Go to next chapter ›

Chapter 1. Introduction to machine learning

This chapter covers

- What machine learning is
- Supervised vs. unsupervised ma
- Classification, regression, dimer and clustering
- Why we're using R
- Which datasets we will use

You interact with machine learning on a daily basis whether you recognize it or not. The advertisements you see online are of products you're more likely to buy based on the things you've previously bought or looked at. Faces in the photos you upload to social media platforms are automatically identified and tagged. Your car's GPS predicts which routes will be busiest at certain times of day and replots your route to minimize journey length. Your email client progressively learns which emails you want and which ones you consider spam, to make your inbox less cluttered; and your home personal assistant recognizes your voice and responds to your requests. From small improvements to our daily lives such as these, to big, society-changing ideas such as self-driving cars, robotic surgery, and automated scanning for other Earth-like planets, machine learning has become an increasingly important part of modern life.

But here's something I want you to understand right away: machine learning isn't solely the domain of large tech companies or computer scientists. ***Anyone*** with basic programming skills can implement machine learning in their work. If you're a scientist,

machine learning can give you extraordinary insights into the phenomena you're studying. If you're a journalist, it can help you understand patterns in your data that can delineate your story. If you're a businessperson, machine learning can help you target the right customers and predict which products will sell the best. If you're someone with a question or problem, and you have sufficient data to answer it, machine learning can help you do just that. While you won't be building intelli after reading this book (like Google and gained the skills to make powerful predi informative patterns in your data.

I'm going to teach you the theory and p at a level that anyone with a basic know since high school, I've been terrible at n expect you to be great at it either. Altho about to learn are based in math, I'm a f no hard concepts in machine learning. All of the processes we'll explore together will be explained graphically and intuitively. Not only does this mean you'll be able to apply and understand these processes, but you'll also learn all this without having to wade through mathematical notation. If, however, you are mathematically minded, you'll find equations presented through the book that are "nice to know," rather than "need to know."

In this chapter, we're going to define what I actually mean by *machine learning*. You'll learn the difference between an algorithm and a model, and discover that machine learning techniques can be partitioned into types that help guide us when choosing the best one for a given task.

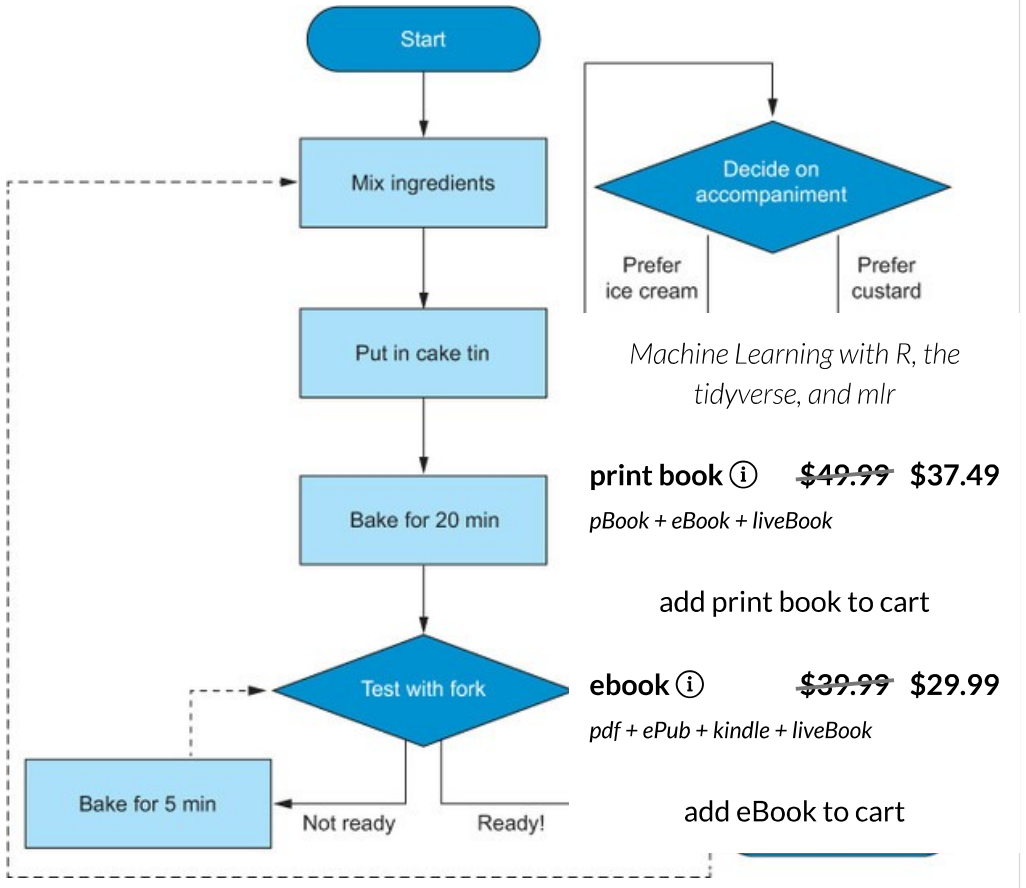
1.1. What is machine learning?

Imagine you work as a researcher in a hospital. What if, when a new patient is checked in, you could calculate the risk of them dying? This would allow the clinicians to treat high-risk patients more aggressively and result in more lives being saved. But where would you start? What data would you use? How would you get this information from the data? The answer

Machine learning, sometimes referred t
subfield of artificial intelligence (AI) wh
patterns in data to perform specific task
may sound complicated, they aren't. In
algorithm is not complicated at all. An a
by-step process that we use to achieve s
beginning and an end. Chefs have a diffe
they call them "recipes." At each stage i
some kind of process, like beating an egg, and then you follow the next instruction in the recipe, such as mixing the ingredients.

Have a look in figure 1.1 at an algorithm I made for making a cake. It starts at the top and progresses through the various operations needed to get the cake baked and served up. Sometimes there are decision points where the route we take depends on the current state of things, and sometimes we need to go back or *iterate* to a previous step of the algorithm. While it's true that extremely complicated things can be achieved with algorithms, I want you to understand that they are simply sequential chains of simple operations.

Figure 1.1. An algorithm for making and serving a cake. We start at the top and, after performing each operation, follow the next arrow. Diamonds are decision points, where the arrow we follow next depends on the state of our cake. Dotted arrows show routes that iterate back to previous operations. This algorithm takes ingredients as its input and outputs cake with either ice cream or custard!



So, having gathered data on your patients, you train a machine learning algorithm to learn patterns in the data associated with the patients' survival. Now, when you gather data on a new patient, the algorithm can estimate the risk of that patient dying.

As another example, imagine you work for a power company, and it's your job to make sure customers' bills are estimated accurately. You train an algorithm to learn patterns of data associated with the electricity use of households. Now, when a new household joins the power company, you can estimate how much money you should bill them each month.

Finally, imagine you're a political scientist, and you're looking for types of voters that no one (including you) knows about. You train an algorithm to identify patterns of voters in survey data, to better understand what motivates voters for a particular political party. Do you see any similarities between these problems and the problems you would like to solve? Then—provided the solution is

hidden somewhere in your data—you can train a machine learning algorithm to extract it for you.

1.1.1. AI and machine learning

Arthur Samuel, a scientist at IBM, first used the term ***machine learning*** in 1959. He used it to describe a form of AI that involved training an algorithm to learn to play th
word ***learning*** is what's important here,
distinguishes machine learning approac

Traditional AI is programmatic. In othe
computer a set of rules so that when it e
knows precisely which output to give. A
using `if else` statements to classify a
snakes:

```
numberOfLegs <- c(4, 4, 0)
climbsTrees <- c(TRUE, FALSE, TRUE)

for (i in 1:3) {
  if (numberOfLegs[i] == 4) {
    if (climbsTrees[i]) print("cat") else print("dog")
  } else print("snake")
}
```

copy

In this R code, I've created three rules, mapping every possible input available to us to an output:

1. If the animal has four legs and climbs trees, it's a cat.
2. If the animal has four legs and does not climb trees, it's a dog.
3. Otherwise, the animal is a snake.

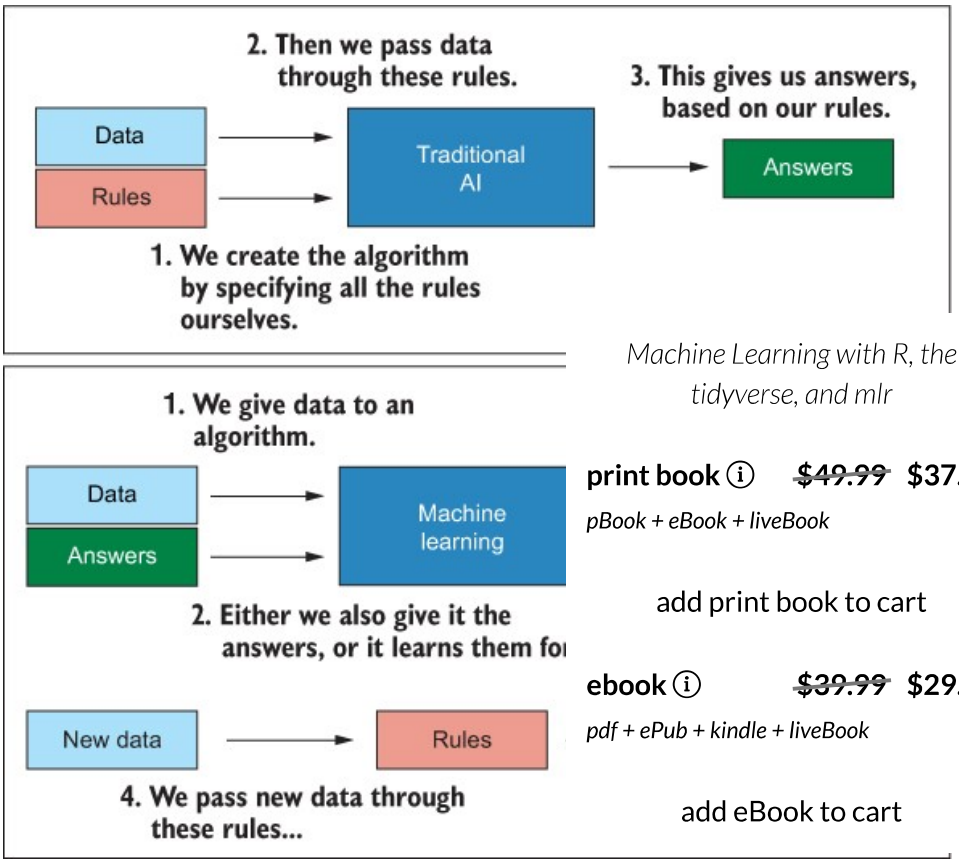
Now, if we apply these rules to the data, we get the expected answers:

```
[1] "cat"
[1] "dog"
[1] "snake"
```

copy

The problem with this approach is that
all the possible outputs the computer sh
will never give us an output that we hav
this with the machine learning approacl
the computer the rules, we give it the da
rules for itself. The advantage of this ap
can "learn" patterns we didn't even knc
the more data we provide, the better it g
patterns (figure 1.2).

Figure 1.2. Traditional AI vs. machine learning AI. In traditional AI applications, we provide the computer with a complete set of rules. When it's given data, it outputs the relevant answers. In machine learning, we provide the computer with data and the answers, and it learns the rules for itself. When we pass new data through these rules, we get answers for this new data.



1.1.2. The difference between a model and an algorithm

In practice, we call a set of rules that a machine learning algorithm learns a *model*. Once the model has been learned, we can give it new observations, and it will output its predictions for the new data. We refer to these as models because they represent real-world phenomena in a simplistic enough way that we and the computer can interpret and understand it. Just as a model of the Eiffel Tower may be a good representation of the real thing but isn't exactly the same, so statistical models are attempted representations of real-world phenomena but won't match them perfectly.

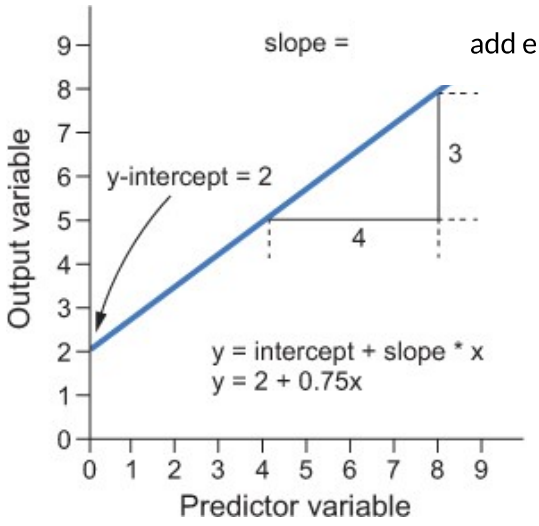
NOTE

You may have heard the famous phrase coined by the statistician George Box that "All models are wrong, but some are useful"; this refers to the approximate nature of models.

The process by which the model is learned is referred to as the

algorithm. As we discovered earlier, an algorithm is just a sequence of operations that work together to solve a problem. So how does this work in practice? Let's take a simple example. Say we have two continuous variables, and we would like to train an algorithm that can predict one (the *outcome* or *dependent* variable) given the other (the *predictor* or *independent* variable). The relationship between these variables can be described by a straight line that can be defined using only two parameters: its s
the y-axis (the y-intercept). This is sho

**Figure 1.3. Any straight line can be de
change in y divided by the change in *x*
it crosses the y-axis when *x* = 0). The e
slope * *x* can be used to predict the va**



An algorithm to learn this relationship could look something like the example in figure 1.4. We start by fitting a line with no slope through the mean of all the data. We calculate the distance each data point is from the line, square it, and sum these squared values. This *sum of squares* is a measure of how closely the line fits the data. Next, we rotate the line a little in a clockwise direction and measure the sum of squares for *this* line. If the sum of squares is bigger than it was before, we've made the fit worse, so we rotate the slope in the other direction and try again. If the sum of squares gets smaller, then we've made the fit better. We continue with this process, rotating the slope a little less each time we get closer, until

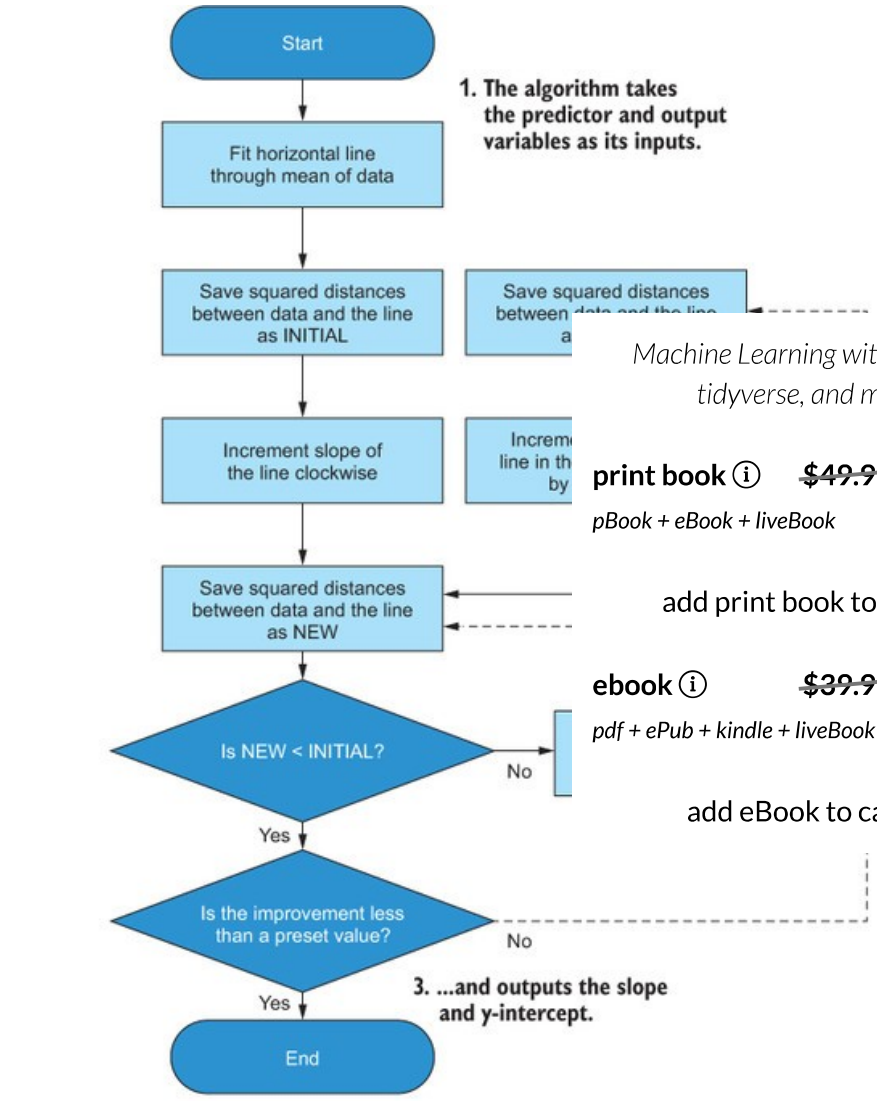
the improvement on our previous iteration is smaller than some preset value we've chosen. The algorithm has iteratively learned the model (the slope and y-intercept) needed to predict future values of the output variable, given only the predictor variable. This example is slightly crude but hopefully illustrates how such an algorithm could work.

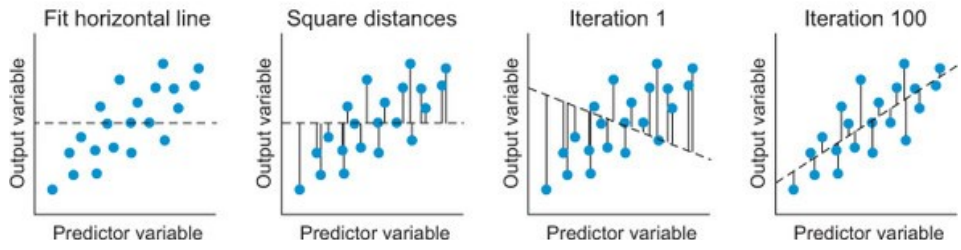
NOTE

One of the initially confusing but ever machine learning is that there is a ple solve the same type of problem. The r people have come up with slightly dif same problem, all trying to improve u For a given task, it is our job as data sc algorithm(s) will learn the best-perfo

While certain algorithms tend to perform better than others with certain types of data, no single algorithm will always outperform all others on all problems. This concept is called the ***no free lunch theorem***. In other words, you don't get something for nothing; you need to put some effort into working out the best algorithm for your particular problem. Data scientists typically choose a few algorithms they know tend to work well for the type of data and problem they are working on, and see which algorithm generates the best-performing model. You'll see how we do this later in the book. We can, however, narrow down our initial choice by dividing machine learning algorithms into categories, based on the function they perform and how they perform it.

Figure 1.4. A hypothetical algorithm for learning the parameters of a straight line. This algorithm takes two continuous variables as inputs and fits a straight line through the mean. It iteratively rotates the line until it finds a solution that minimizes the sum of squares. The parameters of the line are output as the learned model.





1.2. Classes of machine learning algorithms

All machine learning algorithms can be categorized by their learning type and the task they perform. There are three learning types:

- Supervised
- Unsupervised
- Semi-supervised

The type depends on how the algorithm to hold their hand through the learning the answers for themselves? Supervised algorithms can be further split into two

- Supervised
    - Classification
    - Regression
- Unsupervised
    - Dimension reduction
    - Clustering

The class depends on what the algorithms learn to *do*.

So we categorize algorithms by how they learn and what they learn to do. But why do we care about this? Well, there are ***a lot*** of machine learning algorithms available to us. How do we know which one to pick? What kind of data do they require to function properly? Knowing which categories different algorithms belong to makes our job of selecting the most appropriate ones much simpler. In the next section, I cover how each of the classes is defined and why it's different from the others. By the end of this section, you'll have a clear understanding of why you would use algorithms from one class over another. By the end of the book, you'll have the skills to apply a number of algorithms from each class.

1.2.1. Differences between supervised, unsupervised, and semi-supervised learning

Imagine you are trying to get a toddler to learn about shapes by using blocks of wood. In front of them, they have a ball, a cube, and a star. You ask them to show you the cube, and if they point to the correct shape, you tell them they are correct; if they are incorrect, you also tell them. You repeat this procedure until the toddler can identify the correct shape almost all of the time. This is called ***supervised learning***, because you, the person who already knows which shape is which, are supervising the learning and providing the answers.

Now imagine a toddler is given multiple this time is also given three bags. The tc in one bag, the cubes in another bag, an you won't tell them if they're correct— themselves from nothing but the inform them. This is called ***unsupervised learnin*** to identify patterns themselves with no

A machine learning algorithm is said to be ***supervised*** if it uses a ground truth or, in other words, ***labeled data***. For example, if we wanted to classify a patient biopsy as healthy or cancerous based on its gene expression, we would give an algorithm the gene expression data, labeled with whether that tissue was healthy or cancerous. The algorithm now knows which cases come from each of the two types, and it tries to learn patterns in the data that discriminate them.

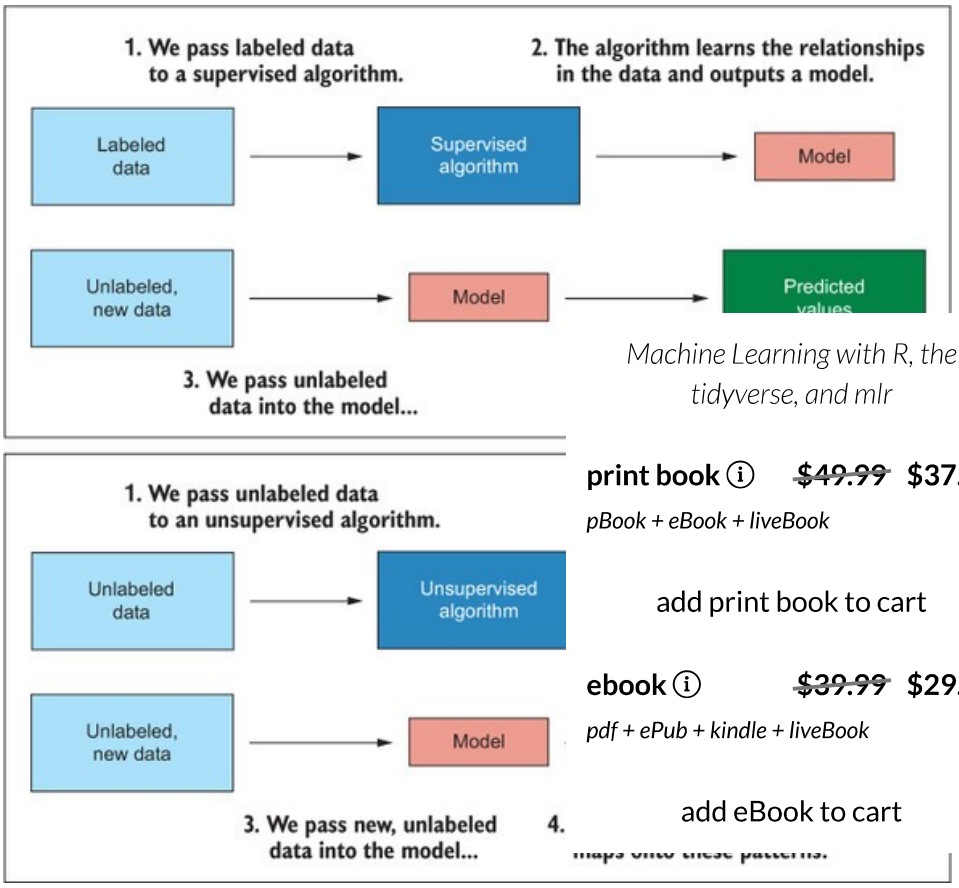
Another example would be if we were trying to estimate a person's monthly credit card expenditure. We could give an algorithm information about other people, such as their income, family size, whether they own their home, and so on, including how much they typically spent on their credit card in a month. The algorithm looks for patterns in the data that can predict these values in a reproducible way. When we collect data from a new person, the algorithm can estimate how much they will spend, based on the patterns it learned.

A machine learning algorithm is said to be ***unsupervised*** if it does not use a ground truth and instead looks on its own for patterns in

the data that hint at some underlying structure. For example, let's say we take the gene expression data from lots of cancerous biopsies and ask an algorithm to tell us if there are clusters of biopsies. A ***cluster*** is a group of data points that are similar to each other but different from data in other clusters. This type of analysis can tell us if we have subgroups of cancer types that we may need to treat differently.

Alternatively, we may have a dataset wi variables—so many that it is difficult to for relationships manually. We can ask a way of representing this high-dimensio dimensional one, while maintaining as original data as possible. Take a look at your algorithm uses labeled data (a grou supervised, and if it does not use labele unsupervised.

Figure 1.5. Supervised vs. unsupervised machine learning. Supervised algorithms take data that is already labeled with a ground truth and build a model that can predict the labels of unlabeled, new data. Unsupervised algorithms take unlabeled data and learn patterns within it, such that new data can be mapped onto these patterns.



SEMI-SUPERVISED LEARNING

Most machine learning algorithms will fall into one of these categories, but there is an additional approach called *semi-supervised* learning. As its name suggests, semi-supervised machine learning is not quite supervised and not quite unsupervised.

Semi-supervised learning often describes a machine learning approach that combines supervised and unsupervised algorithms together, rather than strictly defining a class of algorithms in and of itself. The premise of semi-supervised learning is that, often, labeling a dataset requires a large amount of manual work by an expert observer. This process may be very time consuming, expensive, and error prone, and may be impossible for an entire dataset. So instead, we expertly label as many of the cases as is feasibly possible, and then we build a supervised model using only the labeled data. We pass

the rest of our data (the unlabeled cases) into the model to get their predicted labels, called *pseudo-labels* because we don't know if all of them are actually correct. Now we combine the data with the manual labels and pseudo-labels, and use the result to train a new model.

This approach allows us to train a mod... labeled and unlabeled data, and it can ... predictive performance because we ar... data at our disposal. If you would like ... semi-supervised learning after compl... *Semi-Supervised Learning* by Olivier C... Scholkopf, and Alexander Zien (MIT P... reference may seem quite old, but it is ...

Within the supervised and unsupervised ... learning algorithms can be further categ... perform. Just as a mechanical engineer knows which tools to use for the task at hand, so the data scientist needs to know which algorithms they should use for their task. There are four main classes to choose from: classification, regression, dimension reduction, and clustering.

1.2.2. Classification, regression, dimension reduction, and clustering

Supervised machine learning algorithms can be split into two classes:

- *Classification algorithms* take labeled data (because they are supervised learning methods) and learn patterns in the data that can be used to predict a *categorical* output variable. This is most often a *grouping variable* (a variable specifying which group a particular case belongs to) and can be *binomial* (two groups) or *multinomial* (more than two groups). Classification problems are very common machine learning tasks. Which customers will default on their payments? Which patients will survive? Which objects in a telescope image are stars, planets, or galaxies?

When faced with problems like these, you should use a classification algorithm.

- *Regression algorithms* take labeled data and learn patterns in the data that can be used to predict a *continuous* output variable. How much carbon dioxide does a household contribute to the atmosphere? What will the share price of a company be tomorrow? What is the concentration of insulin in a patient's blood? When t
like these, you should use a regress

Unsupervised machine learning algorith
two classes:

- *Dimension-reduction algorithms* tak
they are unsupervised learning me
dimensional data (data with many
way of representing it in a lower nu
Dimension-reduction algorithms may be used as an exploratory technique (because it's very difficult for humans to visually interpret data in more than two or three dimensions at once) or as a preprocessing step in the machine learning pipeline (it can help mitigate problems such as *collinearity* and the *curse of dimensionality*, terms I'll define in later chapters). Dimension-reduction algorithms can also be used to help us visually confirm the performance of classification and clustering algorithms (by allowing us to plot the data in two or three dimensions).
- *Clustering algorithms* take unlabeled data and learn patterns of clustering in the data. A *cluster* is a collection of observations that are more similar to each other than to data points in other clusters. We assume that observations in the same cluster share some unifying features that make them identifiably different from other clusters. Clustering algorithms may be used as an exploratory technique to understand the structure of our data and may indicate a grouping structure that can be fed into classification algorithms. Are there subtypes of patient responders in a clinical trial? How many classes of respondents were there in the survey? Do different types

of customers use our company? When faced with problems like these, you should use a clustering algorithm.

See figure 1.6 for a summary of the different types of algorithms by type and function.

By separating machine learning algorith you will find it easier to select appropria hand. This is why the book is structured classification, then regression, then dim clustering, so you can build a clear men of available algorithms for a particular a class of algorithm to choose from is usu

- If you need to predict a categorical classification algorithm.
- If you need to predict a continuous variable, use a regression algorithm.
- If you need to represent the information of many variables with fewer variables, use dimension reduction.
- If you need to identify clusters of cases, use a clustering algorithm.

1.2.3. A brief word on deep learning

If you've done more than a little reading about machine learning, you have probably come across the term *deep learning*, and you may have even heard the term in the media. Deep learning is a subfield of machine learning (all deep learning is machine learning, but not all machine learning is deep learning) that has become extremely popular in the last 5 to 10 years for two main reasons:

- It can produce models with outstanding performance.
- We now have the computational power to apply it more broadly.

Deep learning uses *neural networks* to learn patterns in data, a term referring to the way in which the structure of these models superficially resembles neurons in the brain, with connections to

pass information between them. The relationship between AI, machine learning, and deep learning is summarized in figure 1.7.

Figure 1.6. Classification, regression, dimension reduction, and clustering. Classification and regression algorithms build models that predict categorical and continuous variables of unlabeled, new data, respectively. Dir algorithms create a new representati fewer dimensions and map new data Clustering algorithms identify cluster new data onto these clusters.

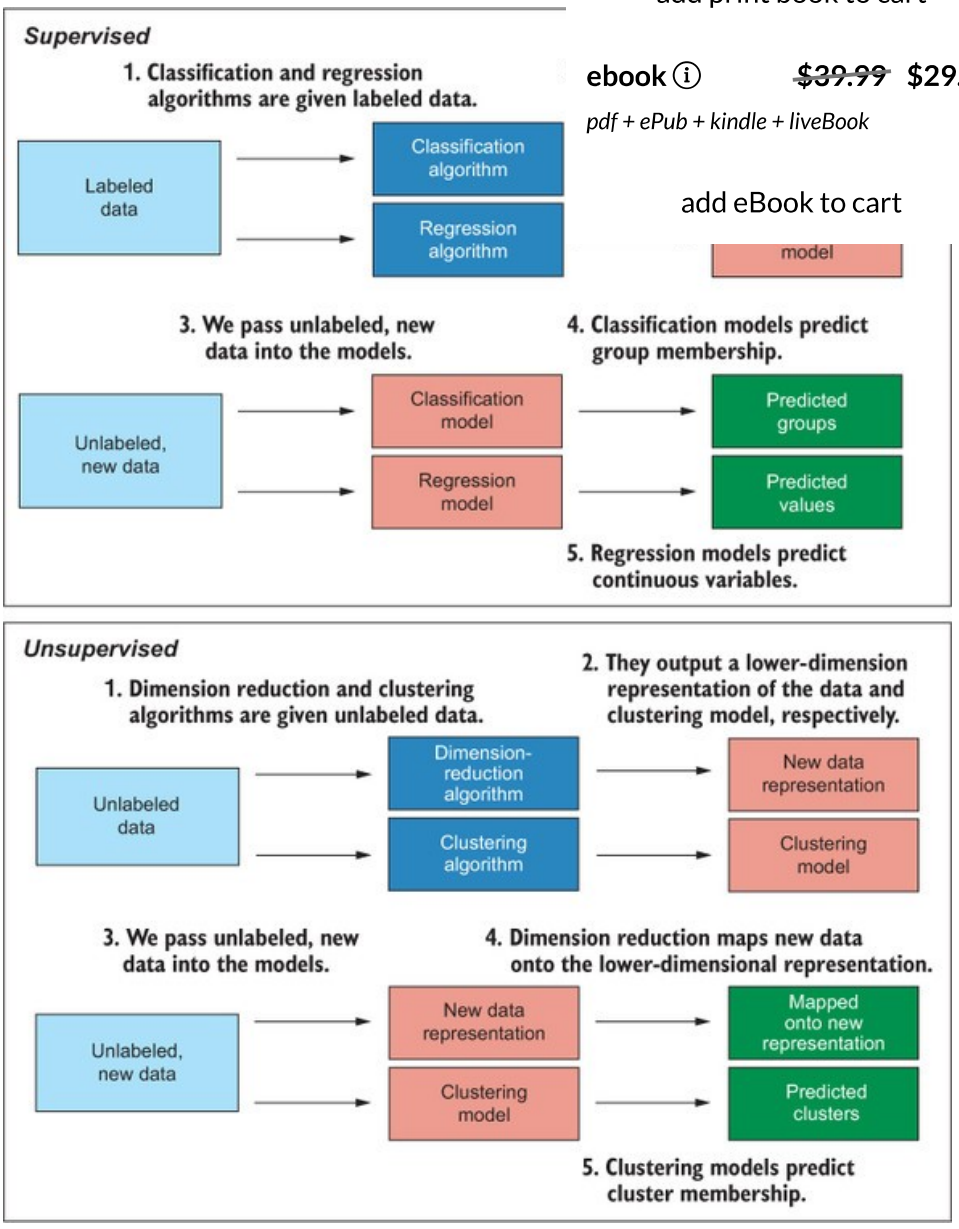
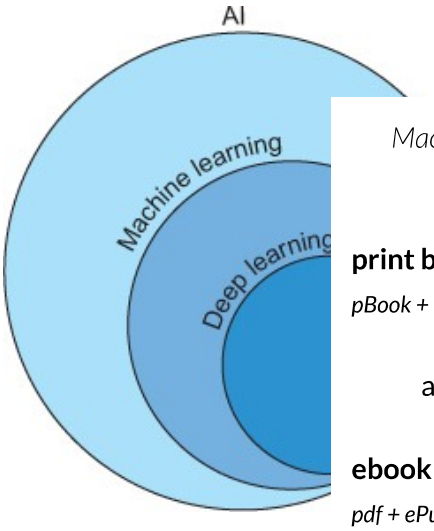


Figure 1.7. The relationship between artificial intelligence (AI), machine learning, and deep learning. Deep learning comprises a collection of techniques that form a subset of machine learning techniques, which themselves are a subfield of AI.



While it's true that deep learning methods will typically outperform "shallow" learning methods (a term sometimes used to distinguish machine learning methods that are not deep learning) for the same dataset, they are not always the best choice. Deep learning methods often are not the most appropriate method for a given problem for three reasons:

- ***They are computationally expensive.*** By expensive, we don't mean monetary cost, of course: we mean they require a lot of computing power, which means they can take a long time (hours or even days!) to train. Arguably this is a less important reason not to use deep learning, because if a task is important enough to you, you can invest the time and computational resources required to solve it. But if you can train a model in a few minutes that performs well, then why waste additional time and resources?
- ***They tend to require more data.*** Deep learning models typically require hundreds to thousands of cases in order to perform extremely well. This largely depends on the complexity of the problem at hand, but shallow methods tend to perform better on small datasets than their deep

learning counterparts.

- ***The rules are less interpretable.*** By their nature, deep learning models favor performance over model interpretability. Arguably, our focus should be on performance; but often we're not only interested in getting the right output, we're also interested in the rules the algorithm learned because these help us to interpret things about the real world and ma research. The rules learned by a neu easy to interpret.

So while deep learning methods can be shallow learning techniques are still inv of data scientists.

NOTE

Deep learning algorithms are particularly good at tasks involving complex data, such as image classification and audio transcription.

Because deep learning techniques require a lot of additional theory, I believe they require their own book, and so we will not discuss them here. If you would like to learn how to apply deep learning methods (and, after completing this book, I suggest you do), I strongly recommend *Deep Learning with R* by Francois Chollet and Joseph J. Allaire (Manning, 2018).

1.3. Thinking about the ethical impact of machine learning

Machine learning can be a force for good, whether that's helping people understand nature or assisting organizations to better manage their resources. But machine learning also has the potential to do great harm. For example, in 2017, a study was published showing that a machine learning model could predict—with startling accuracy—a person's sexual orientation from nothing but an image of their face.[1] Wl
sinister intentions, the study raised con
misuse of machine learning. Imagine if
illegal to be gay (happily, Botswana lega
2019, so this number should now be 71)
persecute or even execute people?

[1]Yilun Wang and Michal Kosinski, "Deep
Accurate than Humans at Detecting Sex
Images," 2017, https://osf.io/zn79k.

Here's another example: in 2015, it was discovered that Google's algorithm for image recognition would classify images of people of color as images of gorillas.[2] The ethical consideration here is that the data the algorithm was trained on was biased toward images of white people and did a poor job of making accurate (and non-racist) predictions on images of non-white people. To avoid this kind of bias, it is imperative that our datasets are adequately representative of the population our model will be let loose on. Whether this is done using sensible sampling strategies or by testing for and correcting biases after training, it is our responsibility to ensure that our models aren't biased against particular groups of subjects.

[2]Jessica Guynn, "Google Photos Labeled Black People 'Gorillas,'" *USA Today*, 2015, http://mng.bz/j5Na.

An additional ethical concern with regard to machine learning research is one of security and credibility. While it may seem like something taken directly from a science fiction film, machine learning research has now reached a point where models can create videos of a person speaking, from just an image of their face.

Researchers have used this so-called ***deep fake*** technology to produce videos of Barack Obama speaking whatever audio they provide.[3] Imagine misusing this technology to fabricate evidence of a defendant in a criminal trial making a statement they never made. Similar technology has also been used to replace one person's face in a video with another person's face. Sadly and notoriously, this has been misused to swap celebrities' faces into pornographic videos. Imagine the poten person's career and dignity.

[3] Supasorn Suwajanakorn, Steven M. Sei Shlizerman, "Synthesizing Obama: Lea *ACM Transactions on Graphics* 36 (4), art /WOQg.

The previous point brings me to the issu consent. In order to train useful machine learning models that perform well, we need data. But it's important to consider whether the data you are using was collected ethically. Does it contain personal, sensitive, or financial information? Does the data belong to anyone? If so, have they given informed consent as to how it will be used? A spotlight was shined on these issues in 2018 when the consultancy firm Cambridge Analytica mined the social media data of millions of people without their consent. The subsequent media outcry and liquidation of Cambridge Analytica should serve as a stark reminder as to the importance of ethical data-collection procedures.

Two more ethical considerations are these:

- When a model suggests a particular course of action, should we follow its prediction blindly, or take it under advisement?
- Who is culpable when something goes wrong?

Imagine that we have a machine learning model that tells us whether to operate on a patient based on their diagnostic data. Would you be happy to follow the advice of the model if it had been shown to be correct in all previous cases? What about a model that

predicts whether a defendant is guilty or innocent? You could argue that this second example is ridiculous, but it highlights my point: should humans be involved in the decision-making processes informed by machine learning? If so, ***how*** should humans be involved in these processes? The answers to these questions depend on the decision being made, how it affects the people involved, and whether human emotions should be considered in the decision-making process.

The issue of culpability poses this quest
by a machine learning algorithm leads t
We live in societies in which people are l
actions. When something bad happens,
expect that someone will be found culpa
self-driving capability collided with anc
was culpable? The manufacturer? The p
pedestrian? Does it matter if the pedestı
quandaries like these need to be considered and carefully worked out before such machine learning technologies are released into the world.

[4]"Death of Elaine Herzberg," Wikipedia, http://mng.bz/8zqK.

When you train a machine learning model, I request that you ask yourself these five questions:

- Are my intentions ethical?
- Even if my intentions are ethical, could someone else do harm with my model?
- Is my model biased in a way that can cause harm or discriminate?
- Has the data been collected ethically?
- Once deployed, how will humans fit into the decisions made by the model?

If the answer to any of them makes you feel uneasy, please carefully consider if what you're doing is ethical. Just because we ***can*** do something, doesn't mean we ***should***. If you would like to explore a deeper discussion of how to perform ethical machine

learning, I suggest *Towards a Code of Ethics for Artificial Intelligence* by Paula Boddington (Springer, 2017).

1.4. Why use R for machine lear

There is something of a rivalry between used data science languages: R and Pyth machine learning will choose one or the their decision will often be guided by the learning resources they have access to, which one is more commonly used in their field of work, and which one their colleagues use. There are no machine learning tasks that are only possible to apply in one language or the other, although some of the more cutting-edge deep learning approaches are easier to apply in Python (they tend to be written in Python first and implemented in R later). Python, while very good for data science, is a more general-purpose programming language, whereas R is geared specifically for mathematical and statistical applications. This means users of R can focus purely on data but may feel restricted if they ever need to build applications based on their models.

There really isn't an overall winner when pitching these two against each other for data science (although of course everyone has their favorite). So why have I chosen to write a book about machine learning in R? Because there are modern tools in R designed specifically to make data science tasks simple and human-readable, such as those from the *tidyverse* (we'll cover these tools in depth in chapter 2).

Traditionally, machine learning algorithms in R were scattered

across multiple packages written by different authors. This meant you would need to learn to use new functions with different arguments and implementations each time you wanted to apply a new algorithm. Proponents of Python could use this as an example of why it was better suited for machine learning, because Python has the well-known scikit-learn package that has a plethora of built-in machine learning algorithms. But R has now followed suit with the caret and mlr packages. While r in purpose and functionality, I believe n intuitive; so, we'll be using mlr in the bo

The mlr package (which stands for *mach* an interface for a large number of mach allows you to perform extremely compli tasks with very little coding. Where poss package throughout this book so that w be proficient at using one of the most m packages available.

1.5. Which datasets will we use?

To make your learning process as fun and interesting as possible, we will use real datasets in our machine learning pipelines. R comes with a considerable number of built-in datasets, which are supplemented by datasets that come with packages we'll be loading into our R sessions. I decided to use datasets that come with R or its packages, to make it easier for you to work through the book while offline. We'll use these datasets to help us build our machine learning models and compare how different models perform on different types of data.

TIP

With so many datasets to choose from, after completing each chapter, I suggest you apply what you've learned to a different dataset.

1.6. What will you learn in this b

This book gives you a hands-on introduction to machine learning with R. To benefit from the book, you should be comfortable with basic R coding, such as loading packages and working with objects and data structures. You will learn the following:

- How to organize, tidy, and plot your data using the tidyverse
- Critical concepts such as overfitting, underfitting, and bias-variance trade-off
- How to apply several machine learning algorithms from each of the four classes (classification, regression, dimension reduction, and clustering)
- How to validate model performance and prevent overfitting
- How to compare multiple models to decide on the best one for your purpose

Throughout the book, we'll use interesting examples to learn concepts and apply our knowledge. When possible, we will also apply multiple algorithms to the same dataset so you get a feel for how different algorithms perform under certain situations.

Summary

- Artificial intelligence is the appearance of intelligent knowledge by a computer process.
- Machine learning is a subfield of artificial intelligence, where the computer learns relationships in data to make predictions about future, unseen data or to identifv meaningful patterns that help us u better.
- A machine learning algorithm is th patterns and rules in the data are le collection of those patterns and rul data, applies the rules to it, and out
- Deep learning is a subfield of mach itself, a subfield of AI.
- Machine learning algorithms are ca supervised and unsupervised, depe learn from ground-truth-labeled data (supervised learning) or unlabeled data (unsupervised learning).
- Supervised learning algorithms are categorized/divided as classification (if they predict a categorical variable) or regression (if they predict a continuous variable).
- Unsupervised learning algorithms are categorized/divided as dimension reduction (if they find a lower-dimension representation of the data) or clustering (if they identify clusters of cases in the data).
- Along with Python, R is a popular data science language and contains many tools and built-in datasets that simplify the process of data science and machine learning.

Up next...

Chapter 2. Tidying, manipulating, and plotting data with the tidyverse

- Understanding the tidyverse
- What is meant by tidy data
- Installing and loading the tidyverse
- Using the tibble, dplyr, ggplot2, tidyr, and purrr packages