**MACHINE LEARNING SPECIALIZATION**

**DEEPLEARNING.AI**

Week 01

By Arthur Samuel in 1959:

Machine learning is the field of study that gives computers the ability to learn without being explicitly programmed.

In 1950 Arthur created a chess game that computers can play, and computer played that game 10,000 times to itself to get what is the good position and what is the bad position to win the game, and Arthur himself didn’t know about chess.

Machine learning algorithms:

1. Supervised Learning(used most in real world applications) (Course 1, Course 2)
2. Unsupervised Learning(Course 3)
3. Recommend systems
4. Reinforcement learning

**Supervised Learning:**

Creating a mapping from X input to Y output, in which we give our model some X input with correct Y labels to actually teach the computer on those conditions what is true, so computer can learn and then later on it can make better predictions.

**Applications of Supervised Learning:**

1. Input X (email) 🡪 🡪 🡪 Output Y (spam? 0/1) | Application (Spam filtering)
2. Input X (audio) 🡪 🡪 🡪 Output Y (text transcripts) | Application (speech recognition)
3. Input X (English) 🡪 🡪 🡪 Output Y (Spanish) | Application (machine translation)
4. Input X (ad, user info) 🡪 🡪 🡪 Output Y (click ? 0/1) | Application (Online advertising)
5. Input X (image, radar info) 🡪 🡪 🡪 Output Y (position of other cars) | Application (Self-driving cars)
6. Input X (image of phone) 🡪 🡪 🡪 Output Y (defect ? 0/1) | Application (Visual inspection)

**Regression**: Where the task is to predict a number from infinite number of possibilities. House price prediction for example. And remember the point is to add a best fit line on our data to actually capture the true data from training data. Maybe straight line isn’t enough so we can add a little curve line that captures the most patterns

**Classification**:

These algorithms are used to predict categories. Whether a person have malignant(a lump that is dangerous or cancerous) or benign(a lump that isn’t that dangerous or isn’t cancerous) in breast cancer detection situation.

In classification problems , people can use terminologies like class or category but that both are same.

In classification problems, you can have multiple categories for example our breast cancer detection can also tell different type of cancer which maybe in our dataset reffered to as “Malignant type 2”, then we can also predict that so in short our classification problem can have multiple categories.

**Difference BW Regression and Classification:**

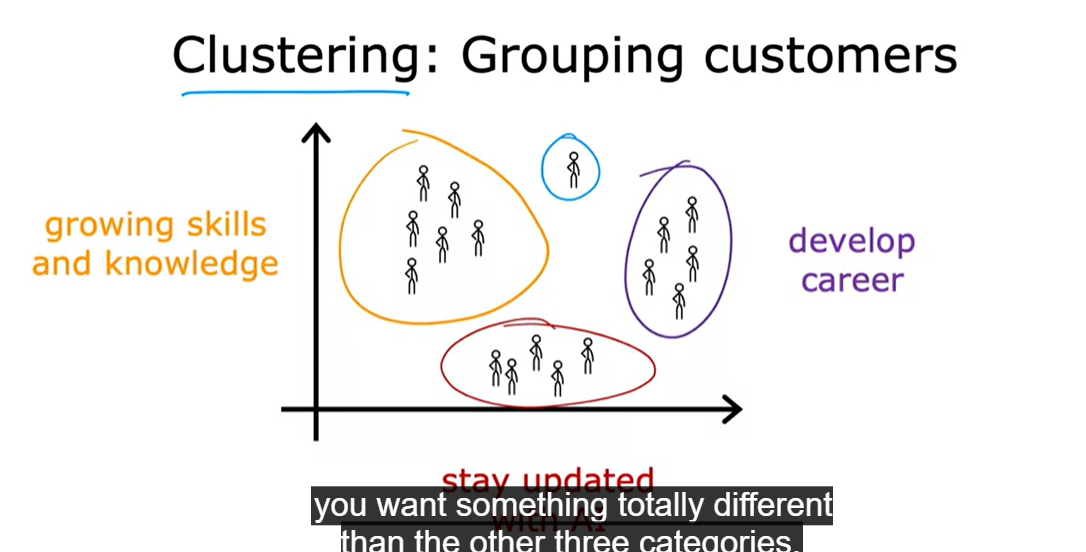
Classification can predict limited, possible number from a set of finite categories such as 0,1 or 0,1,2 but not all possible numbers between them like 0.5, 0.7 etc.

We can use more than one input value to predict the output. Not just the size of the tumor in this case, maybe a patient age and tumor size are both in combination using to predict whether the person has benign or malignant.

**Unsupervised Learning:**

It is a type of machine learning that learns from data without human supervision. Unlike supervised learning, unsupervised machine learning models are given unlabeled data and allowed to discover patterns and insights without any explicit guidance or instruction. We basically provide some data X and it produces insights that what group this is or what this thing can relate to and so on.

A **clustering algorithm** is a part of it, and how it works is basically it groups together the data those patterns are somehow relatably same. For example on deep learning . ai , we want to group customers on the basis of the reasons like why they are here on our platform.



Clustering algorithm takes data without labels and tries to automatically group them into clusters.

Maybe we can also take an example of google news , have you ever wondered how it groups the news on to one category that define all of those news ?



The reason how it clusters into one group that it it somehow finds some patterns in all four news like for example the words like panda, zoo, birth, twin are in also other news so it groups them into a cluster by using clustering algorithm.

Let’s take a look at unsupervised learning algorithms other than clustering.

In unsupervised learning, data only comes with inputs X, but not with output labels y. because in supervised learning we know that we also get output labels in the data but for unsupervised learning there is no supervision. Data only comes with X.

We have looked at clustering algorithm which groups similar data points together.

**Anomaly Detection:** Which is used to detect unusual events. Find unusual data points. For example detecting a fraud transaction where unsual events, unusual transactions could be signs of fraud and for many other applications.

**Dimensionality reduction:** Compress data using fewer numbers, for example compressing a big data losing a little information as possible.

**Regression Model:**

It is a type of supervised learning and it predict numbers.

**Linear regression with one variable:**

**Linear regression model part 1:**

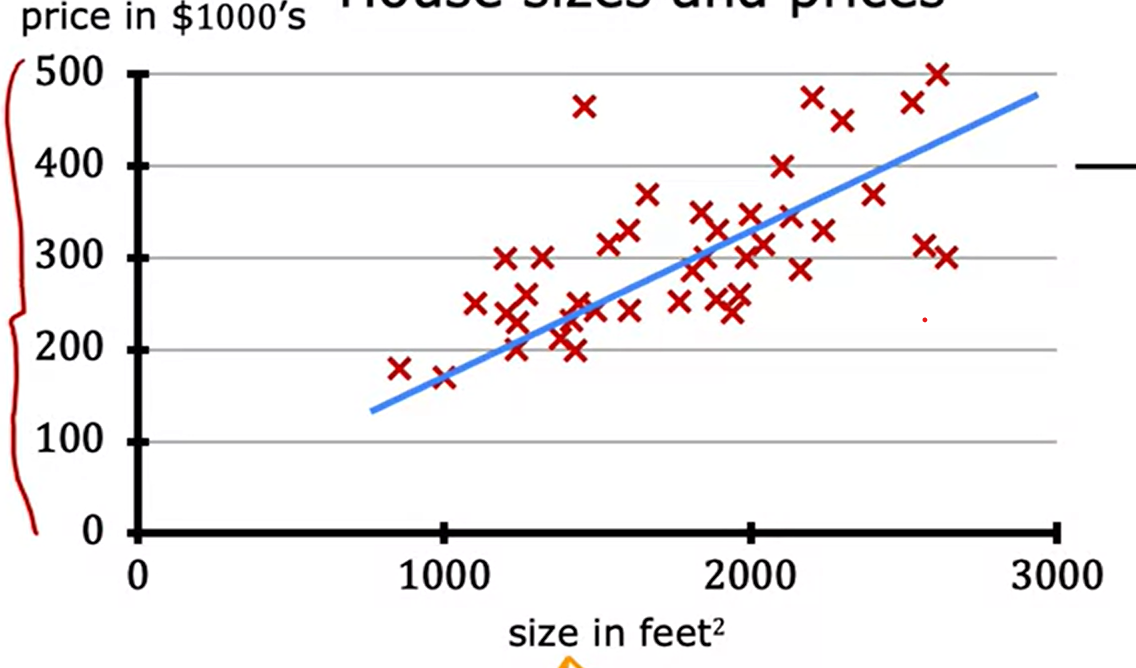
Predicts number such as 220,000 or 1.5 or -33.5. Linear regression is one example of a regression model. There are also other models for addressing regression problems to.

**Classification Model:**

It predicts categories or discrete categories. Such as a picture is a cat or a dog. Or predict if a patient have a particular disease.

Difference between classification and regression:

In classification there are only a small number of possible outputs. If your model recognizes cat vs dog then that’s two possible outputs or maybe you want to recognize 10 possible conditions of a patient. So there’s a discrete(separate) finite set of possible output. We call it classification problem. Where as, In regression there are infinity many possible numbers that the model could output.



**Each cross corresponding to one row of the table.**

**Terminology:**

The dataset that is used to train the model is called a training set.

To predict the unknown value or number from a dataset, we first train the model from training set and then model can predict that unknown value or a number.

**Notation:**

X(uppercase) = “input” variable (also called a feature or an input feature)

y(lowercase) = “output” variable (also called a target variable)

m(lowercase) = total number of training examples (total number of rows)

(X,y) = Single training example (now basically this is referring to every example in a dataset, to specifically select a training example. We can use something like this:

( x(i) , y(i) ) = ith training example ( the I in superscript is not an exponentiation, its just refers to a specific row which is a training example.

**Linear regression model part 2:**

To train the model, we feed the training data( features and targets ) to our learning algorithm, then our supervised learning algorithm will produce some function. We denote this function as lowercase **“ f “,** where f stands for function. Historically, this function used to called a **Hypothesis**. The job with f is to take a new input x and output and estimate or a prediction which we can call ^y, (y-hat) like a letter y and above it there is a little hat ( ^ ), y-hat is the prediction or prediction for y. The function f is called the model. So X is the input feature and the output of that model is the prediction, y-hat. The model’s prediction is the estimated value of y. When symbol is just the letter y, than that refers to the target, which is the actual true value in the training set. In contrast, y-hat is an estimate it may or may not be the actual value.

So let’s say you are to predict a value which is unknown and you know the input variable , then your function will take an input x and outputs the y which is the estimated that is the prediction of what the true price will be. When we design a learning algorithm, key question is how we are going to represent the function **f**?

What is the formula we’re going to use to compute f?  
For now, let’s stick with f being a straight line.

**How to represent a function(f)?**

Your function can be written as :

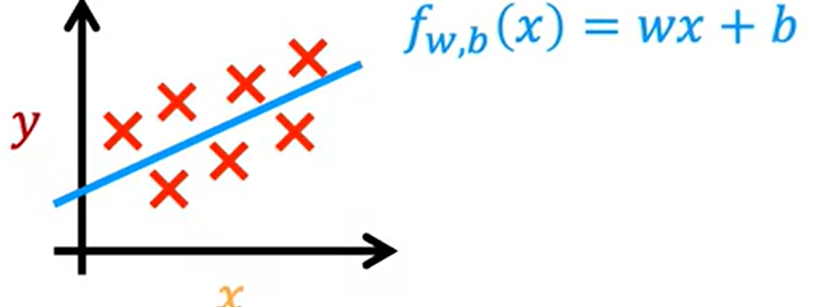
f w,b(x)= wX + b

I can also write f(x) = wX + b

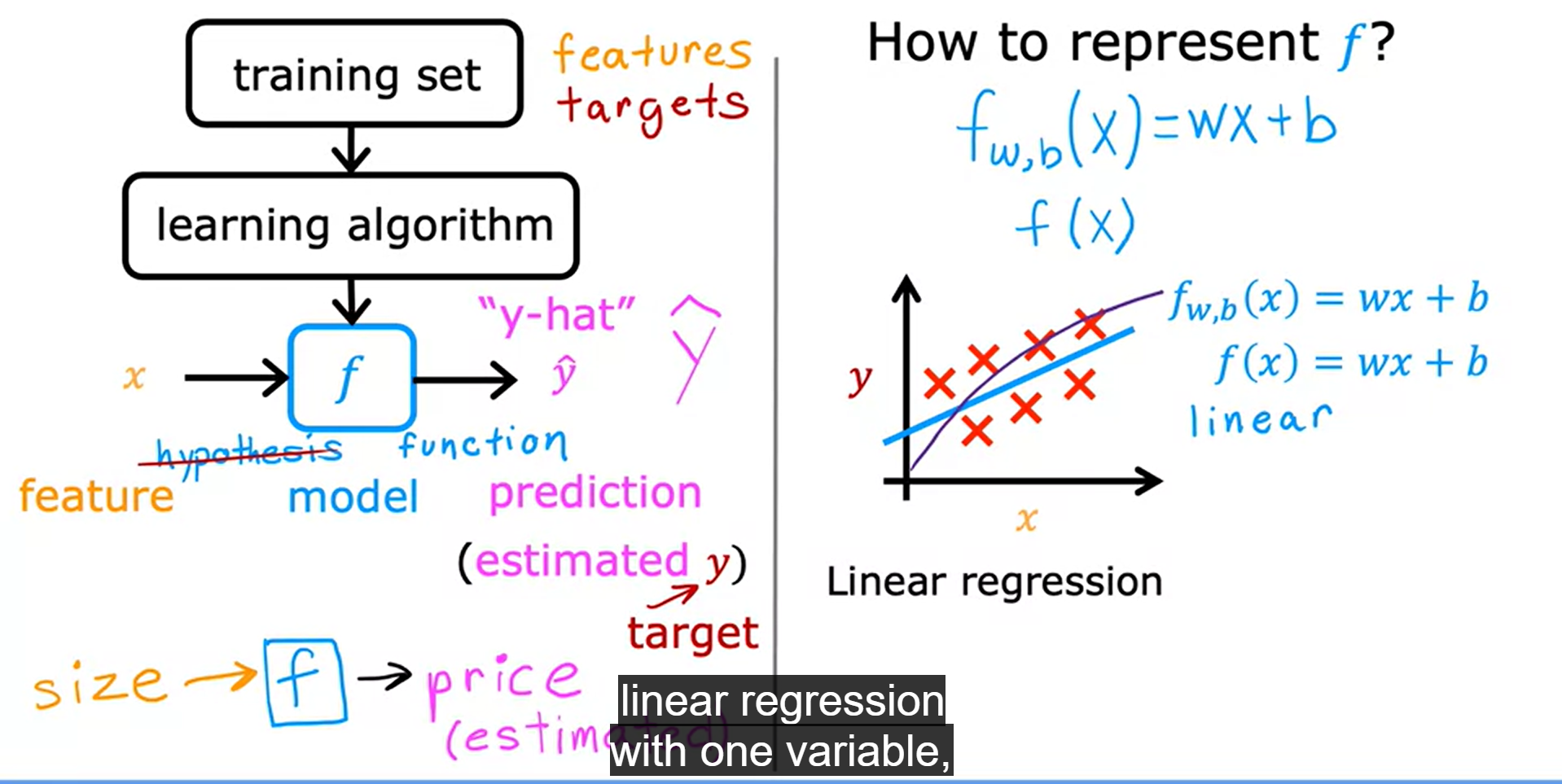
Without explicitly writing w and b as subscript.

The value of w and b will determine the prediction y-hat based on the input feature X. This means f will take an input and depending on the values of w and b f will output some value of a prediction y-hat.

The straight line that pass through the data points in visualization is a linear function which is f(x) = wX + b



Here's what this function is doing, it's making predictions for the value of y using a streamline function of x. You may ask, why are we choosing a linear function, where linear function is just a fancy term for a straight line instead of some non-linear function like a curve or a parabola? Well, sometimes you want to fit more complex non-linear functions as well, like a curve like this. But since this linear function is relatively simple and easy to work with, let's use a line as a foundation that will eventually help you to get to more complex models that are non-linear. This particular model has a name, it's called **linear regression.**



This is linear regression with one variable.

One variable means there is single input variable or feature x. Another name for a linear model with one variable input is **univariate linear regression.** Where uni means one in latin and where variate means variable.

Simple y is used to denote the output or “target” variable.