine learning helps websites such as YouTube, Amazon, or Netflix develop recommendations to their customers about various products or services, such as which movies they might be interested in going to see or which books to buy. There is so much that you can do with Machine Learning. Here you'll learn how to use popular Python libraries to build your model. For example, given an automobile data set, we can use the scikit-learn library to estimate the co2 emission of cars using their engine size or cylinders. We can even predict what the co2 emissions will be for a car that hasn't even been produced yet, and we'll see how the telecommunications industries can predict customer churn. You can run and practice the code of all these samples using the built-in lab environment in this course, you don't have to install anything to your computer or do anything on the cloud. All you have to do is click a button to start the lab environment in your browser. The code for the samples is already written using Python language in Jupyter notebooks and you can run it to see the results or change it to understand the algorithms better. So what will you be able to achieve by taking this course? Well, by putting in just a few hours a week over the next few weeks you'll get new skills to add to your resume such as regression, classification, clustering, scikit-learn, and psy PI. You'll also get new projects that you can add to your portfolio including cancer detection, predicting economic trends, predicting customer churn, rec-

ommendation engines, and many more. You'll also get a certificate in Machine Learning to prove your competency and share it anywhere you like online or offline such as LinkedIn profiles and social media, so let's get started.

### 1.1.2 Introduction to Machine Learning

Hello, and welcome! In this video I will give you a high level introduction to Machine Learning. So let's get started. This is a human cell sample extracted from a patient, and this cell has characteristics. For example, its clump thickness is 6, its uniformity of cell size is 1, its marginal adhesion is 1, and so on. One of the interesting questions we can ask, at this point is: Is this a benign or malignant cell? In contrast with a benign tumor, a malignant tumor is a tumor that may invade its surrounding tissue or spread around the body, and diagnosing it early might be the key to a patient's survival. One could easily presume that only a doctor with years of experience could diagnose that tumor and say if the patient is developing cancer or not. Right? Well, imagine that you've obtained a dataset containing characteristics of thousands of human cell samples extracted from patients who were believed to be at risk of developing cancer. Analysis of the original data showed that many of the characteristics differed significantly between benign and malignant samples. You can use the values of these cell characteristics in samples from other patients to give an early indication of whether a new sample might be benign or malignant. You should clean your data, select a proper algorithm for building a prediction model, and train your model to understand patterns of benign or malignant cells within the data. Once the model has been trained by going through data iteratively, it can be used to predict your new or unknown cell with a rather high accuracy. This is machine learning! It is the way that a machine learning model can do a doctor's task or at least help that doctor make the process faster. Now, let me give a formal definition of machine learning. Machine learning is the subfield of computer science that gives "computers the ability to learn without being explicitly programmed." Let me explain what I mean when I say "without being explicitly programmed." Assume that you have a dataset of images of animals such as cats and dogs, and you want to have software or an application that can recognize and differentiate them. The first thing that you have to do here is interpret the images as a set of feature sets. For example, does the image show the animal's eyes? If so, what is their size? Does it have ears? What about a tail? How many legs? Does it have wings? Prior to machine learning, each image would be transformed to a vector of features. Then, traditionally, we had to write down some rules or methods in order to get computers to be intelligent and detect the animals. But, it was a failure. Why? Well, as you can guess, it needed a lot of rules, highly dependent on the current dataset, and not generalized enough to detect out-of-sample cases. This is when machine learning entered the scene. Using machine learning, allows us to build a model that looks at all the feature sets, and their corresponding type of animals, and it learns the pattern of each animal. It is a model built by machine learning algorithms. It detects without explicitly being programmed to do so. In essence, machine learning follows the same process that a 4-year-old child uses to learn, understand, and differentiate animals. So, machine learning algorithms, inspired by the human learning process, iteratively learn from data, and allow computers to find hidden insights. These models help us in a variety of tasks, such as object recognition,

summarization, recommendation, and so on. Machine Learning impacts society in a very influential way. Here are some real-life examples. First, how do you think Netflix and Amazon recommend videos, movies, and TV shows to its users? They use Machine Learning to produce suggestions that you might enjoy! This is similar to how your friends might recommend a television show to you, based on their knowledge of the types of shows you like to watch. How do you think banks make a decision when approving a loan application? They use machine learning to predict the probability of default for each applicant, and then approve or refuse the loan application based on that probability. Telecommunication companies use their customers' demographic data to segment them, or predict if they will unsubscribe from their company the next month. There are many other applications of machine learning that we see every day in our daily life, such as chatbots, logging into our phones or even computer games using face recognition. Each of these use different machine learning techniques and algorithms. So, let's quickly examine a few of the more popular techniques. The Regression/Estimation technique is used for predicting a continuous value. For example, predicting things like the price of a house based on its characteristics, or to estimate the Co2 emission from a car's engine. A Classification technique is used for Predicting the class or category of a case, for example, if a cell is benign or malignant, or whether or not a customer will churn. Clustering groups of similar cases, for example, can find similar patients, or can be used for customer segmentation in the banking field. Association technique is used for finding items or events that often co-occur, for example, grocery items that are usually bought together by a particular customer. Anomaly detection is used to discover abnormal and unusual cases, for example, it is used for credit card fraud detection. Sequence mining is used for predicting the next event, for instance, the click-stream in websites. Dimension reduction is used to reduce the size of data. And finally, recommendation systems, this associates people's preferences with others who have similar tastes, and recommends new items to them, such as books or movies. We will cover some of these techniques in the next videos. By this point, I'm quite sure this question has crossed your mind, "What is the difference between these buzzwords that we keep hearing these days, such as Artificial intelligence (or AI), Machine Learning and Deep Learning?" Well, let me explain what is different between them. In brief, AI tries to make computers intelligent in order to mimic the cognitive functions of humans. So, Artificial Intelligence is a general field with a broad scope including: Computer Vision, Language Processing, Creativity, and Summarization. Machine Learning is the branch of AI that covers the statistical part of artificial intelligence. It teaches the computer to solve problems by looking at hundreds or thousands of examples, learning from them, and then using that experience to solve the same problem in new situations. And Deep Learning is a very special field of Machine Learning where computers can actually learn and make intelligent decisions on their own. Deep learning involves a deeper level of automation in comparison with most machine learning algorithms. Now that we've completed the introduction to Machine Learning, subsequent videos will focus on reviewing two main components: First, you'll be learning about the purpose of Machine Learning and where it can be applied in the real world; and Second, you'll get a general overview of Machine Learning topics, such as supervised vs unsupervised learning, model evaluation and various Machine Learning algorithms. So now that you have a sense with what's in store on this journey,

let's continue our exploration of Machine Learning! Thanks for watching!

### 1.1.3 Python for Machine Learning

Hello and welcome. In this video, we'll talk about how to use Python for machine learning. So let's get started. Python is a popular and powerful general purpose programming language that recently emerged as the preferred language among data scientists. You can write your machine-learning algorithms using Python, and it works very well. However, there are a lot of modules and libraries already implemented in Python, that can make your life much easier. We try to introduce the Python packages in this course and use it in the labs to give you better hands-on experience. The first package is NumPy which is a math library to work with N-dimensional arrays in Python. It enables you to do computation efficiently and effectively. It is better than regular Python because of its amazing capabilities. For example, for working with arrays, dictionaries, functions, datatypes and working with images you need to know NumPy. SciPy is a collection of numerical algorithms and domain specific toolboxes, including signal processing, optimization, statistics and much more. SciPy is a good library for scientific and high performance computation. Matplotlib is a very popular plotting package that provides 2D plotting, as well as 3D plotting. Basic knowledge about these three packages which are built on top of Python, is a good asset for data scientists who want to work with real-world problems. If you're not familiar with these packages, I recommend that you take the data analysis with Python course first. This course covers most of the useful topics in these packages. Pandas library is a very high-level Python library that provides high performance easy to use data structures. It has many functions for data importing, manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and timeseries. SciKit Learn is a collection of algorithms and tools for machine learning which is our focus here and which you'll learn to use within this course. As we'll be using SciKit Learn quite a bit in the labs, let me explain more about it and show you why it is so popular among data scientists. SciKit Learn is a free Machine Learning Library for the Python programming language. It has most of the classification, regression and clustering algorithms, and it's designed to work with a Python numerical and scientific libraries: NumPy and SciPy. Also, it includes very good documentation. On top of that, implementing machine learning models with SciKit Learn is really easy with a few lines of Python code. Most of the tasks that need to be done in a machine learning pipeline are implemented already in Scikit Learn including pre-processing of data, feature selection, feature extraction, train test splitting, defining the algorithms, fitting models, tuning parameters, prediction, evaluation, and exporting the model. Let me show you an example of how SciKit Learn looks like when you use this library. You don't have to understand the code for now but just see how easily you can build a model with just a few lines of code. Basically, machine-learning algorithms benefit from standardization of the dataset. If there are some outliers or different scales fields in your dataset, you have to fix them. The pre-processing package of SciKit Learn provides several common utility functions and transformer classes to change raw feature vectors into a suitable form of vector for modeling. You have to split your dataset into train and test sets to train your model and then test the model's accuracy separately. SciKit Learn can split arrays or matrices

into random train and test subsets for you in one line of code. Then you can set up your algorithm. For example, you can build a classifier using a support vector classification algorithm. We call our estimator instance CLF and initialize its parameters. Now you can train your model with the train set by passing our training set to the fit method, the CLF model learns to classify unknown cases. Then we can use our test set to run predictions, and the result tells us what the class of each unknown value is. Also, you can use the different metrics to evaluate your model accuracy. For example, using a confusion matrix to show the results. And finally, you save your model. You may find all or some of these machine-learning terms confusing but don't worry, we'll talk about all of these topics in the following videos. The most important point to remember is that the entire process of a machine learning task can be done simply in a few lines of code using SciKit Learn. Please notice that though it is possible, it would not be that easy if you want to do all of this using NumPy or SciPy packages. And of course, it needs much more coding if you use pure Python programming to implement all of these tasks. Thanks for watching.

#### 1.1.4 Supervised vs Unsupervised

Hello, and welcome. In this video we'll introduce supervised algorithms versus unsupervised algorithms. So, let's get started. An easy way to begin grasping the concept of supervised learning is by looking directly at the words that make it up. Supervise, means to observe, and direct the execution of a task, project, or activity. Obviously we aren't going to be supervising a person, instead will be supervising a machine learning model that might be able to produce classification regions like we see here. So, how do we supervise a machine learning model? We do this by teaching the model, that is we load the model with knowledge so that we can have it predict future instances. But this leads to the next question which is, how exactly do we teach a model? We teach the model by training it with some data from a labeled dataset. It's important to note that the data is labeled, and what does a labeled dataset look like? Well, it could look something like this. This example is taken from the cancer dataset. As you can see, we have some historical data for patients, and we already know the class of each row. Let's start by introducing some components of this table. The names up here which are called clump thickness, uniformity of cell size, uniformity of cell shape, marginal adhesion and so on are called attributes. The columns are called features which include the data. If you plot this data, and look at a single data point on a plot, it'll have all of these attributes that would make a row on this chart also referred to as an observation. Looking directly at the value of the data, you can have two kinds. The first is numerical. When dealing with machine learning, the most commonly used data is numeric. The second is categorical, that is its non-numeric because it contains characters rather than numbers. In this case, it's categorical because this dataset is made for classification. There are two types of supervised learning techniques. They are classification, and regression. Classification is the process of predicting a discrete class label, or category. Regression is the process of predicting a continuous value as opposed to predicting a categorical value in classification. Look at this dataset. It is related to CO2 emissions of different cars. It includes; engine size, cylinders, fuel consumption, and CO2 emission of various models of automobiles. Given this dataset, you can use regression

to predict the CO2 emission of a new car by using other fields such as engine size, or number of cylinders. Since we know the meaning of supervised learning, what do you think unsupervised learning means? Yes, unsupervised learning is exactly as it sounds. We do not supervise the model, but we let the model work on its own to discover information that may not be visible to the human eye. It means, the unsupervised algorithm trains on the dataset, and draws conclusions on unlabeled data. Generally speaking, unsupervised learning has more difficult algorithms than supervised learning since we know little to no information about the data, or the outcomes that are to be expected. Dimension reduction, density estimation, market basket analysis, and clustering are the most widely used unsupervised machine learning techniques. Dimensionality reduction, and/or feature selection, play a large role in this by reducing redundant features to make the classification easier. Market basket analysis is a modeling technique based upon the theory that if you buy a certain group of items, you're more likely to buy another group of items. Density estimation is a very simple concept that is mostly used to explore the data to find some structure within it. And finally, clustering: Clustering is considered to be one of the most popular unsupervised machine learning techniques used for grouping data points, or objects that are somehow similar. Cluster analysis has many applications in different domains, whether it be a bank's desire to segment his customers based on certain characteristics, or helping an individual to organize in-group his, or her favorite types of music. Generally speaking though, clustering is used mostly for discovering structure, summarization, and anomaly detection. So, to recap, the biggest difference between supervised and unsupervised learning is that supervised learning deals with labeled data while unsupervised learning deals with unlabeled data. In supervised learning, we have machine learning algorithms for classification and regression. In unsupervised learning, we have methods such as clustering. In comparison to supervised learning, unsupervised learning has fewer models and fewer evaluation methods that can be used to ensure that the outcome of the model is accurate. As such, unsupervised learning creates a less controllable environment as the machine is creating outcomes for us. Thanks for watching.



# Chapter 2

## Week 2

### 2.1 Linear Regression

#### 2.1.1 Introduction to Regression

Hello and welcome! In this video we'll be giving a brief introduction to regression. So let's get started. Look at this data set. It's related to co2 emissions from different cars. It includes engine size, number of cylinders, fuel consumption, and co2 emission from various automobile models.

The question is: given this data set can we predict the co2 emission of a car using other fields such as engine size or cylinders? Let's assume we have some historical data from different cars and assume that a car such as in row 9 has not been manufactured yet, but we're interested in estimating its approximate co2 emission after production. Is it possible?

We can use regression methods to predict a continuous value such as co2 emission using some other variables. Indeed regression is **the process of predicting a continuous value**. In regression there are **two types of variables**: a **dependent variable** and **one or more independent variables**.

The dependent variable can be seen as the state, target, or final goal we study and try to predict. And the independent variables, also known as explanatory variables, can be seen as the causes of those states. The independent variables are shown conventionally by X and the dependent variable is notated by Y.

A regression model relates Y or the dependent variable to a function of X i.e. the independent variables.

The key point in the regression is that our dependent value should be continuous and cannot be a discrete value. However, the independent variable, or variables, can be measured on either a categorical or continuous measurement scale.

So, what we want to do here is to use the historical data of some cars using one or more of their features and from that data make a model. We use regression to build such a regression estimation model; then the model is used to predict the expected co2 emission for a new or unknown car.

Basically there are two types of regression models **simple regression** and **multiple regression**. Simple regression is when **one independent variable is used** to estimate a dependent variable. It **can be either linear or non-linear**. For example, predicting co2 emission using the variable of engine size.

Linearity of regression is based on the nature of relationship between independent and dependent variables. When **more than one independent variable is present** the process is called **multiple linear regression**. For example, predicting co2 emission using engine size and the number of cylinders in any given car. Again, depending on the relation between dependent and independent variables it can be either linear or non-linear regression.

Let's examine some sample applications of regression. Essentially we use regression when we want to estimate a continuous value. For instance, one of the applications of regression analysis could be in the area of sales forecasting. You can try to predict a sales person's total yearly sales from independent variables such as age, education, and years of experience. It can also be used in the field of psychology, for example, to determine individual satisfaction, based on demographic and psychological factors. We can use regression analysis to predict the price of a house in an area, based on its size number of bedrooms, and so on. We can even use it to predict employment income for independent variables such as hours of work, education, occupation, sex, age, years of experience, and so on.

Indeed, you can find many examples of the usefulness of regression analysis in these and many other fields or domains, such as finance, healthcare, retail, and more. We have many regression algorithms, each of them has its own importance and a specific condition to which their application is best suited. And while we've covered just a few of them in this course, it gives you enough base knowledge for you to explore different regression techniques.

### 2.1.2 Simple Linear Regression

In this video, we'll be covering linear regression. You don't need to know any linear algebra to understand topics in linear regression. This high-level introduction will give you enough background information on linear regression to be able to use it effectively on your own problems. So let's get started.

Let's take a look at this data set. It's related to the Co2 emission of different cars. It includes engine size, cylinders, fuel consumption and Co2 emissions for various car models. The question is, given this data set, can we predict the Co2 emission of a car using another field such as engine size?

Quite simply, yes. We **can use linear regression to predict a continuous value such as Co2 emission by using other variables**. Linear regression is the approximation of a linear model used to describe the relationship between two or more variables. In simple linear regression, there are two variables, a dependent variable and an independent variable. The **key point** in the linear regression is that our dependent value **should be continuous and cannot be a discrete value**. However, the independent variables can be measured on either a categorical or continuous measurement scale.

There are two types of linear regression models. They are simple regression and multiple regression. Simple linear regression is when one independent variable is used to estimate a dependent variable. For example, predicting Co2 emission using the engine size variable. When more than one independent variable is present the process is called multiple linear regression, for example, predicting Co2 emission using engine size and cylinders of cars. Our focus in this video is on simple linear regression.

Now let's see how linear regression works. Okay, so let's look at our data set again. To understand linear regression, we can plot our variables here. We show engine size as an independent variable and emission as the target value that we would like to predict. A scatter plot clearly shows the relation between variables where changes in one variable explain or possibly cause changes in the other variable. Also, it indicates that these variables are linearly related. With linear regression **you can fit a line through the data**. For instance, as the engine size increases, so do the emissions. With linear regression you can model the relationship of these variables. A good model can be used to predict what the approximate emission of each car is.

How do we use this line for prediction now?

Let us assume for a moment that the line is a good fit of the data. We can use it to predict the emission of an unknown car. For example, for a sample car with engine size 2.4, you can find the emission is 214.

Now, let's talk about what the fitting line actually is.

We're going to predict the target value  $y$ . In our case using the independent variable engine size represented by  $x_1$ . The fit line is shown traditionally as a polynomial. In a simple regression problem, a single  $x$ , the form of the model would be  $\theta_0 + \theta_1 x_1$ . In this equation,  $\hat{y}$  is the dependent variable of the predicted value. And  $x_1$  is the independent variable.

Theta 0 ( $\theta_0$ ) and theta 1 ( $\theta_1$ ) are the parameters of the line that we must adjust. Theta 1 is known as the slope or gradient of the fitting line and theta 0 is known as the intercept.

Theta 0 ( $\theta_0$ ) and theta 1 ( $\theta_1$ ) are also called the **coefficients of the linear equation**.

You can interpret this equation as  $\hat{y}$  being a function of  $x_1$ , or  $\hat{y}$  being dependent of  $x_1$ . How would you draw a line through the points? And how do you determine which line fits best?

Linear regression estimates the coefficients of the line. This means we must calculate theta 0 and theta 1 to find the best line to fit the data. This line would best estimate the emission of the unknown data points.

Let's see how we can find this line or, to be more precise, how we can adjust the parameters to make the line the best fit for the data. For a moment, let's assume we've already found the best fit line for our data. Now, let's go through all the points and check how well they align with this line. Best fit here means that if we have, for instance, a car with engine size  $x_1 = 5.4$  and actual  $Co_2 = 250$ , its  $Co_2$  should be predicted very close to the actual value, which is  $y = 250$  based on historical data. But if we use the fit line, or better to say using our polynomial with known parameters to predict the  $Co_2$  emission, it will return  $\hat{y} = 340$ . Now if you compare the actual value of the emission of the car with what we've predicted using our model, you will find out that we have a 90 unit error. This means our prediction line is not accurate. This error is also called the **residual error**. So we can say the **error is the distance from the data point to the fitted regression line**.

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (2.1)$$

The mean of all residual errors shows how poorly the line fits with the whole data set. Mathematically it can be shown by the equation **Mean Squared Error**,

shown as **MSE**. Our objective is to **find a line where the mean of all these errors is minimized**. In other words, the mean error of the prediction using the fit line should be minimized. Let's reword it more technically. The **objective of linear regression**, is to **minimize this MSE equation and to minimize it, we should find the best parameters theta 0 and theta 1**. Now the question is **how to find theta 0 and theta 1 in such a way that it minimizes this error?**

How can we find such a perfect line? Or said another way, how should we find the best parameters for our line? Should we move the line a lot randomly and calculate the MSE value every time and choose the minimum one? Not really. Actually, we have **two options here**. Option one, we can use a **mathematic approach**, or option two, we can use an **optimization approach**. Let's see how we could easily use a mathematic formula to find the theta 0 and

As mentioned before, theta 0 and theta 1 in the simple linear regression are the coefficients of the fit line. We can use a simple equation to estimate these coefficients. That is, given that it's a simple linear regression with only two parameters, and knowing that theta 0 and theta 1 are the intercept and slope of the line, we can estimate them directly from our data. It requires that we calculate the mean of the independent and dependent or target columns from the data set. Notice that all of the data must be available to traverse and calculate the parameters. It can be shown that the intercept and slope can be calculated using these equations.

We can start off by estimating the value for theta 1. This is how you can find the slope of a line based on the data.  $\bar{x}$  is the average value for the engine size in our data set. Please consider that we have nine rows here, rows 0 to 8. First we calculate the average of  $x_1$  and of  $y$ , then we plug it into the slope equation to find theta 1.

The  $x_i$  and  $y_i$  in the equation refer to the fact that we need to repeat these calculations across all values in our data set. And  $i$  refers to the  $i$ th value of  $x$  or  $y$ . Applying all values, we find theta 1 equals 39. It is our second parameter. It is used to calculate the first parameter which is the intercept of the line.

Now we can plug theta 1 into the line equation to find theta 0. It is easily calculated that theta 0 equals 125.74. So these are the two parameters for the line, where theta 0 is also called the bias coefficient, and theta 1 is the coefficient for the Co2 emission column.

As a side note, you really don't need to remember the formula for calculating these parameters, as **most of the libraries used for machine learning in Python, R and Scala can easily find these parameters for you**. But it's always good to understand how it works. Now, we can write down the polynomial of the line.

$$\hat{y} = \theta_1 x + \theta_0 = 39x + 125.74 \quad (2.2)$$

So we know how to find the best fit for our data and its equation. Now the question is how can we use it to predict the emission of a new car based on its engine size?

After we found the parameters of the linear equation, making predictions is as simple as solving the equation for a specific set of inputs.

Imagine we are predicting Co2 emission, or  $y$ , from engine size, or  $x$  for the automobile in record number 9. Our linear regression model representation for

this problem would be  $\hat{y} = \theta_0 + \theta_1 x_1$ . Or if we map it to our data set, it would be  $\text{Co2Emission} = \theta_0 + \theta_1 \text{EngineSize}$ .

As we saw, we can find  $\theta_0$ ,  $\theta_1$  using the equations that we just talked about. Once found, we can plug in the equation of the linear model. For example, let's use  $\theta_0 = 125$  and  $\theta_1 = 39$ . So we can rewrite the linear model as  $\text{Co2Emission} = 125 + 39 \text{EngineSize}$ . Now let's plug in the 9th row of our data set and calculate the Co2 emission for a car with an engine size of 2.4. So  $\text{Co2Emission} = 125 + 39 \times 2.4$ . Therefore, we can predict that the Co2Emission for this specific car would be 218.6. Let's talk a bit about why linear regression is so useful. Quite simply, it is the most basic regression to use and understand. In fact, one reason why linear regression is so useful is that it's fast. It also doesn't require tuning of parameters. So something like tuning the K parameter and K nearest neighbors, or the learning rate in neural networks isn't something to worry about. Linear regression is also easy to understand, and highly interpretable.

### 2.1.3 Model Evaluation in Regression Models

### 2.1.4 Evaluation Metrics in Regression Models

### 2.1.5 Multiple Linear Regression

## 2.2 Non-Linear Regression

### 2.2.1 Non-Linear Regression