Understanding Machine Learning

by David Chappell

Course Overview

Course Overview

Hi everybody, I'm David Chappell. Welcome to my course, Understanding Machine Learning. I'm the principal of Chappell & Associates in San Francisco, California, and I'm convinced that the rise of machine learning is among the most important trends of our time. Machine learning underlies many of the services you use today, including things like speech recognition, and recommendations from Amazon, and even whether a grocery store lets you use your credit card for your latest purchase. This course is a quick introduction to machine learning. No prior knowledge is required. The major topics we'll cover include what machine learning is and what it can be used for, the machine learning process, and the basic concepts and terminology of the field. By the end of this course, you'll know enough to go deeper, if you choose to, and to start thinking intelligently about whether machine learning can help your organization. I hope you'll join me to learn about this important topic, with the Understanding Machine Learning course at Pluralsight.

Introduction

Introduction

Here's the truth. You need to understand machine learning. I don't care who you are, I don't care what your job is, you need to know at least the basics of this technology. And here's why. It's because machine learning is becoming so important. Machine learning is a bigger and bigger part of our world every single day. I'm David Chappell, and in this course, I'll introduce you to the basics of machine learning. To do that, we'll walk through a few modules. We'll start by answering the big question, what is machine learning? Then, we'll look at the machine learning process, and we'll end with a closer look at machine learning. When you're done, you'll understand enough about machine learning to go on to watch more courses, or to have intelligent conversation. Ready? Let's go.

What Is Machine Learning?

Getting Started

Let's begin at the beginning. What is machine learning? There's probably no definition that the whole world would agree on, but there certainly are some core concepts. To think about those, think about what machine learning does. The core thing machine learning does is finds patterns in data. It then uses those patterns to predict the future. Some examples, you could use machine learning to detect credit card fraud. Suppose you have data about previous credit card transactions. You could find patterns in that data potentially. That will let you detect when a new credit card transaction is likely to be fraudulent. Or, maybe you want to determine whether a customer is likely to switch to a competitor. Again, you could possibly find patterns in the existing customer data that will help you do that. Or, maybe you want to decide when it's time to do preventive maintenance on a factory robot. Once again, you could look at existing data, you can find patterns that predict when a robot is going to fail. There are lots more, but the core idea is that machine learning lets you find patterns in data, then use those patterns to predict the future.

Finding Patterns

Let's take a step back. What does it mean to learn? For example, how did you learn to read? Well, learning requires identifying patterns. In reading, for instance, you identify letters, and then the patterns of letters together to form words. You then had to recognize those patterns when you saw them again. That's what learning means, just as when you learn to read. And that's what machine learning does with data that we provide. So, for example, suppose I have data about credit card transactions. Suppose I have only four records, each one has three fields; the customer's name, the amount of the transaction, and whether or not it was fraudulent. What's the pattern that this data suggests for fraudulent transactions? Well, it's obvious, isn't it? If the name starts with P, they're a criminal. Well, probably not. The problem with having so little data is that it's easy to find patterns, but it's hard to find patterns that are correct, correct in the sense that they are predictive, they help us understand whether a new transaction is likely to be fraudulent. So suppose I have more data. Now I have more records and I have more fields in each one, now I know where the card was issued, where it was used, the age of the user. Now what's the pattern for fraudulent transactions? Well, turns out that if you look at that, there really is a pattern in this data. It is that a transactions is fraudulent if the cardholder is in their 20s, if the card is issued in the USA, and used in Russia, and the amount is more than $1000. You could have found that pattern, I bet, if you looked at this data for a little while. But once again, do we know that that pattern is truly predictive? Probably not. We don't have enough data. To do this well, you'd have enough data that people just can't find the patterns. You have to use software. That's where machine learning comes in.

Machine Learning in a Nutshell

Machine learning in a nutshell looks like this. You start with data that contains patterns. You then feed that data into a machine learning algorithm, it'd be more than one, that finds patterns in the data. This algorithm generates something called a model. A model is functionality, typically code, that's able to recognize patterns when presented with new data. Applications can then use that model by supplying new data to see if this data matches known patterns, such as supplying data about a new transaction. The model can return a probability of whether this transaction is fraudulent. It knows that because of the patterns. Machine learning in a nutshell.

Why Is Machine Learning So Hot Right Now?

[Autogenerated] Why's machine learning so hot right now? Well, there are several reasons. Ah, big one is that doing the shell learning well requires lots of data which more and more we have. We live in the big data era, right? It requires lots of compute power, which Maur and more we have. We live in the cloud air and it requires effective machine learning algorithms which more? More we have because we have seen researchers spend years decades in this space lording what works. All of these things are now Maura available than ever. And that's a big reason why machine learning is so hot today. Who's interested? Machine learning? Well, you can think about three groups of people who care about this topic. The first is business leaders. They want solutions to business problems, things I've described so far, fraudulent transactions, deciding where the customers are gonna switch or not. All these things these air business problems, good solutions have real business value. The other organizations do things faster, better, cheaper, and so business leaders really want those solutions. This is a good thing because business leaders also have the money to pay for those solutions. Very important software developers also care about this because they want to build better applications. And, as we saw, applications can rely on models. Creativity in machine learning to make better predictions. If you're a software developer, machine learning can help you build smarter APS, even if you're not the one who creates the models, you can just use the models and the third category of people who are really involved in this space, called data scientists who want powerful, easy to use tools. Obvious question here. What is a data scientist? The answer generally is it's someone who knows about three things. Statistics, machine learning software and how to write code typically and ideally AH, problem domain such as credit card transaction fraud or robot preventive maintenance or some other area. There are key things to know about data scientists First, good ones were scarce. Second, unsurprisingly, good ones are expensive. And the reason is if you can solve important business problems with machine learning, you can save a lot of money. There's really business value there. And so good data Scientists who know all three of these things statistics, machine learning and a problem domain can have enormous value. That's why right now that commands substantial salaries are often hard to keep in your employ

The Ethics of Machine Learning

[Autogenerated] Like most technologies, machine learning can raise ethical issues. But the ethics of machine learning are really important. It's critical to understand what some of the possible pitfalls are here. Recall the basic model. We start with data. We processed that data using machine learning algorithms to produce a model. Great the venues that model to make decisions. But what happens if the data is biased? For example, think about bank lending. Suppose we create a model to decide whether or not somebody should get alone. But suppose the data that we used to create that model is from historic loan patterns and suppose that data contains ratial bias. If that's the case, our model will also contain that ratial bias. And we might not even know it because the data could be so large that we couldn't see the bias ourselves. I think about this. Suppose you got a model for doing bank landing, and someone accuses you of having a bias model. How can you explain the model decision? It's important to realize that models generated by machine learning are different from other kinds of software. Traditional software is written directly by people who could work out in great detail exactly what the software does, if you need to. Somebody could look at that code directly to figure out why it behaves in a certain way with the model, though models are generated by the machine learning process. As I've shown you, that process commonly uses complex statistical techniques, and the result is an ordinary computer code. You can't just look at it to see why it's doing what it's doing. The point is that some kinds of models can be very hard to explain, and there are scenarios where you might be required legally required to explain your model. It's important to realize this and to realize this could be a real concern with machine learning. The ethics of machine learning are a very important issue to be aware of.

The Main Points

[Autogenerated] the main points are these. First machine learning lets us find patterns in existing data, then create and use a model that recognizes those patterns in new data. Machine learning has gone mainstream. Lots of organizations use it today in lots of different ways, although it raises ethical concerns. The most important point, though, is this. It is that she learning can probably help your organization.

The Machine Learning Process

Getting Started

Understanding machine learning means understanding the machine learning process, and the machine learning process is iterative. You repeat things over and over, in both big and small ways. The machine learning process also is challenging, typically. It's rarely easy, and the reason is that you're working with what are often large amounts of potentially complex data, and you're trying to find patterns, meaningful patterns, predictive patterns, in this data. This can be hard. It's why we work with specialists, it's why data scientists are often so important to machine learning projects. And finally, the machine learning process is often rewarding. As I've said, the benefits of success here can be substantial. But not always. It's always possible that you will fail; be aware of that. This process is worth doing in many, many cases, but it doesn't always succeed.

Asking the Right Question

The first problem you face in the machine learning process is deciding what question to ask. Asking the right question is really important. In fact, it's fair to say that choosing what question to ask is the most important part of the process. 1 And the reason why this is true is probably obvious. It's that if you ask the wrong question, you won't get the answer you care about. Choosing what question to ask is really important, and then you've got to ask yourself, do you have the right data 2 to answer this question? Maybe, for example, the question you want to ask is how can I predict whether a credit card transaction is going to be fraudulent? Well, maybe it's the case that the most predicted piece of data for doing this is whether the customer is a homeowner or a renter. Or maybe it's how long they live at a current address. You might not have this data, and you won't know this until some later point, if ever. So you want to ask yourself, do you think you have the right data to answer this question? Because if you don't, you won't get an answer you like. 3 You also want to ask yourself this, do you know how you'll measure success? Because ultimately what you're going to get is a model that makes predictions. How good must those predictions be to make this entire process qualify as a success? For example, for credit card transactions, if you find that you're accurate about fraud prediction in, say, 8 out of 10 cases, is that good enough? How about 6 out of 10? Do you demand 9 out of 10? How do you decide? Knowing this up front is important, because if you don't, you will never know when you're done.

Illustrating the Machine Learning Process

1 Let's look at the machine learning process in a little more detail. To start, you choose the data that you want to work with. You often are going to work with domain experts in the area to do this, people who know a lot about, say, transaction fraud or robot failure detection, or whatever problem you're trying to solve. These are the ones who know what data is most likely to be predictive. But the data you start with, the raw data, is almost never in the right form. It has duplicates, it has missing data, it has extra stuff. Typically you've got to apply some pre- 2 processing to that data, and machine learning products commonly provide a variety of data pre-processing modules to do this. The result is some prepared data, data that's been worked on to be more appropriate as an input for machine learning. Do you do this just once? Oh, no. You commonly iterate until the data is ready. The truth here is that in typical machine learning projects, you'll spend most of your time right here, working on the data, getting it ready, getting it clean, getting it prepared. Once you have that data, you can then begin applying learning 3 algorithms to the data. And again, machine learning products commonly provide a number of machine learning algorithms. The result of this is a model, but is it your final model? No. It's a

candidate model. Is the first model you create the best one? Almost certainly not, and you can't know that until you've produced several, and so once again, you iterate. As I said before, this process is iterative. You do this until you have a model that you like, that you think is good enough to actually deploy. Once you deploy the model, applications can now make use of it. So, there's iteration at small levels, as you can see here, and there's also iteration at the largest level. You've got to repeat the entire process over and over, you've got recreate your model regularly. Why is that? It's because the world changes, and so you need to keep your model up-to-date with reality. That might mean processing new data or new algorithms or something else. But recognize that you need to recreate your model regularly. This process is iterative at both small and large scales.

Example Machine Learning Scenarios

[Autogenerated] to put all of this in context, let's walk through a few scenarios illustrating how you might actually use machine learning. Let's start with the example I've been using throughout this course detecting credit card fraud. Suppose you have some number of credit card customers who are supplying their credit cards to sum payment application. Maybe it's a point of sale terminal at a grocery store, and they're providing Visa or MasterCard. The challenge is to work out which of those transactions the application should reject because they're like, be fraudulent. To do this as we've seen, we could start with historical transaction data, run that through the machine learning process and get a model that the application can call to make this decision simple, straightforward. Here's a slightly more complex city area. Suppose your problem is predicting customer churn. Imagine that your mobile phone company, for example, you got customers who call into a call center whose staff relies on some call centre application. Here's what you want to do for each caller. You want the call center staff to be able to figure out how likely that customer is to churn that is switched to a competitors. There's real value here because you might offer a customer who's about to switch a better deal than one who's loyal. There's a real business value in doing this, and it might seem like magic. But in fact, machine learning can help with this, at least in some cases for mobile phone company, for example, they have lots of very detailed call data about their customers. Imagine that in this case that data is just too for detailed, and so they create on application toe aggregate it. And that application might use some big data technology like Hadoop or Spark or something else. The phone company might then combine this area to call data with other data, such as data from their CR M system to actually create the data that machine learning wants to use. That's very common. It's common to use data from different sources as input to the machine learning process. The result is a model that the call center application can use to make good estimations of whether or not a given customer is going to turn. This is a really example. Firms really do this today, and it has really value. It also illustrates another important thing, which is that machine learning is commonly used in concert with other data technologies. In this case, it's Hadoop or Spark, or something like that. There could be others as well. And here's one more scenario. Suppose you got a bunch of devices, robots, former stats, whatever that generate. Lots of streaming data that's being handled by some kind of real time data processing software. That software is looking for anomalies or patterns that predict imminent failure. When it finds these things, it contacts some application, which then notifies business users to take action, such as performing proactive maintenance. Well, the obvious question is, how does that really time data processing software know what to look for? How could it tell when the device is about to fail, for example? Well, the answer could well be that you've got a historical database from the data these devices have produced already. You use that data as input to machine learning and create a model that the real time data processing software can use once again. Ah, pragmatic solution to a very real problem that depends on good data and machine learning. There are many other ways that machine learning is used, for example. But she learning underlies the recommendation engines that you see in video sites, shopping sites and so on the software that recommends other things you might like if you've purchased this or watch this or whatever. But she and learning also underlies much of modern speech recognition. This wasn't true originally, but machine learning approaches turn out to be very effective at doing speech recognition. Similarly, Monsieur Learning underlies language translation. Things like Google translate and well, more important example is facial recognition. Machine learning is very good at creating software that recognizes faces now recall. I mentioned earlier that machine learning can bring some serious ethical issues. This is an area where that's exactly true. Facial recognition well, very useful in some circumstances can also be great for surveillance. And there are plenty of people who don't much like the idea of living in a country where that is ubiquitous. The point is, this approach to building software is incredibly useful. It has been, is being used in all sorts of ways expected to be used and lots more going forward

The Main Points

The machine learning process begins when you ask the right question. You then need to choose the right data to answer that question, and get that data into good shape. So I mentioned earlier, this part of the process typically takes a majority of the time. Once you have the right prepared data, you iterate on that data until you have a model that makes good predictions. You then periodically rebuild that model, because you want to reflect the world as it is. And, of course, you deploy the model. If it can't be used, this is just a science project, so deploying the model is a very important part of the process. In the next module, we'll look in more detail at the machine learning process. There's more you need to know, so don't go away.

A Closer Look at the Machine Learning Process

Getting Started

It's time we took a closer look at machine learning. In this last module, I want to talk about machine learning concepts in a somewhat more detailed way. I also want to use the terminology that a machine learning person would use. So, we're going to talk about training data. We're going to talk about supervised and unsupervised learning. We're going to look at how we classify machine learning problems and algorithms. I'll discuss training a model, which actually means something very simple, as you'll see. We'll look at testing a model, and finally, we'll talk just a little more about using a model.

Some Terminology

The first thing we need to do is walk through some terminology. Like most fields, machine learning has its own unique jargon, which you must understand. Let's start with the idea of training data. Training data just means the prepared data that's used to create a model. So rather than prepared data, I will from now on refer to training data. Why is it called training data? Because in the jargon of machine learning, creating a model is called training a model. So, training data is used to train to create a model. Also, there are two big broad categories of machine learning. One is called supervised learning, and what it means is that the value you want to predict is actually in the training data. For instance, in the example I've been using of data for predicting credit card fraud, whether or not a given transaction was fraudulent is actually contained in each record. That data in the jargon of machine learning is labeled, and so we're doing what's called supervised learning when we try to predict whether a new transaction is fraudulent. The alternative, unsurprisingly, is called unsupervised learning, and here the value you want to predict is not in the training data. The data is unlabeled. Both approaches are used, but it's fair to say that the most common approach is supervised learning.

Data Pre-processing

The machine learning process starts with data. It might be relational data, it might be from a NoSQL database, it might be binary data. Wherever it comes from, though, you need to read this raw data into some data preprocessing modules typically chosen from the things your machine learning technology provides. You have to do this because raw data is very rarely in the right shape to be processed by machine learning algorithms. As I said earlier, you'll spend lots of your time, often the majority of your time, in a machine learning project on this aspect of the process. For example, maybe there are holes in your data, missing values, or duplicates, or maybe there's redundant data where the same thing is expressed in two different ways in different fields, or maybe there's information that you know will not be predictive, it won't help you create a good model. You want to deal with all of these issues. The goal is to create training data. The training data, as we saw in my simple example earlier, commonly has columns. Those columns are called features. So, for example, in the simple illustration I showed of data for credit card fraud, there were columns containing the country the card was issued in, the country was card was used in, the amount of the transaction. Those are all features in the jargon of machine learning. And because we're talking now about supervised learning, the value we're trying to predict, such as whether a given transaction is fraudulent, is also in the training data. In the jargon of machine learning, we call that the target value.

Categorizing Machine Learning Problems

It's common to group machine learning problems into categories. There are lots of categories, but three of them show up an awful lot. One of those is the category called regression. The problem here is that we have data, and we'd like to find a line or a curve that best fits that data. Regression problems are typically supervised learning scenarios, and an example question would be something like, how many units of this product will we sell next month? A second category of machine learning problems is called classification. Here we have data that we want to group into classes, at least two, sometimes more than two. When new data comes in, we want to determine which class that data belongs to. This is commonly used with supervised learning, and an example question would be something like, is this credit card transaction fraudulent? The example I've been using throughout this course fits here. Because when a new transaction comes in, we want to predict which class it's in, fraudulent or not fraudulent. And often what you'll get back is not yes or no, but a probability of which class this new transaction might be in. A third category on machine learning problems is commonly called clustering. Here we have data, we want to find clusters in that data. This is a good example of when we're going to use unsupervised learning, because we don't have labeled data. We don't know necessarily what we're looking for. An example question here is something like, what are our customer segments? We might not know these things up front, but we can use machine learning, unsupervised machine learning, to help us figure that out.

Styles of Machine Learning Algorithms

[Autogenerated] the kinds of problems that machine learning addresses aren't the only thing that can be categorized. It's also useful to think about the styles of machine learning algorithms that are used to solve those problems. For example, there are regression algorithms, which are good option for regression problems. There are decision trees, which are sometimes group together into decision forests, their own neural networks, which imitate in some ways how the brain works. One particular way of using neural networks is known as deep learning, something you might have heard of. They're also Beijing algorithms that use based serum to work of probabilities. And there are K means algorithms that are used for clustering. They're also lots more. The details of these algorithms are way outside the scope of this course. They get complicated fast, but having some broad sense of what the styles are is useful

Training and Testing a Model

Let's take a closer look at the process of creating a model, of training a model. We start with our training data, which we've worked with until it's beautiful, pristine, just what we need. Because we're using supervised learning, the target value is part of the training data. In the case of the credit card example, for instance, that target value is whether a transaction is fraudulent or not. Our first problem is to choose the features that we think will be most predictive of that target value. For example, in the credit card case, maybe we decide that the country in which the card was issued, the country it's used in, and the age of the user are the most likely features to help us predict whether it's fraudulent. We've chosen, let's say, features 1, 3, and 6 in our training data. We then input that training data into our chosen learning algorithm. But notice this. We only send in 75%, say, of all the data for the features we've chosen. Why is that? I'll tell you in a minute. But first, think about this. How do we decide which features were most predictive, and how do we choose a learning algorithm? There are lots of options as we've seen. The answer is, if it's a simple problem, or maybe our technology is simple for machine learning, the choices can be limited, not too hard. If we have a more complex problem, though, with lots of data and a powerful machine learning technology with lots of algorithms, this can be hard. If we have, for example, training data that has, I don't know, how about 100 features, how about 200? Which ones are predictive? How many should we use? 5, 10, 50? The answer is this is what data scientists are for. This is why people who have knowledge and facility with these technologies, as well as domain knowledge about some particular problem, are so valuable. It's because they can help us do this. It can be a hard problem. In any case, the result of this is to generate a candidate model. The next problem is to work out whether or not this model is any good. And so, we do that in supervised learning like this. We input test data to a candidate model. That test data is the remaining 25%, the data we held back for the features we're using, in this case, 1, 3, and 6. We use that data, because our candidate model can now generate target values from that test data. But here's the thing. We know what those target values should be, because they are in the training data. All we have to do is compare the target values produced by our candidate model from the test data with the real target values, which are in the training data. That's how we could figure out whether or not our model is predictive or not when we're doing supervised learning. Suppose it's not. Suppose our model's just not very good. How can we improve it? Well, there are some usual options. One of them is, maybe we've chosen the wrong 1 features. Let's choose different ones. How about 1, 2, and 5 this time? Or maybe it's the case that we have the 2 wrong data, 3 let's get some new data, or at least some more example data. Or maybe the problem is the algorithm. Maybe it's the case that we can modify some parameters in our algorithm, they commonly have them, or choose another one entirely. Whatever we do will generate another candidate model, and we'll test it, and the process repeats. It 4 iterates. Now, iteration is a fancy way of saying trial and error. So, don't be confused. This process is called machine learning, but notice how much people do. People make decisions about features, about algorithms, about parameters. The process is very human, even though it's called machine learning.

Using a Model

[Autogenerated] There's one last thing for us to talk about, and that's a using a model. In some ways, this is the most important topic of all, because until models air used that really have much value. An application, for example, can call a model, providing values for the features the ~~Mahler~~ model requires. Remember, models make predictions based on the features that were chosen when the model was trained. The ~~Molik~~ then return the value predicted using these features. That value might be whether ~~introduction~~ is fraudulent estimated revenue, a list of movie recommendations or something else. The point here is that machine learning can help people create better applications.

Implementing Machine Learning

[Autogenerated] the focus of this course is on concepts, not implementation of machine learning. Still, it's good to have some basic knowledge of how people implement this stuff. So here are a few examples. One approach is to create custom models in languages like R and Python. Visit both popular, and they both have lots of general machine learning packages available. They're also both open source. Another option is to use somewhat more specialized, more focused packages to create custom models. Among the most well known of these is Tensorflow. Tensorflow has built in support for deep learning, which, as I mentioned earlier, uses neural networks. Another choice is to rely on the cloud service is that we're seeing appear for machine learning, such as Amazon Sage Maker. These provide complete end end solutions, and they let you use other things as well, like python and tensorflow. And finally, one more possibility is to realize that you can use machine learning without ever building your own model. Turns out there are lots of fairly standard problems for which predefined models already exist, such as those in azure. Cognitive service is if you're doing image processing or speech recognition or building a recommendations engine. You might just be able to use a pre existing model provided for you by typically a cloud service and not have to build your own at all. If this is possible, it's generally the best choice.

Summary

[Autogenerated] and that's it. Let me end by summarizing the key points. First machine learning has come of age. It's no longer some technology that's only for researchers in faraway labs. Machine learning also isn't hard to understand. I hope you agree after watching this, although it can be hard to do well. Finally, I said this before, but I want to say it again because in some ways it might be the most important point. I'm making the entire course. It is this. It is that machine learning can probably help your organization. I'm David Chappelle. Thanks for watching.

Course author

David Chappell

David Chappell is Principal of Chappell & Associates in San Francisco, California. Through his speaking, writing, and consulting, he helps people around the world understand, use, and make better...

Course info

Level	Beginner
Rating	★★★★☆ (1801)
My rating	★★★★★
Duration	0h 43m
Updated	23 Sep 2019

Share course