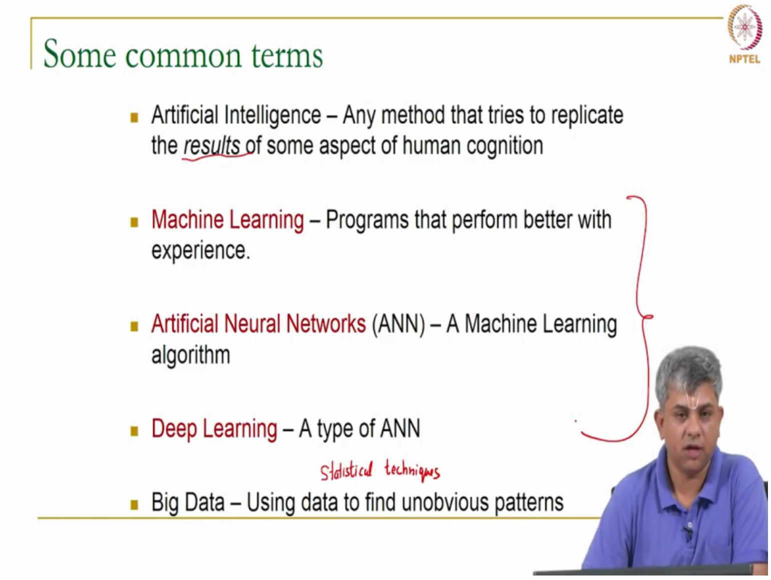
Machine Learning for Engineering and Science Applications.
Professor Dr. Balaji Srinivasan.
Department of Mechanical Engineering.
Indian Institute of Technology, Madras.
Overview of Machine Learning.

We will be looking at an overview of machine learning algorithms. In the last video, we saw a brief overview of the history of machine learning. Today we will be looking at a broad set of ideas that play themselves again and again in machine learning.

(Refer Slide Time: 0:33)



Some common terms

- Artificial Intelligence – Any method that tries to replicate the *results* of some aspect of human cognition
- Machine Learning – Programs that perform better with experience.
- Artificial Neural Networks (ANN) – A Machine Learning algorithm
- Deep Learning – A type of ANN
- Big Data – Using data to find unobvious patterns

Statistical techniques

So here are some common terms that you would encounter if you have just been new to machine learning. One is the term of artificial intelligence. Artificial intelligence is a very broad term, it simply means animator that tries to replicate the results of some aspect of human cognition. The reason the word results is being emphasised, is because we might not actually replicate the processes themselves but only the results. So, if somebody is playing chess, somebody is driving car, all you want to do is to make sure that the final output is the same, whether it is a machine or whether it is a human being.

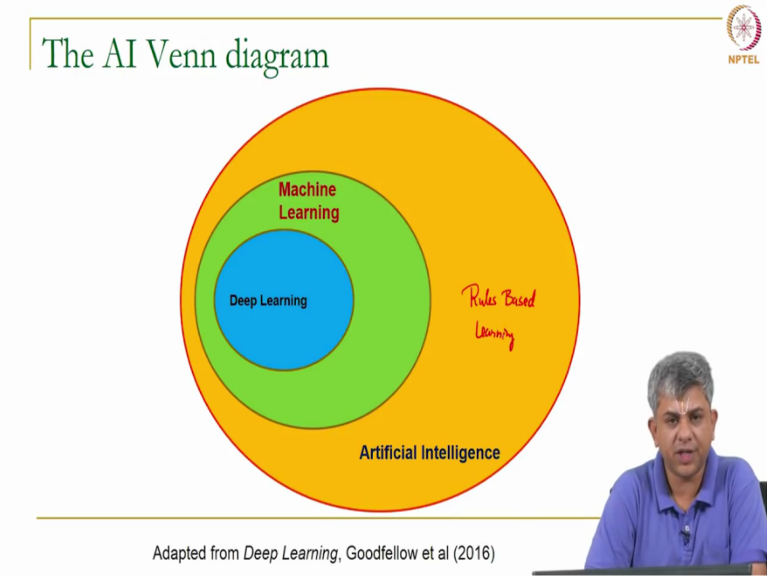
As against this, machine learning is a specific term, that means programs that actually perform better as your experience grows. What is meant by experience is something that we will discuss a little bit later. At what it means is if you have, let us say Calculator, the calculator is not getting better. You know, as you ask it to do multiplications again and again and again, but if a human being is there, the person might actually get more accurate or faster as they do multiplications for a while.

So, machine learning, if suppose to replicate this process which is as experience in a field grows, whether it is spam detection, or whether it is vision or anything of that sort. Machine learning if the set of algorithms which actually gets better. Artificial intelligence might or might not actually get better with experience. You would have also heard the term neural networks or artificial neural networks, there are type of machine learning algorithm.

And most commonly, you would have heard a term Deep Learning, which is a certain type of artificial neural network. Nowadays it is being used in a broader sense, but more technically, all it means is a neural network with a bunch of layers, which we will see later. Finally, you would have heard the term Big Data, this is not a term that we will be using as far as this course is concerned but simply it is a set of statistical techniques, which we also use within Machine learning. The basic idea between, in big data, let us say often used very commercially, is to find out an obvious pattern.

In Machine learning typically is to find out patterns which are obvious to human beings, but might not be obvious to programs. Okay. But big data is typically try to find out patterns which are not really obvious to human beings. So as far as this course is concerned, we will be looking at primarily these 3, we are not looking at big data techniques or more general artificial intelligence techniques.

(Refer Slide Time: 3:18)

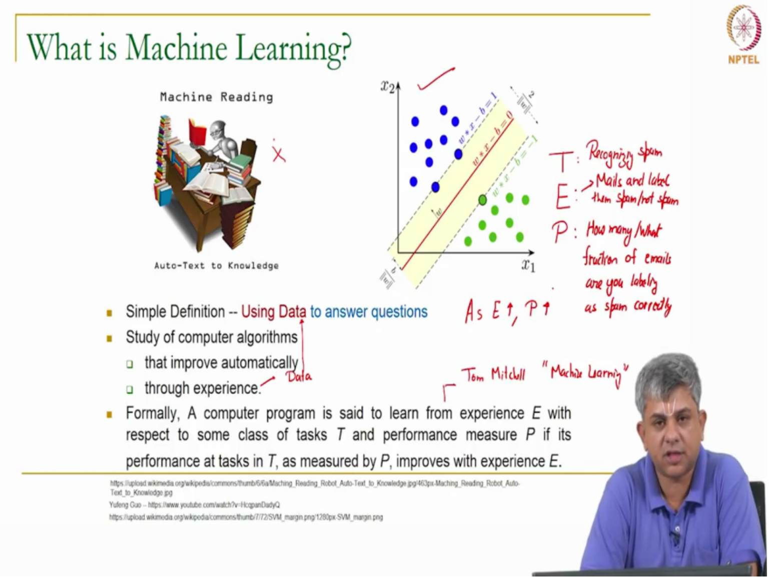


The AI Venn diagram

Adapted from *Deep Learning*, Goodfellow et al (2016)

So here is a kind of Venn diagram to show the relationship between various terms. This has been kind of adapted from Goodfellow's book. So artificial intelligence as you can see is a broad term, broad term that encompasses a lot of things, it also encompasses rules-based learning, which I discussed in the last video. Machine learning is explicitly not rules-based, which we will see a little bit later. And in deep learning is a particular subset of machine learning itself.

(Refer Slide Time: 4:02)



What is Machine Learning?

T: Recognizing spam

E: → Mails and label them spam/not spam

P: How many/What fraction of emails are you labeling as spam correctly

As E ↑, P ↑

- Simple Definition -- Using Data to answer questions
- Study of computer algorithms
  - that improve automatically
  - through experience. Data
- Formally, A computer program is said to learn from experience *E* with respect to some class of tasks *T* and performance measure *P* if its performance at tasks in *T*, as measured by *P*, improves with experience *E*.

Tom Mitchell "Machine Learning"

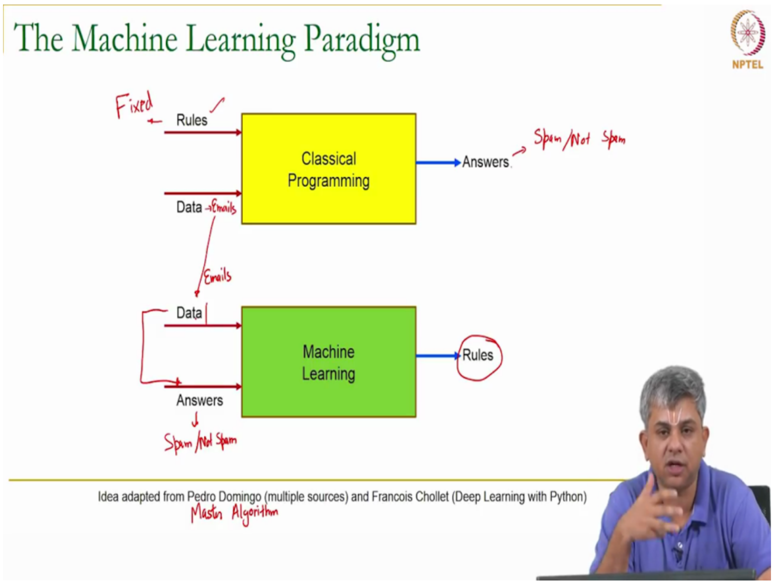
https://upload.wikimedia.org/wikipedia/commons/thumb/6/6a/Maching_Reading_Robot_Auto-Text_to_Knowledge.jpg/463px-Maching_Reading_Robot_Auto-Text_to_Knowledge.jpg
Yufeng Guo – https://www.youtube.com/watch?v=HcqpanDadyQ
https://upload.wikimedia.org/wikipedia/commons/thumb/7/72/SVM_margin.png/1280px-SVM_margin.png

So what is machine learning? If you are completely unfamiliar with the field, you might think it looks something of this sort, this is obviously not true. Okay, it is not a machine which is reading books or trading information and trying to learn something. A very simple definition, which is from Gou is that it is simply using data to answer questions, more specifically an actual machine learning algorithm looks more like this rather than this. It actually looks like this, this is an algorithm called support vector machines, will be saying this later on some way towards the middle of this course.

So, support vector machines or other machine learning algorithms work as follows. Machine learning is simply a study of computer algorithms that actually improve automatically through experience. So the term experience simply means lot of data, okay. A formal definition, which is there in the textbook by Tom Mitchell, this textbook is called machine learning. It is that suppose you have a task T, okay. And you have some experience E on it and you have a performance measure P.

A standard example, task T could be let us say recognising spam. Okay, suppose you have emails, you want to recognise whether the email is spam email or not. The experience E is the data that you give, you give mails and label them spam or not spam. So this would be the experience that you are giving this program. P is the performance measure, the performance measure is how many or what fraction of emails are you labelling as spam. Okay.

(Refer Slide Time: 7:02)



The Machine Learning Paradigm

NPTEL

Fixed

Rules

Data → Emails

Classical Programming

Answers → Spam / Not Spam

Emails

Data

Answers

Spam / Not Spam

Machine Learning

Rules

Idea adapted from Pedro Domingo (multiple sources) and Francois Chollet (Deep Learning with Python)

Master Algorithm

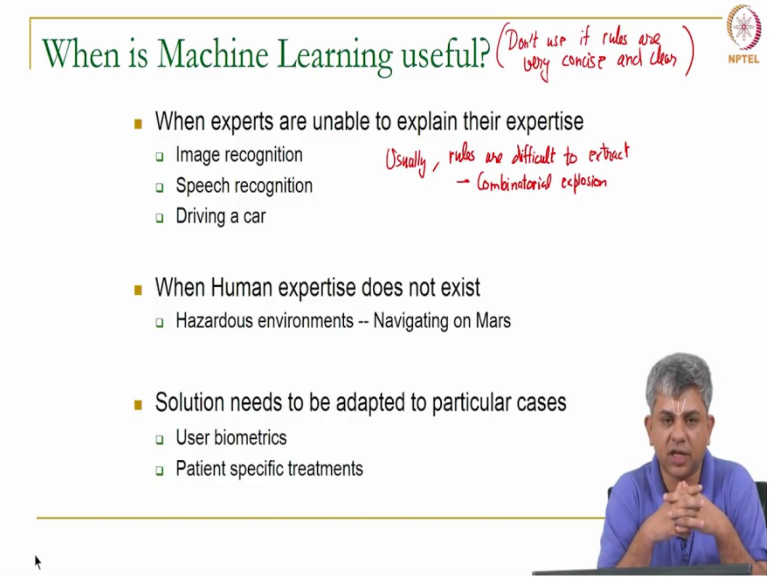
So what this definition says is, as E increases, the performance should get better and better. So any algorithm that achieves this is called a machine learning algorithm. So, here is a machine learning paradigms, okay, so this idea is adapted from Pedro Domingo, he has got a very good book. And actually multiple sources, he has a course online also, a book called Master Algorithm, which is a popular kind of book. I would recommend that you read it. Also Francois Chollet's Deep Learning with Python book have this idea.

So this is a classical programming thing. So you have certain rules and your certain data, it is processed by the program and gives answers. For example if you have classical programming approach to spam detection, you would have certain rules. For example if there are too many caps or if the email talks about money and puts a dollar in the middle, something of that sort, those would be the rule. Then the data would be the emails that you are giving it and once the rules in the emails are given, it will give you some answers, spam or not spam, okay.

So the important thing here is these rules are fixed, that would be classical approach as against a machine learning approach. Now machine learning approach is as follows. You give the data which is still the same set of emails, you also give the answers which is, whether it is spam or not spam, and it figures out the rules for itself. Okay. What is the rule that maps this data to this answer. Okay, so this is the basic idea of machine learning which is you have to find out a mapping between your input and your output. In this case the input is that it out the emails and the output is the answers, whether it is spam or not spam.

In other cases you could have data like, you have an image, is this a cat, is this a dog, is this a horse, those are the answer. So to show it thousands of images of cats, dogs and horses and you label each one, this would be an example of what is called supervised learning. And then it finds out what rule is it that we are implicitly using in order to figure out what a cat looks like, what a dog looks like, what a horse looks like, etc., etc. So you can use this kind of paradigm for practically everything, as you will see throughout this course.

(Refer Slide Time: 9:20)



When is Machine Learning useful? (Don't use if rules are very concise and clear)

- When experts are unable to explain their expertise
  - Image recognition
  - Speech recognition
  - Driving a car

Usually, rules are difficult to extract
— Combinatorial explosion

- When Human expertise does not exist
  - Hazardous environments -- Navigating on Mars
- Solution needs to be adapted to particular cases
  - User biometrics
  - Patient specific treatments

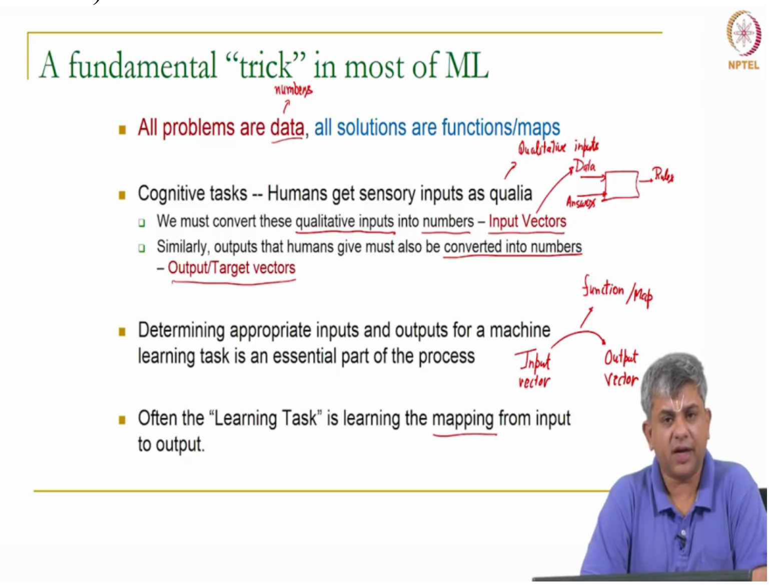
So when is this kind of machine learning useful? It is not a generally a good idea to use machine learning when you are actually very very clear about the rules. So this is some, generally this is true, we will see some exceptions for this. One thing I will mention is, typically a rule of thumb is do not use if the rules are very concise and clear. Okay, so there is no ambiguity about what the rules are and you are not a victim of combinatorial explosion, in such cases machine learning is probably not the best thing to go for.

However in cases where experts are not able to explain their expertise. For example, you drive a car, how do you drive a car, it is not very easy to concisely explain it into a set of finite rules, that this is how I am driving a car, this is how I recognise that something is spam or not spam. It seems kind of obvious to us when we see our friend, whether this friend has a cap on, different shirt on, we can immediately recognise that this is the same friend, that our parent is so-and-so, even a child recognises this fairly quickly.

In such cases, when we are not able to explain our expertise, it usually means rules are difficult to extract. The more obvious it is, the more difficult it is to extract the rules, okay. And usually will have combinatorial explosion, that is that the problem gets more and more complex, even for slight amount of increase of complexity, the number of rules you will have to give are too many. In such cases, it is usually better to use a machine learning paradigm, that is to simply say this is my input, this is my input, figure out the rules for yourself.

In certain other cases, even if you might note the rules, though the examples that I have used here, even there navigation is a hard problem. Even for hazardous environments, it is usually a good idea to use machine learning or any other artificial intelligence algorithm. Also when you have solutions that need to be an adapted to very specific cases. For example if you want a patient specific treatment for their particular, for their particular allergies, again the number of rules that you will have to give will be too many.

(Refer Slide Time: 11:51)



A fundamental "trick" in most of ML

- All problems are data, all solutions are functions/maps
- Cognitive tasks -- Humans get sensory inputs as qualia
  - We must convert these qualitative inputs into numbers – Input Vectors
  - Similarly, outputs that humans give must also be converted into numbers – Output/Target vectors
- Determining appropriate inputs and outputs for a machine learning task is an essential part of the process
- Often the "Learning Task" is learning the mapping from input to output.

So in such cases also machine learning can be quite useful, okay. So here is the fundamental trick that is utilised in most of machine learning. Almost all of machine learning, this is, uses this fundamental idea which is every problem that you have, whether it is a face recognition problem, spam recognition problem, you know fluid mechanics problem, whatever it is, every problem can be posed as a data problem. Okay. A data here means something involving numbers. Okay.

And all solutions that we offer can be thought of as a function or a map, okay. So, here is the problem, so for example let us say we are doing an image recognition problem, I will go back to the same example. You have an image, he will not recognise whether it is a cat or a dog, okay. So the problem is when we get sensory input, this is as qualia or basically we get qualitative inputs. These are not numbers. So when you see a cat, almost invariably all you see is a certain features of the cat, use the eyes, ears, nose, etc., you do not actually see numbers.

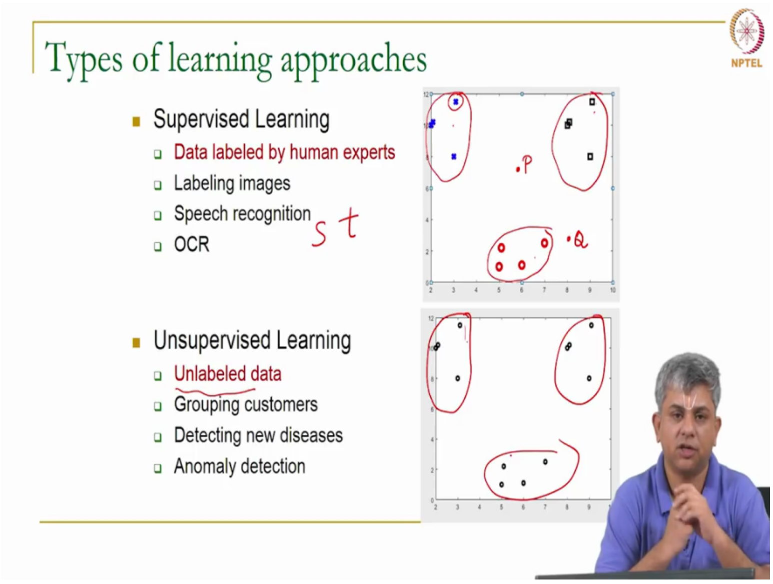
However if you want to turn it into a data problem, you will actually have to somehow change this from an image to numbers. Okay. So these images which we are getting as inputs for our problems, these qualitative inputs have to be turned into numbers and after this transformation, this is called an input vector. This is what goes into the program, okay. So when I have the box and we have data coming in here, that was this. Those are the input vectors that you are giving to the problem.

Similarly you have output that we give, so let us go back to the same example. If I see an image I can call it a cat or I can call it a dog but cat and dog are words, these are not numbers. You again have to turn these into numbers as well and these will be called output or target vectors, okay. So they are the answers in the previous slide, these also have to be posed as numbers. There is a slight difference between output and target vector. Output vector is what the machine will give out, in the final case target vectors are what begins as examples in the middle, we will see this later. Okay.

So, an essential part of the process of machine learning is to somehow decide on what are the appropriate inputs and what are the appropriate outputs. This can also be easily turned into numbers and with which you can train your algorithm. This is an essential part of the process. Even the rules of the get out have to be finally posed in terms of formulae, programs or numbers, okay. Now the learning task is to find a map that takes the input and gives out the output. So this could be thought of as a function that takes in an input vector and gives out an output vector. Okay, so this is a function or a map that does this.

So this is the fundamental trick that we will always use. Any problem you have, whether it is a cognitive problem, any problem can always be turned into a problem which takes in a bunch of numbers and gives out a bunch of numbers and what we want to find out is what will map these input numbers into the output numbers. So this is the fundamental idea behind most of the machine learning. Let us come to various types of learning problems.

(Refer Slide Time: 16:12)



Types of learning approaches

- Supervised Learning
  - Data labeled by human experts
  - Labeling images
  - Speech recognition
  - OCR
- Unsupervised Learning
  - Unlabeled data
  - Grouping customers
  - Detecting new diseases
  - Anomaly detection

Now, before I go into this, I want to point out that even though we have split into several types of learning approaches, this has been done traditionally, not all of them have clear boundaries. So you might find a case that goes into one type of learning approach or the other type of learning approach, let us see a few. So, one of the most popular, even the examples that I have used most commonly are what is called supervised learning. Supervised learning is data which is labelled by human experts. You have somehow labelled this data and you have set for this input, this is the output.

An example is something of this sort, let us say you have some log data points, each of these data points could represent anything, please remember from the last slide, each point here could represent a whole image because any image can also be turned into a vector, it can be turned into a subset of numbers, okay. So let us say we have 3 types of data which you can see, you have one set of crosses which are blue, one set of squares which are black and one set of circles which are red.

And suppose somebody gives a new point, which is here or someone gives a point here and you want to find out whether it is of type cross, type square or type circle? Okay, this is a supervised learning problem. The spam-not spam example I gave you was also a supervised learning problem because each example you gave, each email you gave you also simultaneously said is this spam or not spam. So your dataset if it is labelled by an human expert already and tells you example outputs, that is a supervised learning problem. Okay.

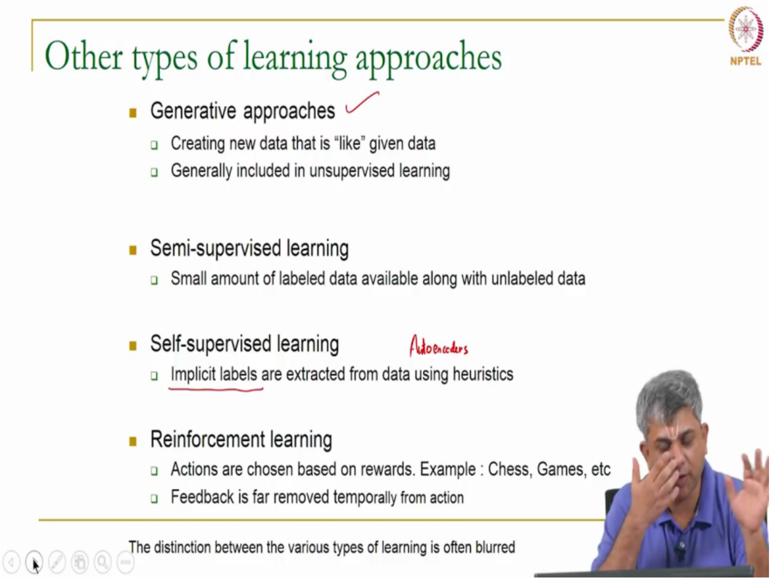
So some examples are labelling images, speech recognition, optical character recognition which is to turn written stuffs by human beings into actually finding out whether this is you know, is this S, P, etc, which is called optical character recognition. When you do handwriting recognition, or printed material recognition out of images, that is a supervised learning problem. So large parts of problem actually can be turned into supervised learning problems. Another important category is what is known as a supervised learning. In this case, the label for the data is not given.

So let us see the same data here, except the difference is that instead of giving Cross, square and circle, I have not made any distinction between the data. Nonetheless, as human beings we can automatically recognise that there are some clusters here. Then this might be one type of vector, this might be another type of data, and this might be a 3rd type of data. In such cases, supervision order labels are not given, nonetheless we are supposed to automatically recognise the natural clusters that are forming, okay.

So such cases can be used in multiple applications, such as you know, you have, let us say customers about 40 are purchasing in a certain way, customers believe 20 are purchasing in a certain way but you do not know a priori that these are customers above 40 and these are customers below 20, etc. But you see certain buying patterns, in such cases, you know the data will naturally formed clusters and the machine is supposed to recognise automatically, even though it seems obvious to us. At the machine is supposed to recognise through some algorithm that this is one cluster that is another cluster.

So in such cases detecting new diseases, finding out something like credit card fraud, a customer has been pertaining in a certain way for a long time and suddenly there is a change in pattern of purchase, that would be an anomaly detection and that is a type of unsupervised learning problem. There are some types of learning approaches which often lie at the interface of supervised and unsupervised learning.

(Refer Slide Time: 19:55)



Other types of learning approaches

- Generative approaches
  - Creating new data that is "like" given data
  - Generally included in unsupervised learning
- Semi-supervised learning
  - Small amount of labeled data available along with unlabeled data
- Self-supervised learning Autoencoders
  - Implicit labels are extracted from data using heuristics
- Reinforcement learning
  - Actions are chosen based on rewards. Example : Chess, Games, etc
  - Feedback is far removed temporally from action

The distinction between the various types of learning is often blurred

One set of problems that we will be looking at are what are called as generative approaches. The idea behind the generate approach is to create a new data, that is somewhat like a given set of data for the files Apple if I show you 100 images of cats, any human being can try at least and draw in new cat which will not look like the 100 images that you already saw but it will look somewhat different but it will at least extract key portions of a cat.

So such a learning approaches called a generative approach. This is neither labelling, nor clustering what it is actually generating new data. Typically this is included within unsupervised learning. We will be covering generative approaches towards the end of this course and also during some sequence learning. There is another type of learning, this is called semi-supervised learning, this is also quite possible, especially in medical images. You have small amount of labelled data available, along with unlabelled data.

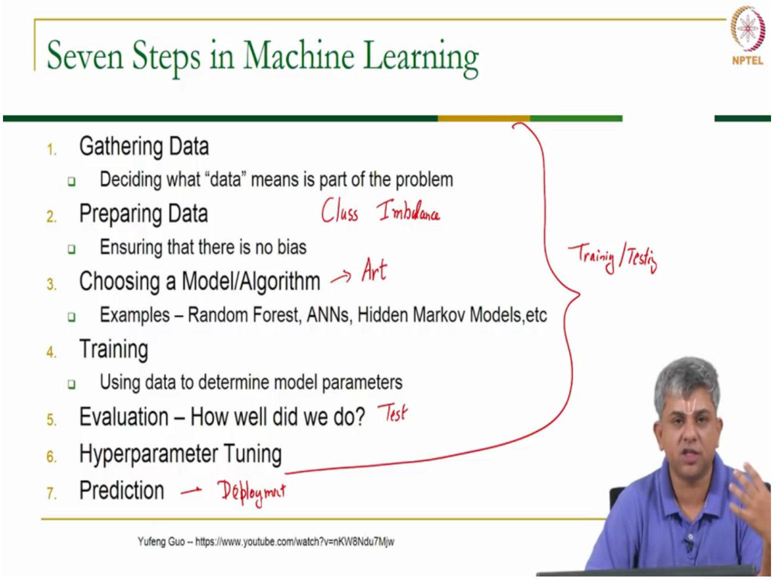
So you have let us say if you are MRI scans, and you have some let us say labelled tumours etc. within that. But you also have a lot of other data where the expert has not been able to go over the data. In such cases you kind of leveraged the labelled data and then use the amiable data and start solving a full supervised learning problem and this is called semi-supervised learning. There is also something called self supervised learning, where you actually do not have any labelled data at all, but you can kind of figure out some implicit labels, okay, from data using heuristics.

An example of this what was called auto encoders, which we will cover later on in the course. Another example would be something like you have a few video frames and you want to predict the next video frame. In such a case, you would kind of use self supervised learning, okay. Finally, we have something called reinforcement learning, which is getting a lot of traction nowadays. So, in such cases the easiest example for reinforcement learning would be something again like chess or any video game that you play.

So you make a move and you know maybe 20-30 moves later you get to know and you get to know only one thing, it even, did you lose or did you draw. But early on, 20-30 moves ago, you do not know whether that particular move that you chose led you to win or led you to lose. Okay. So you are trying to find out what action to take at a particular point based on rewards that are really really far removed in time. Okay. So unlike, let us say simple supervised learning, where I show you an image of a cat and say cat, here you are making a move, you do not know whether the move is right or wrong, whether it led you to win, lose or draw but you know the result of a combination of moves after a long time, okay.

So trying to learn under such an environment is called reinforcement learning, we will be looking at a brief introduction to this also towards the end of the course. Now again I will repeat the same point that I made earlier, which is the distinction between these various classes is actually quite blurred quite often. We will mostly approach the course as if we are doing neither supervised learning or unsupervised learning. So here are again according to Guo some 7 steps in machine learning. This is not a hard and fast rule but this is a very good abstraction of the whole machine learning process.

(Refer Slide Time: 23:39)



Seven Steps in Machine Learning

1. Gathering Data
   - Deciding what "data" means is part of the problem
2. Preparing Data Class Imbalance
   - Ensuring that there is no bias
3. Choosing a Model/Algorithm → Art
   - Examples – Random Forest, ANNs, Hidden Markov Models,etc
4. Training
   - Using data to determine model parameters
5. Evaluation – How well did we do? Test
6. Hyperparameter Tuning
7. Prediction → Deployment

Training/Testing

Yufeng Guo – https://www.youtube.com/watch?v=nKW8Ndu7Mjw

The first step of course is to decide on what data you want to use for the problem and then gather the data. Often you will have to do other things, here you will have to clean the data app, etc, etc, etc, which is the 2nd step. You also want to ensure that there is no inherent bias. So, for example people doing an election polling. They want to make sure that they have not taken one section or the other. Similarly when you are like, let us say data for whether a person has cancer or not, you are likely to have something which is called class imbalance.

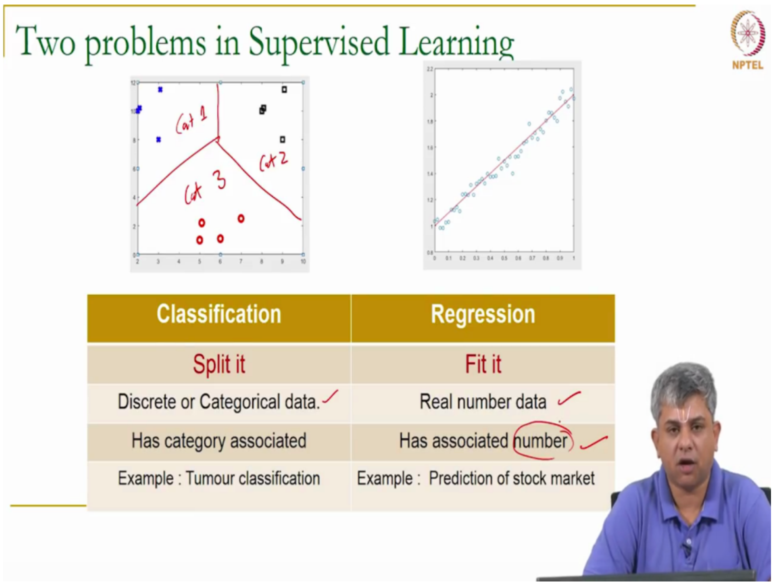
That is because if I randomly collect data from the population, 99.5 percent of the people are bound not to have cancer. So even if I take random data and say this person does not have cancer, I am going to be right 99.5 percent of the time. Because the amount of data that I have for people with cancer is actually very very low. So when you prepare data, you want to make sure that either the data is without bias or that you have sufficiently accounted for this in your algorithm.

The 3rd part is choosing a model or algorithm. So we will be covering a large number of algorithms through this course, okay. So some of those are written here random forest, artificial neural networks, etc, etc. And so, choosing an algorithm is part of the problem, there is no hard and fast simple rule for which algorithm works the best for which problem. This is very much like modelling in engineering sciences, there is no always clear model that you can use. Some models perform well in some domains and some perform well in some other domains.

We are discussing details of several models and algorithms in this course so that you can appropriately choose and find out. Of course choosing this is more of an art than a science, okay. And then comes training, each model that you will have will have certain unknown parameters which we will see in the rest of the course. For using data in order to determine model parameters is known as training. And of course, after you do this, you then try and find out, you test how did my particular model and particular set of parameters do.

And if it did not do well, you might have to tune a few things, hyper parameters we will come to in the end. And after this whole process is over, this is the training process, training and testing process, prediction is final deployment. So let us say out of all this he made an app which does machine learning and it is a cat identifier. Final prediction is you deploy the app and the customer uses it in order to deploy and check whether this person has a cat or not or whether it is spam or not spam.

(Refer Slide Time: 26:35)



Two problems in Supervised Learning

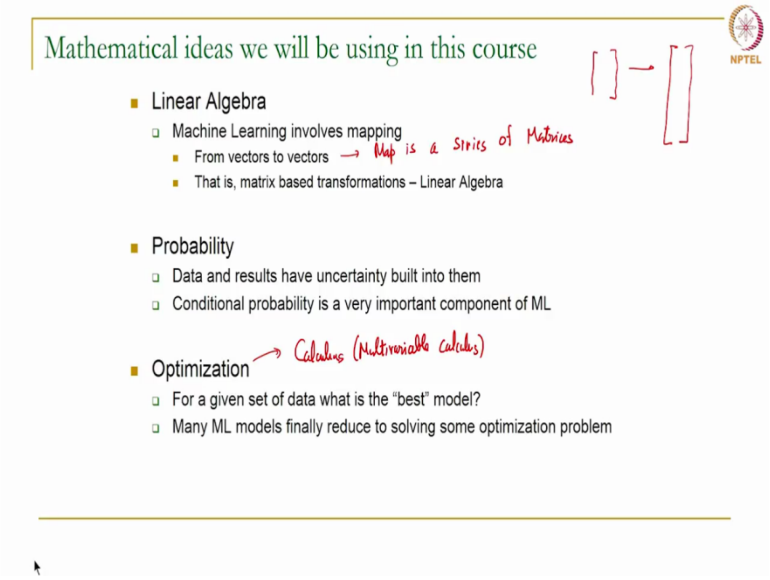
Classification	Regression
Split it	Fit it
Discrete or Categorical data. ✓	Real number data ✓
Has category associated	Has associated number ✓
Example : Tumour classification	Example : Prediction of stock market

So the first set of algorithms we will be doing in the course will be supervised learning algorithms. And typically in supervised learning it splits into 2, you either have classification problem. Classification problem simply means you want to split the data into discrete categories. So this could be category 1, this could be category 1 and this could be category 3. So, all the persons, the end result that they want is to know what is this, is that A, B or C, this is a cat, dog or a horse, is this email spam or not spam, such a problem is called a classification problem.

This happens whenever you have discrete data. For example cancer, not cancer, benign tumour, malignant tumour, etc. tumour classification, etc would be classification problems. Another problem is a regression. Regression says it typically has real number data, has a number of associated with it and you have an example of something happening in the past. You could have house prices depending on their area, you could have for example the example that is written on the slide, you could have previous stock prices and you want to know what the stock price is going to be tomorrow.

Such problems are regression problems, they are not really speaking classification problems. This is not good or bad but you actually want an actual number out of this one set of numbers. These problems are known as regression problems.

(Refer Slide Time: 28:17)



Mathematical ideas we will be using in this course

- Linear Algebra
  - Machine Learning involves mapping
    - From vectors to vectors → Map is a Series of Matrices
    - That is, matrix based transformations – Linear Algebra
- Probability
  - Data and results have uncertainty built into them
  - Conditional probability is a very important component of ML
- Optimization → Calculus (Multivariable calculus)
  - For a given set of data what is the "best" model?
  - Many ML models finally reduce to solving some optimization problem

So, some of the mathematical ideas that we will be using in this course are linear algebra. Why do we need linear algebra? Remember that as I said earlier, machine learning involves mapping. It involves mapping of what, from an input vector to an output vector. Now what maps vectors to vectors? This is a series of matrices, okay. So if I take one vector and I have to map to another vector of a different size, I have to use a matrix, okay. Which is why we are going to look at linear algebra.

Again we will only cover very very rudimentary ideas, most of it should be already familiar to you with linear algebra. Okay. Next is probability. So the reason we use probability is whether it is the data that is given to us or the results that we see. You might see a person from far and might not know whether this is quite your friend or not. A person identifying a criminal from a line-up might not be 100 percent sure that this is exactly the criminal that they want. Similarly the machine and need not be completely sure that this image is that of a dog or cat or this tumour is cancerous or not cancerous.

So they have some amount of uncertainty built into them. So we account for this uncertainty using probability theory. A very important component of machine learning is the idea of conditional probability. So in case you do not know it, please do refresh it, we will be looking at it through this course also but this is just a heads up for you, that conditional probability is particularly important. The next idea that we will be looking at is that of optimisation.

The reason we require optimisation is that we have whole bunch of models within machine learning and we want to find out which set of parameters is the best for a given model. Of course when we come to optimisation, automatically you come to differentiation and you come to multivariable calculus. So we will be looking at simple calculus, even though we will not be covering calculus, we will be looking at multivariable ideas such as gradients, etc. which are very important to find out optimisation.

Optimisation is really important because finally most machine learning models actually reduced to just solving some optimisation problems or the other. In fact modern machine learning theory is, extensively it uses optimisation theory.