PYTHON MACHINE LEARNING

FORECAST, PREDICT, AND DETECT ANOMALIES WITH STATE-OF-THE-ART MACHINE LEARNING

DANIEL BEGUM

Python Machine Learning

DANIEL BEGUM

Table of Contents

Introduction

The following chapters will discuss all the different parts of machine learning that you can do with the help of the Python coding language. When these two programming ideas come together, there are so many things that you are able to do, and the applications are endless! This guidebook is going to take some time to look at the basics of Python machine learning and how you can get started—even as a beginner.

To start this guidebook, we are going to take a look at the basics of machine learning and then move on to some of the basic parts of working with Python when you combine it together with machine learning. If you have never worked in coding—or at least, have never done any work with the Python coding language—we also have a chapter that will explore a few of the basics that you need to know to start writing your own Python codes and even how to set up the right environment, with the right libraries, so that you can get started with writing codes for machine learning.

From here, we are going to take a look at Scikit-Learn and this library and its importance—before moving on to some of the different algorithms that you are able to work within machine learning, with the help of the Python coding language. Some of the algorithms that we will explore here will include the KNN, K-Means Clustering, Support Vector Machines, Neural Networks, Random Forests, Recurrent Neural Networks, and Linear Classifiers. We can then end the guidebook with a brief talk about some of the other neat things that you are able to do with Python machine learning and how much it can help you with your programming goals.

Many people are interested in learning more about machine learning, and they want to be able to use the Python coding language to help them see the best results.

When you are ready to see what machine learning is all about, as well how you are able to combine it together with your own codes and applications to get the best results, then make sure to read through this guidebook to help you get started!

There are plenty of books on this subject on the market—thanks again for choosing this one! Every effort was made to ensure it is full of as much useful information as possible. Please enjoy!

Chapter 1: Understanding the Basics of Machine Learning

The first thing that we need to explore in this guidebook is some of the basics that come with machine learning. It isn't going to do us much good to learn the Python code and the different parts that come with that, as well as how to use it with machine learning, if we really have no idea what machine learning is all about. Machine learning is sometimes known as a form of artificial intelligence that is there to provide a computer or another system with the ability to learn from its own past experiences, rather than it being programmed ahead of time with everything that it should know how to do. This makes it more versatile and better at its job overall and opens up so many doors and opportunities that just aren't present with traditional coding.

This learning process can begin with some data or observations—kind of like instructions for the code, examples, and even direct experiences so that the machine is better able to find the right patterns from that data. Then, the machine

can take those predictions in order to know what it can do in the future. The main goal that goes with machine learning here is that it will allow a computer to learn on its own, automatically, without someone coming in and adjusting things or having to code out for every situation.

Any time that machine learning shows up in your process, you may find that it will really make analyzing information, especially larger amounts of data, really easy. Machine learning is even able to give you some of the most profitable results—but before you can get to that point, you need to have a good understanding of what it takes to make this all happen.

While there are many uses and advantages to machine learning, it is a type of coding that will take a bit more work than other kinds. This is because you are working on training the models of machine learning how to behave and learn, even when you are not there to show it how to behave. This is a bit more difficult to code, but it is going to do wonders when it comes to increasing the number of things the system can do.

There are a lot of different things that you are able to use the processes of machine learning. If you are uncertain about how the input will be from the user or if you aren't sure how the end result will be, then machine learning can be useful. If you would like the computer to sort through the data and come up with some predictions for you, then machine learning can be helpful as well. Some good examples of what you can use machine learning for include:

1. Voice recognition
2. Facial recognition

3. Search engines – The machine learning program is going to start learning from the answers that the individual provides, or the queries, and will start to give better answers near the top as time goes on.
4. Recommendations after shopping
5. Going through large amounts of data about finances and customers and making accurate predictions about what the company should do to increase profits and happy customers along the way

These are a few of the examples that you may find useful in figuring out what machine learning is already able to do. It is likely that over time, the applications of this are just going to grow as more programmers learn how to use it for their needs. Many of the traditional types of programs that you have learned how to use are going to be much simpler than this, but this means that they are going to be a bit limited on what they are able to do. You have to tell the system exactly what it is allowed to do or not do. And this doesn't allow for a lot of different options from the user along the way.

This kind of coding is not always going to be the best option, especially with some of the examples that we are going to show you with machine learning. For example, if you are trying to put together a search engine, it is impossible to figure out all the different search queries that the user will put into it. Or, if you are working with some of the voice recognition software that is out there, it is impossible to figure out ahead of time each request or how each person is going to talk. These are things you would need to know with traditional coding, making it impossible to use. But machine learning is able to handle it.

The Importance of Bringing Python On Board

While we are going to take a closer look at the Python coding language and how it works in a bit, it is important to note here that we are using Python with machine

learning because it is one of the best languages for doing this. Python coding is simple, making it easy enough for a beginner to start out with but it still has plenty of power behind it to ensure that you are going to be able to get the work done. The Python language also comes with a big library, it can work well with other languages if you need to have two or more work together, and it is actually really easy to read, even as a beginner.

As we go through this guidebook, you will notice that a lot of the examples that we take a look at are going to be in Python. This is going to help you to learn a bit about the Python language, while also seeing how some of the machine learning algorithms work as well.

There are a lot of other coding languages that can work well with machine learning. But often they add a layer of complications to the process, and they are often saved for those who really know that particular language really well, or for some of the most complex options with machine learning.

The Python language is so much better at helping you to complete the tasks that you would like. It is simple to work with, it is easy to understand, and it does have all the power you need to work with the algorithms in this guidebook. In fact, we will even go through some of the algorithms and show you the different Python codes that you can use to get the most out of each one. There is nothing like the ease of use, power, and variety that you are able to find when you use the Python coding language, even when it comes to machine learning.

Classifying Algorithms in Machine Learning

Another thing that we need to note here is that there are actually a few types of algorithms that go with machine learning—three, to be exact. These include reinforced learning, unsupervised learning, and supervised learning.

Supervised Learning

First on the list is supervised learning. This one is going to involve the help of a person the most out of the three. A computer programmer is going to need to provide the computer with some input and tell it the desired output—meaning that they need to be able to furnish the right feedback to the system, based on how accurate the predictions of the system are during that training session. This is going to mean that the trainer will need to show the system a lot of examples to the system and show it what answers are right and what aren't, which allows the system a way to learn along the way.

After the training is done, the algorithm is then going to take the knowledge that it learned from the data, and then works to make the best predictions. The concept with this kind of learning is similar to learning in a classroom. The teacher will hand out a lesson to the students, usually providing some examples. And from this, the students are able to derive the new rules and the knowledge that they need to gain from these examples. The students are then able to take that knowledge and will apply it to different situations, even if the new data isn't exactly the same as the information they got during training.

Any time that we are looking towards this kind of learning, it is good to note that there are going to be some differences that show up between regression problems and classification problems. A regression problem to start out with is going to be when the target is a numeric value of some kind. But the classification means that it will have a tag or a class.

A regression task can sometimes help the programmer figure out the average of the cost of all the houses for sale in town for example. And then the classification

problem would help you to figure out what type of flower is inside a picture that is given, based on the petal length and other factors.

Supervised learning is what will occur at any time that you choose an algorithm that is able to learn the right response to the data at hand based on what the user gives to it. There are different methods that machine learning of this kind is able to do this. It is able to look at the examples and some other targeted responses that you are able to provide to the computer. The program is also able to look at examples and some other targeted responses that are given to the computer even in the form of values and strings so that it is able to learn the right way to behave.

Supervised learning is a pretty simple process to work with. You are basically just going to show the system a lot of different examples of what is right, and from that, it can look at new examples and determine if they match up and are right, or if they don't match up and would be considered wrong.

Unsupervised Learning

Another form of machine learning that you are able to work with is known as unsupervised machine learning. With these kinds of algorithms, you do not have to go through and provide the data to the computer like we had to do above. This is because you would rather have the machine learn what the output should be based on an input that is unknown. An approach that is called deep learning or the iterative approach is going to be used in order to review the data and help you come up with new conclusions.

What this ends up doing for you is making the unsupervised learning approach more suitable for a lot of different processing tasks, which are going to be more complicated to deal with. This allows the learning algorithms to learn from examples, without a response to it. The algorithm is going to strive to find the right

patterns that come from these kinds of examples, all on its own, rather than having the programmer tell it the answers.

Many of the recommended types of systems that you encounter, such as when you are purchasing something online, are going to work with the help of an unsupervised learning algorithm. In this kind of case, the algorithm is going to derive what to suggest to you to purchase based on what you went through and purchased before. The algorithm then has to estimate the customers you resemble the most based on your purchases and then provide you with some good recommendations from there.

As we mentioned a little bit before, there is more than one type of machine learning that you can work with. Supervised learning is the first one. It is designed for you to show examples to the computer and then you teach it how to respond based on the examples that you showed. There are a lot of programs where this kind of technique is going to work well, but the idea of showing thousands of examples to your computer can seem tedious. Plus, there are many programs where this is not going to work all that well.

And this is where you are going to notice that machine learning is going to be useful. Unsupervised learning allows the system to learn either from examples that it is given, or from mistakes, without a response that goes with either of them. This means that with the unsupervised algorithms, they are going to be in charge of figuring out and also analyzing the data patterns based on the input that it is given.

There are a few options that you can choose from when you want to do an algorithm of unsupervised machine learning. No matter which algorithm you decide to use, it is going to take the data that you provide, and can then restructure it so that all of the data ends up in some classes. This makes it easier for anyone, even if they

are not a programmer, to take a look at the information and form some conclusions from it. And unsupervised machine learning is going to be the best one to get this done.

Reinforcement Learning

And the third type of machine learning that we need to take a look at is the idea of reinforcement learning. This is a kind of learning that is going to happen when the algorithm is presented with some examples that don't come with labels, pretty similar to what we see with unsupervised learning. However, with this learning type, there is going to be negative and positive feedback based on the kind of solution that the algorithm suggests. The applications that will tend to use this are the ones that will need to make some decisions on its own, and then these decisions will either be right or wrong with consequences to match. Think of this kind of learning as trial and error.

Errors are actually fine with this method because they ensure that the model is better with learning and making predictions later on. The learning method is going to have some kind of a penalty—such as pain, cost, or a loss in time if they get the answer wrong. This helps it to learn what to avoid and will direct it to the path of making the right decision.

You will find that the learning processes that come with machine learning will be similar to what is seen with data mining and predictive modeling. In both of these scenarios, these patterns can be adjusted inside the program accordingly—a good example of how this may work the system for recommendations. If you decide to purchase an item online, if the site has this, then you will see an ad that is going to be related to that item.

There are some programmers who will think that unsupervised learning and reinforcement learning is the same thing. There are some similarities with it, but there are some differences. First, the input that is given into these kinds of algorithms will need to have some mechanisms for feedback. You are able to set them up to be positive or negative, and this is going to depend on the algorithm that you choose to work with.

Any time that you would like to work with this kind of machine learning, you will see that this is going to work like trial and error. Think about when you are working with a younger child. When they end up going through and doing an activity that you didn't approve, you will start out by telling them it is time to stop, or you will do another consequence that you thought through ahead of time. Then there will be times when the child will do actions that you approve of, and you will need to take some time to praise them and let them know they are doing things that are right. With these two steps, the child will start to learn acceptable behavior.

This is the same idea that reinforcement learning is going to work with as well. This whole process is going to work on the idea of trial and error, and it will require that your application uses an algorithm to help it make the right decisions. It is a good one to use if you would like an algorithm is able to make decisions without any mistakes and a good outcome. Remember in this that your program is going to need some time to learn. It is going to make some mistakes along the way, but as it does this trial and error more often, it is going to get better at doing the work.

Understanding What Deep Learning Is About

The next thing that we need to focus on is the idea of deep learning. This is going to be a subfield of machine learning involving algorithms that are inspired by the function and structure of the brain known as artificial neural networks. It is going

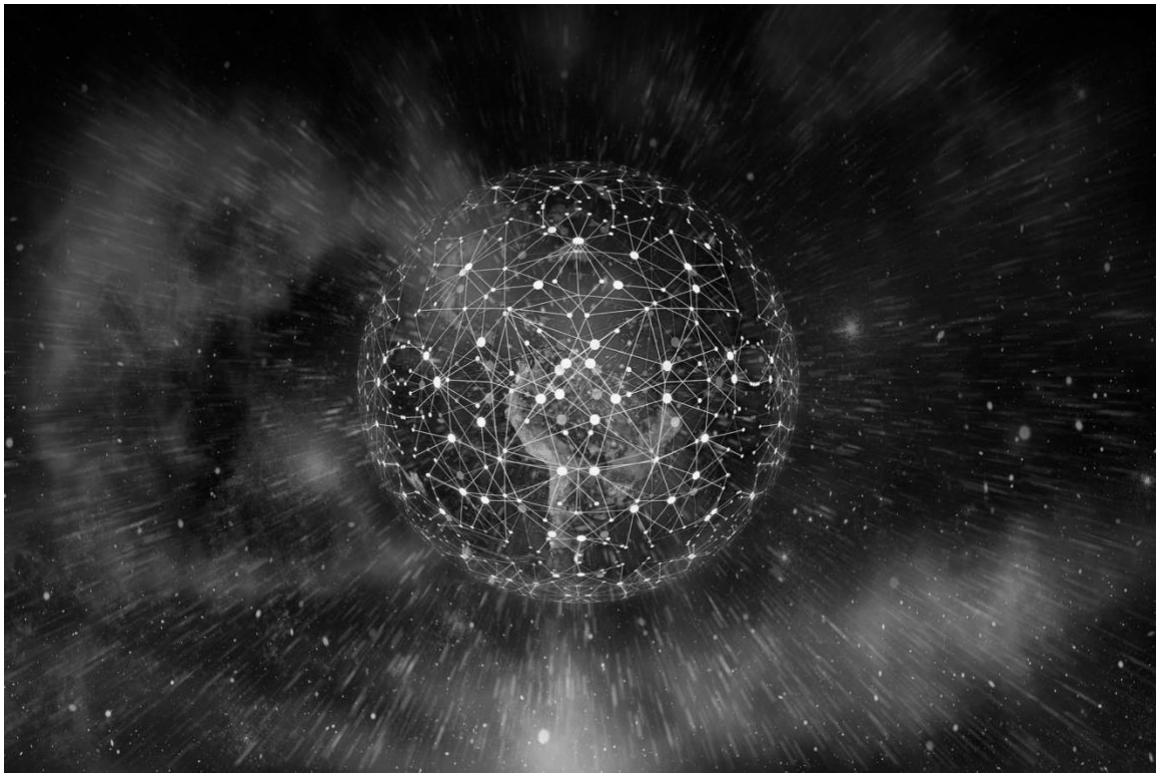
to work to teach your computer to act and behave the way that a natural to humans, that is, that the system will be able to learn by example.

It is through the help of this kind of deep learning that your computer is going to be able to learn to perform the various classification tasks directly from sound, image, and text. Deep learning models are able to achieve the state-of-the-art accuracy, which in some cases will be able to exceed the performance that you can get at the human level. Large sets of labeled data and neural network architectures are going to be used to train models with some deep learning.

As you can see, there are a lot of different parts that can come with machine learning. It is a great tool that you can use in order to get your computer or your system to do some of the things that you may struggle with. For example, machine learning can be used in order to help work with search engines, and maybe with providing you with some recommendations when you are shopping online. Even if you need to sort through a large amount of data, machine learning can come into play and help out.

The rest of this guidebook is going to spend some time looking through some of the basics that come with machine learning, and how to work with it on Python, to ensure that you are able to get the most out of this process. Let's take a closer look at some of the processes that you need to know how to work with to see amazing results with machine learning with Python.

Chapter 2: Some of the Basic Parts of Working with Python and Machine Learning



Now that we have a better understanding of what machine learning is all about, it is time to move on to some more of the things that you will be able to do with machine learning and how two of the libraries from Python are going to be able to come in and help you make the process easier. You will notice that both of these work really well with Python, and they are there to ensure that the codes you write in Python will be as effective and strong as possible. And since they can work with machine learning, you are going to enjoy the benefits as well.

The two main processes and libraries that we need to look in this chapter will include TensorFlow and Scikit-Learn. Let's explore more about what these are like and why they can be so important to your program on a machine learning project.

What Is Scikit-Learn?

We also need to take some time to learn about Scikit-learn. This is going to provide your users with a number of supervised and unsupervised learning algorithms through a consistent Python interface. We are going to take some time to learn more about Python later in this guidebook, but it is a fantastic tool that you are able to use to enhance your machine learning, and since it is for beginners, even those who have never worked with coding in the past will be able to use it.

The Scikit-learn was developed in 2007by David Cournapeau as a Google Summer of code project. This process is going to be suitable to use whether you need it commercially or academically.

Scikit-learn has been in use as a machine learning library inside of Python. It is going to come with numerous types of classification, regression, and clustering to help you get more results. Some of the algorithms that you will get to use with this system is going to include DBSCAN, k-means, random forests, support vector machines, and gradient boosting to name a few. And Scikit-learn was designed so that it would work well with some of the other popular libraries that are found on the Python code, including with SciPy and Numpy libraries.

This particular library, in particular, has been all done by Python, but some of the different formulas and algorithms that you are going to rely on to make this one work will be written with the help of Cython. If you want to make sure that the performance that you get is the best with this, you will find that the Scikit-Learn library is the one that you need to focus on. It is especially good at building up some of the models that you need with machine learning. And since it is a library that is open sourced and easy to get started with, you will easily be able to open it up and start using it as soon as needed.

Understanding More About Tensor Flow

The second topic that we are going to explore in this chapter is a second library known as TensorFlow. This is a framework that you can get through the Google platform, and it is used when you would like to create some models in deep learning. This TensorFlow library is often going to rely on some data flow graphs that work on numerical computations. And it is able to stop in and make the process of machine learning easier than before.

You will find that working with TensorFlow makes the process of getting data, of training the models that you would like to use with machine learning, of making predictions, and even modifying some of the results that you see in the future so much easier. Since all of these are going to be really important when it comes to machine learning, you can see why we want to spend some time learning TensorFlow.

TensorFlow is a library that the Brain team from Google developed to use on machine learning, and it is especially effective when you want to do some machine learning on a larger scale. TensorFlow is going to bring together algorithms that work with deep learning and with machine learning, and it helps to make them more useful trough a common metaphor.

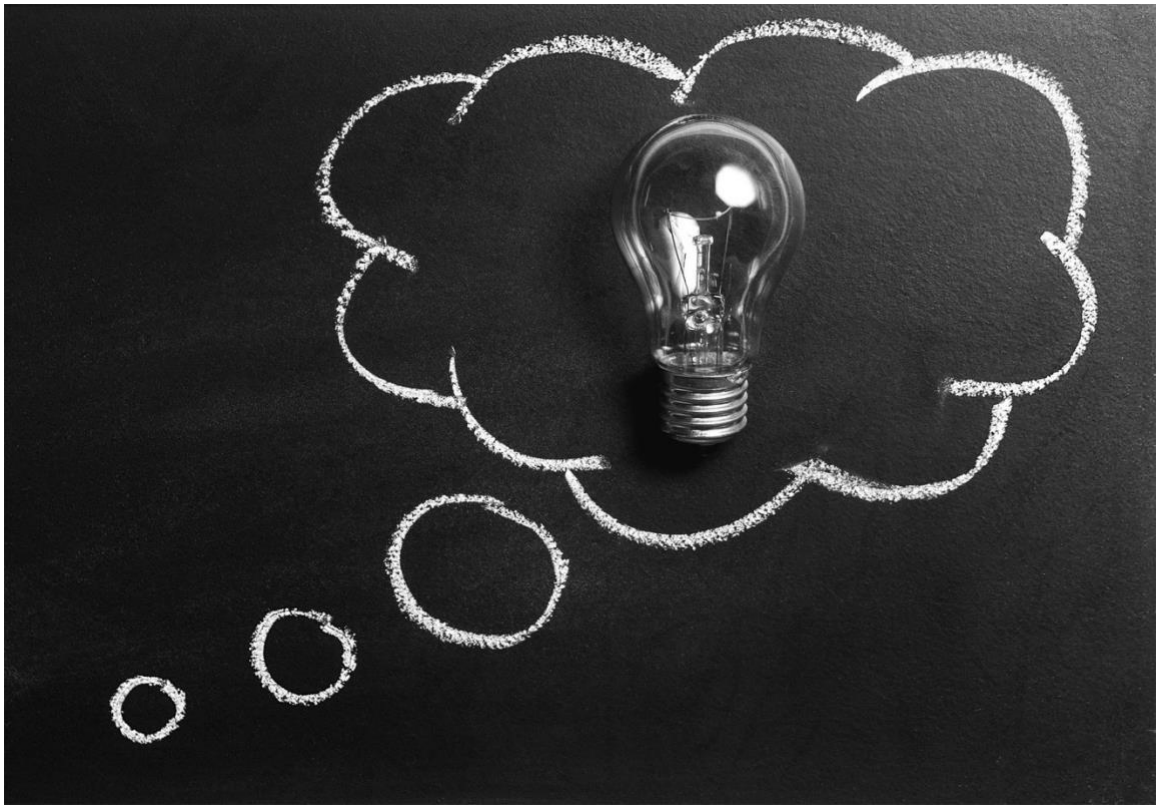
Just like what we saw when we were working on the other library, TensorFlow is going to work together well with Python, and it will ensure that you are able to have a front-end API that can be used when you would like to build a new application. And when you execute these applications, you will see them done in what is known as high-performance C++.

TensorFlow can be used to help out with running deep neural networks, for training, building, handwritten digit classifications, recurrent neural networks,

word embedding, and even natural language processing to name a few of the neat things you will be able to do.

Both of these two libraries work well with Python, and they are going to do an important job when it comes to ensuring that you are on the right track with your machine learning. Both of these are going to take on a lot of different tasks, and you will need to pick out the right one based on what you would like to get done on your project.

Chapter 3: What Is Python, and How Do I Use It?



We have spent some time talking about the Python coding language and some of the neat things that you are able to do with this. However, in order to complete this book, we also need to get a good understanding of the Python language and what it is all about. There are a lot of different parts that show up in the Python coding language, but knowing some of the basics, as well as learning some of the power that comes with Python, makes all of the difference on the success that you are able to get when you combine this coding language together with machine learning.

We have talked briefly about the Python coding language already, as well as some of the reasons why it may be so beneficial when you want to work with machine learning. Even though it is considered one of the easier coding languages to learn

how to work with, it has a lot of power and strength that comes in behind it, which makes it the best option to use whether you are a beginner or someone who has been coding for quite some time. And since the Python language does have some of the power of the bigger and more powerful coding languages, you will be able to do a lot of cool things with machine learning.

There are going to be a few different parts that come into play when you start to learn how to work with the Python code even with machine learning. You are able to work with the comments, functions, statements, and more. Let's take a look at some of the basic parts that come with coding in Python so that we are able to do some of these more complicated things together as we progress through machine learning.

The Comments

The first aspect of the Python coding language that we need to explore is that of comments. There is going to be some point when you are writing out a code where you would like to take a break and explain to others and to yourself what took place in the code. This is going to ensure that anyone who sees the code knows what is going on at one point to another. Working with a comment is the way that you would showcase this in your project, and can make it easier for others to know the name of that part, or why you are doing one thing compared to another.

When you would like to add in some comments to the code, you are going to have a special character that goes in front of your chosen words. This special code is going to be there in order to help you tell the computer program that it should skip reading those words and should move on to the next part of the code instead. The special character that you are going to be able to use for this one is the # sign in front of the comments that you are writing. When the compiler sees this, it is going to know that you don't want that part of the code to execute at all. It will wait until

the next line before it gets started with reading the code again. An example of a comment that you may see in your code would include:

```
#this is a new comment. Please do not execute in the code.
```

After you have written out the comment that you want here, and you are done with it, you are then able to hit the return button or enter so that you can write more code that the compiler is able to execute. You are able to have the freedom to make a comment as long or as short as you would like, based on what you would need in the code. And you are able to write in as many of these comments as you would like. It is usually seen as a better option if you keep the comments down to just what you need. Otherwise, it makes the code start to look a little bit messy overall. But you can technically add in as many of these comments to your code as you would like.

The Statements

The next part of the code that we need to focus on is the statements. Any time that you are starting out with your new code, whether you are working with Python or with some other coding language along the way, you must add these statements inside of the code. This allows the compiler to know what you would like to happen inside. A statement is going just to be a unit of code that you would like to send to your interpreter. From there, the interpreter is going to look over the statement and execute it based on the command that you added in.

Any time that you decide to write out the code, you are able to choose how many statements are needed to get the code to work for you. Sometimes you just need to work with one statement in a block of code, and other times you will want to have more than one. As long as you are able to remember that the statements should be

kept in the brackets of your code, it is fine to make the statement as long as you would like, and to include as many statements as you would like.

When you are ready to write your code and add in at least one statement to the code you are working with, you would then need to send it over so that the interpreter is able to handle it all. As long as the interpreter is able to understand the statements that you are trying to write out, it is going to execute your command. The results of that statement are going to show up on the screen. If you notice that you write out your code and something doesn't seem to show up in it the right way, then you need to go back through the code and check whether they are written the right way or not.

Now, this all may sound like a lot of information, but there is a way to minimize the confusion and ensure that it is able to make more sense to you. Let's take a look at some examples of how this is all going to work for you.

```
x = 56
Name = John Doe
z = 10

print(x)
print(Name)
print(z)
```

When you send this over to the interpreter, the results that should show up on the screen are:

```
56
```

```
John Doe

10
```

It is as simple as that. Open up Python and give it a try to see how easy it is just to get a few things to show up in your interpreter.

The Variables

The next thing that we need to take a look at inside our Python codes are the variables. These variables are important to learn about because they are the part that will store your code in the right places so you can pull them up later on. This means that if you do this process in the right way, the variables are going to be found inside the right spot of the memory in the computer. The data that you are doing in the code will help determine which spots of the memory these points will be stored on, but this makes it easier for you to find the information when it is time to run the code.

The first thing that we need to focus on here is to make sure that the variable has a value that is assigned to it. If there is a variable without a value, then the variable won't have anything to save and store. If the variable is given a good value from the start, then it will behave the way that you are expecting when you execute the program.

When it comes to your Python code, there are going to be three types of variables that you are able to choose from. They are all important and will have their place to work. But you have to choose the right one based on the value that you would like to attach to that variable. The main variables that are available for you with Python are going to include:

1. Float: This is going to be an integer variable that includes numbers like 3.14
2. String: This is going to be one of those statements that you would write out. You can add in any kind of phrase that you would like to this one.
3. Whole number: This is going to be any of the other numbers that you would want to use, ones that are not going to contain a decimal.

When you are trying to work with the variables in your program, you won't have to go through and make a declaration in order to make sure the memory space is available. This is something that Python is able to do for you automatically, as soon as a variable is assigned a value. If you would like to make sure that this is happening in your code, and to avoid some of the surprises along the way, you just need to double check that the equal sign, the sign of giving a variable a value, is in the right place. A good example of how this is going to look when you write out a code includes the following

```
x = 12              #this is an example of an integer assignment
pi = 3.14           #this is an example of a floating point assignment
customer name = John Doe #this is an example of a string assignment
```

In some instances, you may need to have one variable with two or more values attached to it. There are certain times when you won't be able to avoid this and need to make it happen. The good news is that you are able to work with the same kind of procedure that we talked about above to make this happen, you just need to make sure that there is an equal sign to each part so that the compiler knows how to assign everything. So, when you want to do this, you would want to write out something like a = b = c = 1.

The Keywords

Any time that you are using the Python language, just like with what we find in other coding languages as well, you are going to come across some words that are reserved as commands in the code, and these are going to be known as keywords. You need to be careful about the way that you use these because they are there to tell the program some commands, and how you would like it to behave. You don't want to bring these and use them in any other place outside of a particular command.

If you do use these keywords in the wrong manner, it is going to cause some confusion inside the code. The interpreter isn't going to know what to do, and the computer may get stalled in what it needs to do. As we go through this guidebook some more and develop a few Python codes for machine learning, we will start to recognize some of the most common keywords that you need to watch out for.

Naming Your Identifiers

The next thing that we need to focus on is how to name all of the identifiers. You need to be able to do this in a way that makes sense and will not confuse the computer along the way. Any time that you are writing out a particular code in Python, you are going to have at least a few identifiers that show up. Some of the most common of these identifiers are going to including classes, entities, variables, and functions.

At one point or another, you will need to name out the identifiers so that they are more likely to work in the way that they should and to make it easier for the compiler to pull them up when it is time to execute the code. No matter which of the four identifiers that you use, the rules for naming are going to be the same, which can make it easier for you when you get started. Some of the rules that you

should remember when it comes to naming your identifiers include:

Any time that you are using letters, it is fine to use either upper case or lower case, and a combination of both is fine as well. You can also add in numbers, symbols, and an underscore to the name. Any combination of these are acceptable, just make sure that there are no spaces between the characters of the name.

Never start out the name of one of your identifiers with a number. This means that writing out the name of 4babies would result in an error on your computer. However, it is acceptable to name an identifier `"fourbabies"` if you would like.

The identifier should never be one of the Python keywords, and the keyword should not appear in the name at all.

Now, if you are trying to go through and write out a few of your own codes along the way, and you don't follow all of the rules that are above, you will confuse the compiler, and it will send you an error message to let you know something is wrong. The error will then be on your computer, and the program will close out because it doesn't know how to proceed. This is obviously something that you don't want to happen, so be careful about the way that you are naming these identifiers.

Another thing to remember with these identifiers and naming them is that you want to make sure that you are picking out names that are ready for you to remember. And picking out names that are easy to understand and read will make it easier if another programmer comes in and is trying to look through the code as well.

What Else Do I Need to Know About the Python Coding Language?

The Python coding language is often considered to be one of the best languages for programming by the experts, and this is even truer if you are someone who has never worked with coding in the past. This language is simple, it has plenty of power, and there are a lot of the tools and the resources that are needed to help you to work on your project, even if that project has to do with machine learning. While there are other options that you can use with machine learning, those are often a bit more difficult to work with, and Python is often the better choice.

One thing that you will like when you first start working with the Python language is that the whole thing is based on the English language. What this means is that you will recognize the words and what they mean, rather than trying to guess. And even though it is a language that is great for beginners who have never worked with coding before, it is going to be plenty strong enough to work with machine learning and any other topic that you would like.

As someone who is ready to learn and get started with the Python language, you will notice that it has a nice big library available that you can utilize when you are coding. And this library is going to have the tools and the resources that you need, making it easier for beginners and experts alike to write out any code that they need.

Chapter 4: How to Set Up the Perfect Environment in Python

The next thing that we need to work on here is to set up your environment the way that you need to get to start with machine learning and see the results that you want. Now that we know a bit more about how all of this works with machine learning and know a bit about the Python coding language, it is time to set up the environment so that you can get started. This is a very important step to take before you try to start on any machine learning and deep learning techniques. And to help you get all of this set up, you need to make sure that your two libraries—Scikit-Learn and TensorFlow—are all set up and ready to go to get started.

Getting Started with Scikit-Learn

Both the Scikit-Learn and the TensorFlow library are going to need to be set up in order to start with Python machine learning. But the first one that we are going to work with setting up is the Scikit-Learn. This is a library that you are able to work with if you have Python 2.7 or higher on your system. If those are already installed, then you should already have things set up here. It is usually best if you are working with machine learning that you have one of these newer versions so installing that can help.

Before you start with installing this library, double check and see if the SciPy and Numpy libraries are already installed—if these are not present, then you need to install them first, and then you can go through and install the Scikit-Learn.

The installation of these libraries may seem like it is really time-consuming, it is important, and you will be able to do this with the help of pip. This pip is going to be a tool that will come along with Python so you will be able to use the pip as soon as the system is all installed. From here, you will be able to work with the command that is below to help you get the Scikit-Learn library ready to go:

From here the installation will be able to run and then it will complete once all of that is done. It is also possible for you to go through and use the option of `conda` to help install this library. The command that you will want to use to make sure that this happens is:

```
conda install scikit-learn
```

Once you notice that the installation of scikit-learn is complete, it is time to do some importation to get it over to the Python program. This step is necessary in

order to use the algorithms that come with it. The good news is that the command to make this happen is going to be done. You simply need to go to your command line and type in import `sklearn`.

If your command is able to go through without leaving behind an error message, then you know that the installation that you just did was successful. After you are done with all of these steps, your scikit-learn library is on the computer, it is compatible with the Python program, and it is going to be ready to use.

Installing TensorFlow

Once you have the Scikit-Learn library put in place on your environment, it is time to start installing the TensorFlow library. When you download this one, you will be able to get a few different APIs for other programming languages outside of just Python, including Java and C++, so if you need to use these, you will be able to get a hold of them pretty easily. You are able to download this on a few different devices if you would like, but for this guidebook, we are going to take a look at how you can install this library on a Windows computer. You are able to use the pip that we had from before, or Anaconda, to get this downloaded on your Windows computer.

The native pip is often the best way to get TensorFlow installed on your system, without having to worry about the virtual environment that comes with it. But one of the things that you are going to need to keep track of here is that when you install this library with a pip, there are times when this will interfere with some of the other installations of Python that are on the computer, and you need to be aware of this ahead of time.

The good thing to know here is that the only thing that you need to have up and running to make this work is a single command. And once you know this command, TensorFlow is able to install on your system and get it up and running. Once you are able to get this library installed using the pip, the user is going to see that there are some options, including the ability to choose which directory you would like to store this on.

Now, you can also choose to install the TensorFlow library on your computer with the help of Anaconda. To do this, you need first to go through and create your own virtual environment. You may also want to work with a pip with this one as well. Before you start on this one, make sure that you have a Windows system and Python 3.5 or above. Pip 3 program needs to be in place as well to help with this kind of installation. The commands that you are going to need in order to get started with this one include:

```
pip3 install – upgrade tensorflow
```

If you would like to make sure that you are installing the GPU version of the Tensorflow program, you would need to go through and use the following command to make it happen:

```
pip3 install – upgrade tensorflow-gpu
```

This is going to ensure that you are able to install TensorFlow on the Windows system that you are using. But another option that you can use to install this library so that you can use it with Python and all of your other machine learning algorithms will include being able to install it with the help of the Anaconda package.

Pip is a program that is automatically going to get installed when you get Python on your system. But the Anaconda program isn't. This means that if you would like to make sure that TensorFlow is able to get installed with the use of the Anaconda program, you first need to take the time to install this program. To do this, visit the website for Anaconda, download it from the website, and then find the instructions for installation from that same site.

Once you have gone through and installed the Anaconda program, you should notice that it comes with a package that is called conda. This is a good package to take some time to look at and explore a bit because it will work when you want to manage any installation packages or manage virtual environments. To get some access to this package, you just need to start up the Anaconda program.

After you get to this point, you will be able to head over to the Windows main screen and then click on the Start menu. You can choose All Programs and expand it out until you can see the folder for Anaconda. You should click on this prompt in order to launch the folder. If you need to see what details are in this package, you can just run a command in the command line for "conda info." This makes it easier to see what the details are for that package and the manager as well.

There are a lot of cool things that come with this Anaconda program, but one of the options that you will want to learn more about to help with machine learning. Anaconda is able to help you to create an environment of Python for your own using this kind of package. The virtual environment is going to be its own isolated copy of Python, and it will have the right capabilities of maintaining all of the files that it needs, including the directories and paths. This can be helpful because it allows you to do all of these things while still working with the version of Python that you want and any of the other libraries that you want.

These virtual environments may seem like they are complicated, but it is actually helpful because they are going to provide you with a way to isolate your projects, and can avoid some of the major problems that can arise along the way. Note that this is going to be a separate environment compared to the normal Python environment that was downloaded before. This is important because you won't be able to have any effect on the regular Python environment, whether it is bad or good.

At this point, we want to do some work to help us to create a virtual environment for that TensorFlow package. This can be done when we use the command of "conda create." Since we want to create a brand new environment that we have called `tensorenviron`, we would need to use the formula below to help:

```
conda create -n tensorenviron
```

At this point, the program is going to ask you whether or not you would like to allow the process of creating the environment to continue on, or if you would like to cancel the work that you are doing. You will want to type in the "y" and then hit the enter key to move on. This will allow the installation to continue on successfully to see the results that you want.

Once you have been able to go through this whole thing and create an environment, you will need to take a few minutes and let this environment to activate it. If you do not do the activation the right way, you won't be able to use this new environment that you just did—you won't be able to get the new environment to work, either. You will be able to do this with the help of the command to activate that we talked about before. From that point, you will then be able to list out the name that you would like for the environment. A good example of what we are

talking about here and how it will all work together includes:

```
Activate tensorenviron
```

Now that you have been able to activate the TensorFlow environment, it is time to go ahead and make sure that the package for TensorFlow is going to be installed, too. You are able to do this by using the command below:

```
Conda install tensorflow
```

From here, the computer is going to present you with a list of all of the different packages that you can install together along with the package for TensorFlow if you would like. You will be prompted to decide if you would like to install these packages or not. You can then type in the "y" and hit the enter key on the keyboard.

Once you agree to do this, the installation of this package is going to get started right away. However, notice that this particular process for installation is going to take a bit of time, so you need to wait it out and remain patient. However, the speed that your connection for internet goes is going to determine the amount of time that the installation process is going to take. The progress and how far the installation has gone and has yet to go is going to be shown on a prompt window.

After a certain amount of time has passed, the installation process is going to be complete, and you can then take some time to determine if the installation process was successful or not. This is pretty easy to do because you just need to run the import statement with Python. The statement is going to be done from the regular terminal of Python. If you are doing this with the Anaconda prompt, you can just type in "python" and hit the enter key. This is going to make sure that you end up in the terminal for Python and from there, you can run the import statement below:

```
Import tensorflow as tf
```

If you find that the package wasn't installed in the proper manner, you are going to end up with an error message on the screen after you do this code. If you don't see an error message, then you know that the installation of the package was successful.

Chapter 5: What Is Scikit-Learn, and Why Should I Learn About It?

There are a lot of things that you are going to enjoy when it comes to the Scikit-Learn environment and library. This is one of the best options that you are able to work with through Python and will make your machine learning projects so much more successful than before. If you are a programmer who is learning how to work with the process of machine learning or if you want to be able to do more with your Python codes, then you need to make sure you have a good understanding of this library and how it is going to work.

The Scikit-Learn library was developed in 2007. Later, the company has really started to grown and has made a lot of changes over time. Currently, it gets to enjoy more than 30 active contributors, and there are even some paid sponsorships from the INRIA, Google, and the Python Software Foundation to ensure that this library is going to be developed. And it is all done in a way that ensures that the user is not

going to have to pay to use it!

But this stars to bring up some questions about this library and what it is all about. This library is going to ensure that the computer programmer has a lot of algorithms for both the unsupervised learning and the supervised learning that they want to do. And these algorithms are adjusted so that they can stay consistent with the Python environment. This means that you will be able to use these algorithms and work on some machine learning projects, all on Python.

This particular library is going to be licensed under what is known as a permissive simplified BSD license, and many of the Linux distributions are able to use it as well. It is also going to be built with the use of the SciPy library that will help make things even easier. The stack that is found inside of this, and which you will find as really helpful when you are working with machine learning includes

1. NumPy: This is a good one to use because it allows you to work on the n-dimensional array package
2. SciPy: This one is going to be a fundamental kind of library that you would use if you wish to do computations in the scientific field
3. Matplotlib: This is a good library to use because it is going to help you do some plotting, whether that plotting is in 2D or 3D.
4. iPython: This is a good library to use because it is going to allow you a console that is more enhanced and interactive than others.
5. Sympy: This is a library that works well if you want to do some things in symbolic mathematics.
6. Pandas: This is the number one part that you need to use because it is going to include all of the analysis and the data structure that is needed to make machine learning successful.

The different extensions and modules that you are able to use with SciPy are known collectively as SciKits. This is why the module that helps to provide us with the learning algorithms that we need are going to be known together as the Scikit-Learn library.

The vision that is going to come in with this kind of library is going to include a lot more support and more robustness than you are able to find with some of the other topics that you explore. This is a good thing because both of these are going to require some higher levels to make sure that the production system will work the way that we expect and want. When we are going through this process, there has to be a deeper focus on the concerns, including ease of use, the collaboration, documentation, code quality, and performance, or it isn't going to work the way that we would like.

Knowing Some of the Features of This Library

At this point, we have talked a bit about this kind of library, but we haven't gone into any of the details of the features, or the reasons that you would choose to work with this kind of system over one of the others. When you decide to work with this particular library, you are probably going to be curious as to what it is all about, and even why some people want to work with this while learning and working with machine learning.

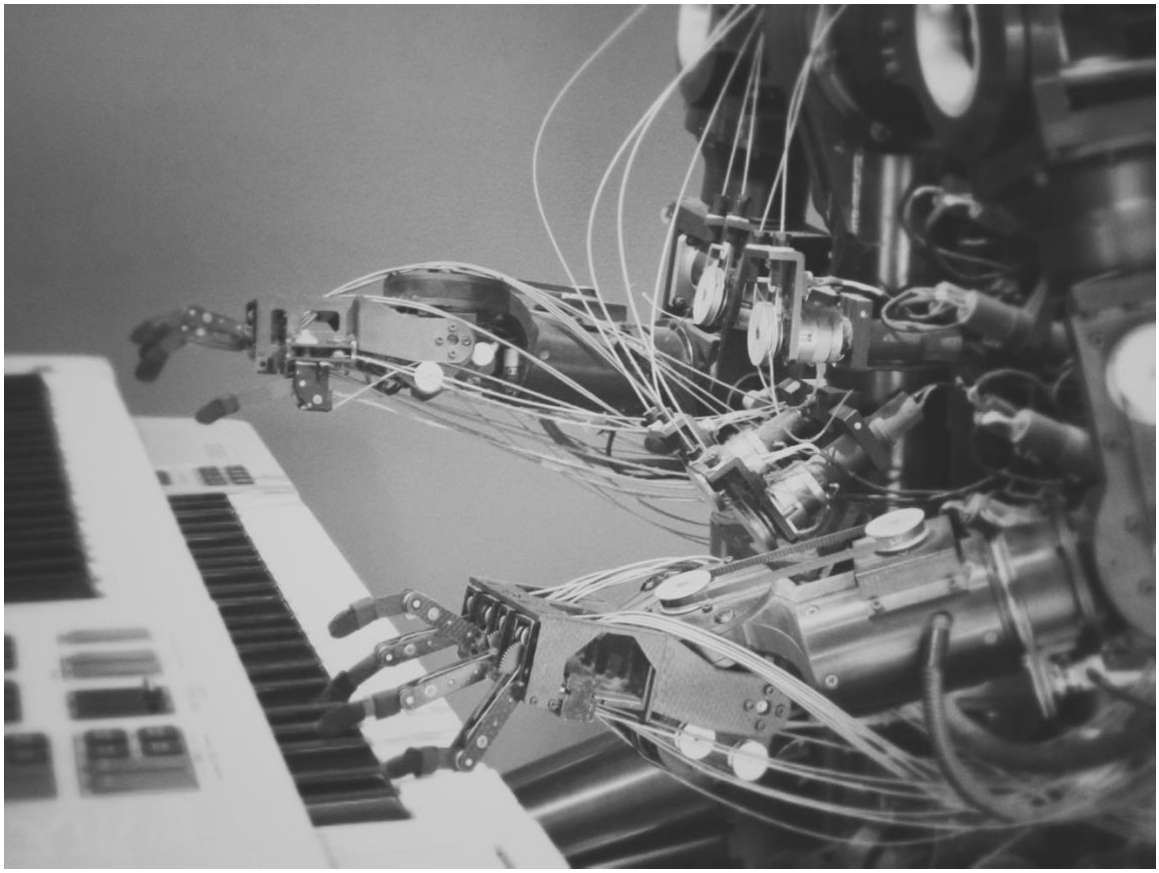
The Scikit-Learn library is going to be the most focused on modeling data. It isn't going to take that much time to look at how to summarize the data, to load the data, or to manipulate the data. If you want to take the time to work through these three topics, then you would want to spend some time in the libraries of NumPy and Pandas. However, some of the features that you are able to get through this library, and some of the different models that are available here include:

1. Supervised models: This Scikit-Learn library is going to provide you with many linear models (mostly generalized) that can work well in machine learning. This could include some of the algorithms like a discriminate analysis, decision trees, lazy methods, the Naïve Bayes, support vector machines and neural networks to name a few.
2. Manifold learning: These are important because they are often going to be used in order to help depict and then summarize some of the multi-dimensional data that may seem hard to get through for a beginner.
3. Parameter tuning: This is a type of tool that you may find useful when you want to learn more and get more out of your supervised models.
4. Feature selection: This is going to be a part of the library that is going to help you see, and then also identify, meaningful attributes from which to create a new model that is supervised.
5. Feature extraction: This one is helpful because it is going to help you learn how to define the attributes that are in text data and in images that are presented to the machine.
6. Ensemble methods: You will enjoy this feature because it is going to be helpful when you would like to combine the predictions that you have from several models of supervised learning, and then have these all come together to form a new prediction using all that information.
7. A reduction in dimensionality: This is a method found in this library that is helpful when you would like to find a way to reduce the number of attributes that are needed in data to help with feature selection, visualization, and summarization. A good example of how this works is going to be the principal component analysis.
8. Data sets: This is helpful because it is the place where you are able to test some of the sets of data that you have, the ones that are going to generate the right sets of data with the right properties so you can do a bit of investigating.

9. Cross-validation: There are times when you will want to figure out whether a model is giving you the accurate results that you would like. The cross-validation will help you get an estimation on how well that model is going to be able to perform on the data that you have, without it even seeing that data.
10. Clustering This is where you are able to group any of the data that doesn't have a label, such as with the K-means method that we will discuss a bit later.

These are just a few of the benefits that you are going to be able to get when it comes to working with this kind of library. It is a strong library to use, one that is going to provide you with the results and answers that you want to many supervised machine learning algorithms. Without this one in place, it is going to be hard to figure out how you would like to do these algorithms at all and to determine if the information you get is accurate when you work with machine learning.

Chapter 6: Working with the K-Nearest Neighbors Algorithm



There are a lot of different types of algorithms that you are able to work with when it comes to the idea of machine learning. Some of these are going to be easy to work with, and some of them are going to be a bit more complex—but they are able to handle information that is a bit more complex as well. The first algorithm that we are going to explore here is known as the K-Nearest Neighbors or the KNN algorithm.

The KNN algorithm is going to be a type of supervised machine learning—hence, we will take some time to understand what this is all about a bit more. When you work with this kind of algorithm a bit more, you are going to notice that you will

use it when you want to search through any data that you already have to find the k most similar examples of any instance that you would like to work with. Once you can do this well, then the algorithm is able to move into the data set and will look through and summarize the information that you have. When this algorithm is done, it is going to provide you with the results so that you can make the right predictions for your needs.

Any time that a programmer decides to work with this kind of model, they are going to find that the learning they see is more competitive overall. The reason that this happens and because it works is that there will be competition between the parts or elements. These compete against each other to ensure that, based on any of the data you present to it, you will get the best predictions.

As we go through this guidebook and explore a bit more about the different algorithms, you may start to notice how the KNN algorithm is different than some of the other options. For example, many experts like to avoid this approach because they think it is a lazier option, mainly because it isn't going to create any of the models that you may need for the information—unless you specifically go through and ask it for some new predictions. Depending on what kind of data you are working with this may be a good thing because it ensures that when predictions are made, it is using the most up to date and accurate information.

If your program had just gotten into the habit of trying to make predictions automatically on regular intervals, or just when you added in new data, that could be helpful in some situations. But if you would rather look at the data when it is the most convenient to yourself, or you just want to be able to divide up the data into certain customer groups, then the KNN algorithm and the methods attached to it are going to be the best.

The KNN algorithm can be a good one to work with because it provides you with a lot of benefits. When you are working with this algorithm, you can go through the data and work on cutting out some of the noise. In some cases, the more data that you have, the more noise that you have to get through. And this algorithm is particularly good when you have a large amount of data to go through. It is able to sort through as much data as you would like and give you some accurate predictions, which is a lot more than what some of the other algorithms are capable of.

Now, as you are working on this, you may notice that one of the biggest issues that can come up when you work on this algorithm is that the costs of computing the information are going to be a bit higher than what you see with other algorithms. This is going to occur because this algorithm takes the time to look through each data point that you have, rather than taking those data points and clustering them together. This can sometimes give you the best predictions, but the costs are sometimes too high to make it worth it.

When Do I Want to Use the KNN Algorithm?

The KNN algorithm is a nice one to work with because it works really well with both the classification and the regression problems you have when it comes to prediction. This makes it a very useful tool to go with. With that said, it is going to be used mostly with classification kind of problems. When you are evaluating this technique and trying to figure out if it is the right one for you to work with, there are going to be three things that you should take the time to consider, and these will include:

1. The power of prediction of this algorithm.
2. The time it takes to do the calculations
3. How easy it will be to interpret the output that you have

When you take a look at this algorithm and compare it to other options, including algorithms for logistic regression, random forests, and CART, you will find that it is going to fair well with all of these parameters. Often you will find this algorithm is used because it is easy to interpret the results that it provides, and the calculation costs are going to be lower?

How Do I Use the Algorithm?

Since we have been talking about this algorithm for a bit now, you are probably curious as to how you would be able to work with and use this algorithm to see the results with your data. There are a few steps that you will want to work with when it comes to this algorithm and making sure that it works the way that you want including:

1. Load the data into the algorithm for it to read through.
2. Initialize the value that you are going to use and rely on `for k`.
3. When you are ready to get the predicted class, iteration from one to the total number of the data points that you use for training, you can use the following steps to help.
    a. Start by calculating the distance that is in between each of your test data, and each row of your training data. We are going to work with the Euclidean distance as our metric for distance since it's the most popular method. Some of the other metrics that you may choose to work with here include the cosine and Chebyshev.
    b. Sort the calculated distances going in ascending order, based on their distance values.
    c. Get the k rows from the sorted array.
    d. Get the most frequent class for these rows.

e. Return back the class prediction.

Why Would We Consider This Algorithm Non-Parametric?

This is a question that a lot of beginners are going to have when they get started with this kind of algorithm. When we look at what it means for an algorithm to be non-parametric, it means that we are not making any assumptions of the way that the distribution of the data is done. These kinds of methods won't have a fixed number of parameters that must be met in the model.

With this in mind, when we have the KNN algorithm, the model parameters are going to grow with the training data set. A good way to think about this whole process is that each training case is going to be more of a parameter of the model, and that is basically how this kind of algorithm is going to work.

Is There a Difference Between the KNN and the K-Mean Algorithm?

When we get to the next chapter, you will see that we will spend some time talking about an algorithm that is known as the K-means algorithm, and this can be confusing for those who are just getting into the ideas of machine learning. There are a few differences that do come up with these two methods, and they will include:

1. K-means is more of a technique that you are able to use when you want to do some unsupervised learning projects. It isn't going to have any variables present that are dependent. But with KNN, you will have some variables that are dependent present.

2. K-means is going to use clustering in order to split up the data points and then look at these clusters rather than each individual data point.

When Are Some Times That KNN Will Be Useful?

There are several different algorithms that work well with the model of KNN. So, why would you want to choose to use some of the algorithms over some of the others, when you have so many options to choose from. There are always some benefits and negatives that come with each algorithm that you choose to go with, but some of the benefits that you will enjoy with the KNN algorithm will include:

1. It can work well with problems, even if they are considered multi-class.
2. You are able to apply this algorithm to both problems that are regressive and those that are classification.
3. There aren't any assumptions that come up with the data. This ensures that you get the information that you want, rather than having any assumptions in the place causing some issues.
4. It is an easy algorithm to work with. It is easy to understand, especially if you are brand new to the machine learning process.

However, there are more options of algorithms that you are able to work with because the KNN algorithm isn't going to be perfect in each and every situation that you go to. Some of the negatives that come with using the KNN algorithm includes:

1. It is going to be computationally and memory intensive expensive. If you don't have the right system and the right amount of space to work with them, it is going to make it more difficult to see the results that you want from this algorithm.

2. If there are a lot of independent variables that you are going to work with, you will find that the KNN algorithm is going to struggle.
3. The KNN algorithm isn't going to work that well if you have any rare event, or skewed, target variables.
4. Sensitive to the scale of data.

Now, if you are working on a problem, you are going to have a small value of k that will then lead you to deal with a bigger variance in the predictions that come up. In addition, when you set a bigger value to the k, there are sometimes going to be problems with a large model bias that you need to be aware of from the beginning.

One thing to be aware of here is that there are going to be times when the computer programmer has to go through and create some dummy variables so that they can figure out the categorical variables here. This is going to be something that you do instead of the original categorical variables. This is different than working with regression because you could work with the k dummies, instead of having to work with the (k-1).

For example, you may go through this and create a new categorical variable that is known as the Department. Then, inside of this category, you have five new levels that are unique to that one. Because of this, each of the variables that are dummies are going to have one against its department and else 0.

What Are the Steps to Find the K Value?

And the final thing that we need to focus on in this guidebook is able to find the optimal or best value of K. The best way to do this is through some cross-validation. It is going to be so important to take the time to complete cross-validation to estimate the error that may be present in the algorithm. This is a simple thing to

do because you just need to hold out a subset of your set of training from the process of building a model.

Cross-validation, which is also called 10-fold validation, means that the computer programmer has to go through and randomly divide up the training set so that you have ten groups, or folds, while also trying to get these groups to end up fairly close in size. From that, you will be able to use up to 90 percent of your data in order to train your choice of model. The final 10 percent is going to be saved behind so that you can validate the model at hand.

You will also have what is known as a misclassification rate. You will need to pay attention to this and compute it with the help of that validation data of ten percent that we saved back. This procedure is going to go through and repeat itself ten times. Each of these observations is done in order to validate the information that you want. You will end up with ten different estimates, along with their validation error, and these are averaged out to give you the K value that you need.

Chapter 7: Understanding K-Means Clustering

The next thing that we need to explore here is the algorithm for k-means clustering. This is a great algorithm to use with machine learning, and it can help out a lot with the kind of programming that you do. The basic idea that comes with this is that you are able to take the data from your system—the kind that hasn't been labeled yet—and then put it together into clusters.

Clustering is going to be a type of unsupervised machine learning. It is going to be applied when the data that you have doesn't already come with labels to it—and the goal of using this algorithm, rather than one of the others available, is to make it easier to find the various groups or related clusters that are already present in the data.

The main goal of a programmer working with these clusters is that the objects that end up together in one cluster should be closely related to one another and they

are not going to have a lot of similarities to those that are in other clusters. The similarity here is going to be some kind of metric that will show how strong this relationship is between the data objects.

The field of data mining is going to be able to use this kind of clustering quite a bit. This is especially true if you are doing explorations. But this is not the only field that is going to benefit from this algorithm. You could also use it in fields like data compression, information retrieval, image analysis, pattern recognition, and bioinformatics.

The algorithm is able to work in order to form some clusters of data based on how similar the different values of data are. You will then go through and specify what you would like the value of K to be, which will basically be the number of clusters that you would like the algorithm to make from your data. This algorithm is able to start out by selecting a centroid value that goes with each of the clusters and then it will go through a total of three steps including:

1. You will want to start with the Euclidian distance between each data instance and the centroids for all of the clusters.
2. Assign the instances of data to the cluster of centroid with the nearest distance possible.
3. Calculate the new centroid values, depending on the mean values of the coordinates of the data instances from the corresponding cluster.

Working with the K-Means Clustering Algorithm

To work with this algorithm, the input that you would like to use for the k-means is just going to be found on the matrix X. You can then go through and add in some organization to any choices to ensure that the rows you create will be a different

sample, while each of the columns are going to include a different factor or feature that you would like to consider. To ensure this is going to happen, you have to focus on two main steps.

For the first part, you need to take some time to choose the centers that will make up the individual clusters. If you are looking through the data and you are not sure about where you should put these centers, you may have to start out with some random points for this to see what happens. If you do this and things don't seem to be matching up that well, you can go back through and change it up and see if a new center point is going to work better.

The second step that we need to focus on is going to be the main loop. After you have the chosen centers in place, you can then decide which cluster all of the points of data need to go to. You can look at all of the samples that you have in order to help you figure out which of the clusters the point of data is going to belong to.

From this point, you will need to go through and do some recalculations on the centers of your clusters. You will need to do this based on the points that you assigned for each part. To make this happen, you will just need to grab all of your samples and then figure out what the mean is for these samples. And once you can come up with this answer, you have the k-means.

You will need to go through these steps until you are able to make convergence happen with this algorithm. There are not going to be more changes to the cluster centers once this happens. For the most part, depending on how much data you are starting out with, this is going to be done in five steps or less. But if you have a lot of data points that have a big variance, then it is possible that you will need to do it in more steps.

To understand what is going on, let's take a look at how the k-means is going to work.

This is going to be our initialization point. Here we will have four vectors that are going to be labeled as 1, 2, 3, and 4. The two cluster centers, which are k=2, have been randomly assigned in this one to points 2 and 3. We used the (*) and the (^) signs to help denote these. It is now time to start with the main loop.

The first step that we need to complete during this point is to decide which cluster of our points is going to head over to. Looking above, we will be able to see that both point one and point three are going to be the centers of the clusters on the left because they are both going to be closer to that one than the center cluster. And then the center points on the right are going to be both points 2 and 4.

The second thing that we need to work with here is to do our own recalculation. This needs to be done with the centers of the clusters and will be based on the points that are inside each cluster. The (*) cluster is going to be the one that moves between points 1 and points 2 because this ends up being the main of those points. The same thing will happen with the (^) cluster, but it will move between points two and four. It is pretty easy to come up with the mean for these two points

because of the lower amount of data points, but with more complex data, you would be able to use an algorithm to make it happen. For this example, you would need to use the following code:

You will not see any changes happen in the subsequent iterations, so for this example, we are going to be all done.

The Difference Between Fuzzy and Soft K-Means

As you are working on these, you may notice that when you are working with the k-means, they are going to be really sensitive as you move them, especially to what is known as the initialization. But how do we fix this so that the areas aren't so sensitive, and you will be able to get some accurate answers that you are actually able to work on?

There are several strategies that you are able to use with this one, but one of the best is to go through and restart the k-means at least two times. When you do this, you are able to compare the results and see which one provides you with the best results when it is all done with. We will take a look at how to do this in the next

section. But what it allows us to do here is that we know the cost function and how it is going to be affected, and we can then compare the results and figure out which result is the best one to use.

As you do this, you may find that one of the methods that are going to make it easier to get through any challenges, and will ensure that you are going to find the best results is to add in information that is known as a membership that is fuzzy to each class, and see what is going on in that matter.

For example, your fuzzy point may end up being able to fit in 60 percent of your first cluster, along with matching up to 40 percent of the information in the second cluster. You would then use the process of the soft k-means in order to make adjustments to the algorithm you used before. The first part of it is going to be the same, but you need to make sure that the first k-cluster centers go back to the random points that are a part of your data. But the changes that you will see can come inside some of the loops that you write.

To make this happen, we need to be able to go through and write out some of the responsibilities of the clusters. The best formula to use to make this happen and work out includes:

```
r(kn) = exp[ -b * d(m(k), x(n)) ] / sum[j=1..K] { exp[ -b * d(m(j), x(n))
] }
```

From here, you can see that the `r(k,n)` is going to work out as a fraction, some number that is in between 0 and 1, where you can interpret the hard k-means or the regular k-means to be the case where `r(k,n)` is always exactly equal to 0 or 1. The `d(*,*)` can be any valid distance metric, but the Euclidean or squared Euclidean distances are the ones that are used the most.

Then for the second step, we are going to work on a formula that is similar to the hard k-means, but we are just recalculating the means based on the responsibilities that we want to send with it. The algorithm that we are going to use for this includes:

```
m(k) = sum[n=1..N] { r(k,n)* x(n) ] / sum[n-1..N] { r(k,n) }
```

so, when you take the time to look at this algorithm above, you are going to see that it is similar to the weighted mean. This is going to show you that if the `r(k,n)` is higher, then the mean is going to be more important to the cluster of k. When you see this, it is going to show that this is going to have the biggest influence on the calculation of the mean. But if the mean is higher when you look at the algorithm, you will know that the opposite is true.

What Is the K-Means Objective Function?

Just like what we spent some time talking about with supervised learning before, you must make sure that you spend some of your time looking at and talking about the objective function that you wish to get the most out of with your unsupervised learning. To get started with this idea, we will need to use what is known as the Euclidean distance to make it easier to measure the distance of each center. The function that we will need to use to figure this out includes:

```
J = sum[n=1..N] {sum[k=1..K] { r(k,n) || m(k) = x(n) ||^2} }
```

What all of this means is that you are working with the squared distance that was weighted through the responsibilities that it is given. Hence, if the x(n) part ends

up being far away from the mean of your k cluster, hopefully, the responsibility that goes with this one is set to be quite low. What happens then is known as a coordinate descent. What this is going to mean is that we are looking to move in the direction of a smaller J with respect to just one of the variables at a time. We know that this is something that happened here because we went through and just updated one single variable at a time.

Now, when you take a look back at all of this, you will see that there is actually a bit of a mathematical guarantee that all of the iterations are able to result in the objective function that you work with going down. If you continue to do this for long enough, and you are doing the algorithm in the proper manner, you will find that the points are going to converge together.

Keep in mind that just because you are able to see a convergence show up doesn't mean that they are going to necessarily come up with the global minimum that you are looking for. The numbers will go off the patterns that show up there, and they will spend time focusing on math, and less about what you may think is the most important. Going through and reading through the results and making sure that they work how you want, and retrying the whole process a few times to see if it comes up with different results can help to fix any issues this can bring up.

How to Add in the K-Means?

In this chapter, we have taken some time to take a closer look at k-means and how this algorithm is able to work, along with the different ways that this algorithm is used to create the right solution to your program. Now that this is done, it is time to implement in some of the ideas that we have talked about, with the help of machine learning skills and Python to ensure that this can work. And the best way to make all of this come together is to implement the soft k-means into your code.

But, how are we supposed to make sure that this happens? For this to work, you have to make sure that you have your standard imports in place, and the functions of the utilities ready. This will be pretty much the same thing as the Euclidean distance as well as the function of cost put together. The formula that can help you figure out this information is going to be below:

```
import numpy as np
import matplotlib.pyplot as plt


def d(u, v):
    diff = u - v
    return diff.dot(diff)



def cost(X, R, M):
    cost = 0
    for k in xrange(len(M)):
        for n in xrange(len(X)):
            cost += R[n,k]*d(M[k], X[n])
    return cost
```

Once you have taken the time to add this part into the compiler and you use it to define your function so that it is able to run the k-means algorithm before it plots out the results. This will mean that you will end up with a scatterplot of the information. And the colors that are there will represent the amount of membership of the information found in each cluster. To make this happen, you need to work with the code below:

```
def plot_k_means(X, K, max_iter=20, beta=1.0):
    N, D = X.shape
    M = np.zeros((K, D))
    R = np.ones((N, K)) / K

    # initialize M to random
    for k in xrange(K):
        M[k] = X[np.random.choice(N)]

    grid_width = 5
    grid_height = max_iter / grid_width
    random_colors = np.random.random((K, 3))
    plt.figure()

    costs = np.zeros(max_iter)
    for i in xrange(max_iter):
        # moved the plot inside the for loop
        colors = R.dot(random_colors)
        plt.subplot(grid_width, grid_height, i+1)
        plt.scatter(X[:,0], X[:,1], c=colors)

        # step 1: determine assignments / resposibilities
        # is this inefficient?
        for k in xrange(K):
            for n in xrange(N):
                R[n,k] = np.exp(-beta*d(M[k], X[n])) / np.sum( np.exp(-
beta*d(M[j], X[n])) for j in xrange(K) )
```

```
        # step 2: recalculate means
        for k in xrange(K):
            M[k] = R[:,k].dot(X) / R[:,k].sum()


        costs[i] = cost(X, R, M)
        if i > 0:
            if np.abs(costs[i] - costs[i-1]) < 10e-5:
                break

    plt.show()
```

Notice in this one that both the R and the M are going to have their own matrices. The R is going to be a new matrix because it is able to hold onto two different indices—n and k. But the M will also become a matrix because it is going to include the D-dimensional vectors of K. The beta variable is going to be there because it is responsible for controlling how spread-out or fuzzy the memberships of the cluster are, and it is known as the hyperparameter. From this, we will be able to create a new main function that will create the random clusters and then calls up the functions that we defined above. The code that you will need to use to make all of this work well together includes:

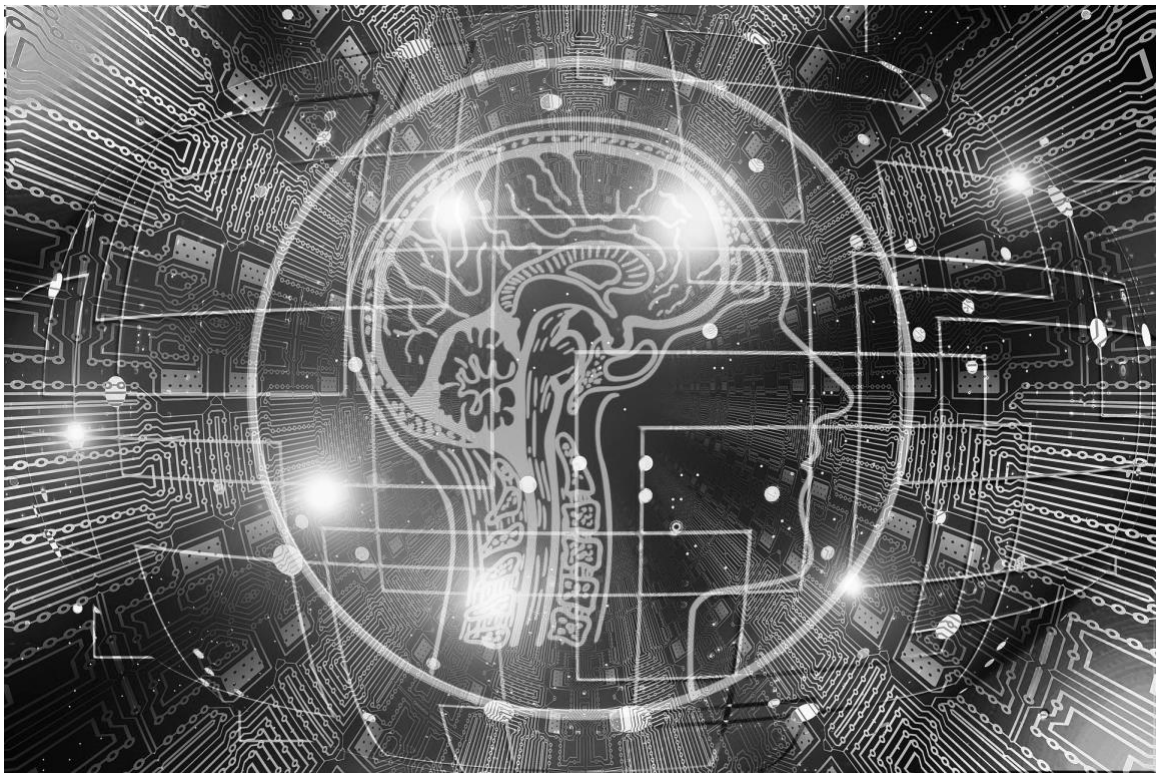
```
def main():
    # assume 3 means
    D = 2 # so we can visualize it more easily
    s = 4 # separation so we can control how far apart the means are
    mu1 = np.array([0, 0])
    mu2 = np.array([s, s])
    mu3 = np.array([0, s])
```

```
    N = 900 # number of samples
    X = np.zeros((N, D))
    X[:300, :] = np.random.randn(300, D) + mu1
    X[300:600, :] = np.random.randn(300, D) + mu2
    X[600:, :] = np.random.randn(300, D) + mu3

    # what does it look like without clustering?
    plt.scatter(X[:,0], X[:,1])
    plt.show()

    K = 3 # luckily, we already know this
    plot_k_means(X, K)

    # K = 5 # what happens if we choose a "bad" K?
    # plot_k_means(X, K, max_iter=30)

    # K = 5 # what happens if we change beta?
    # plot_k_means(X, K, max_iter=30, beta=0.3)



if __name__ == '__main__':
    main()
```

With the formulas that we have gone through above, you will be able to work with the k-means algorithm, and you should have a better understanding of how this all works. Open up your compiler and take some time working on some of these codes and getting a bit of practice with writing your own codes in Python. You will find

that this can really sharpen your skills and makes it so much easier for you to get the results that you want from your machine learning.

Chapter 8 What Are Support Vector Machines?



Now that we have taken some time to look at the other methods that work well with machine learning, it is time to learn more about support vector machine or SVM. This is going to be something that you can often utilize for many of the machine learning challenges that come up—both with classification and regression issues. When you work with this one, a lot of the work that has to be done on problems that are more classification can be a bit tricky. However, the algorithm for SVM will be able to help out with this.

When you want to do some work with the SVM algorithm, you need to take each of your items in the data set and plat them so that they come to just one point on your n-dimensional space. The N is going to be the number of features that you would like to use with this one. Then you are able to take the value of these various features and make it so that it translates over to the values that tend to show up on your coordinates. The next step, once this point has been reached, is to determine

where the hyperplane is going to be. This part is important because it will show you what differences are there in your classes.

When you are working with this kind of algorithm, you may notice that it is possible for more than one of these support vectors to show up. You will not be able to use all of these, or your information will end up being really confused. You want to work with just the SVM so that you can separate out this information into classes, and then you will know which one to focus on in the line and hyperplane.

Here you have reached a point where a lot of the information may not make all that much sense. And you may not even know why you would like to learn and use SVM in the first place. But it is really a powerful algorithm that you are able to work with, and there are a few steps that a beginner is able to work with in order to make sure that they are really getting the full benefits of all this.

So, to help us start here, we need to look at our hyperplane. As you go through all of this, you may notice that there are a few different hyperplanes that you are able to pick out from. And there is also another challenge that is added here in that you want to be able to make sure that you understand all of the options, ensuring that you choose the best one for your needs. But even though you are going to end up with a few different options, there are some simple steps that you can utilize in order to help you pick the right one for your needs. The steps that you need to follow to make this happen includes

- We are going to start out with three hyperplanes that we will call 1, 2, and 3. Then we are going to spend time figuring out which hyperplane is right so that we can classify the star and the circle.
- The good news is there is a pretty simple rule that you can follow so that it becomes easier to identify which hyperplane is the right one. The

hyperplane that you want to go with will be the one that segregates your classes the best.

- That one was easy to work with, but in the next one, our hyperplanes of 1, 2, and 3 are all going through the classes, and they segregate them in a manner that is similar. For example, all of the lines or these hyperplanes are going to run parallel with each other. From here you may find that it is hard to pick which hyperplane is the right one.
- For the issue that is above, we will need to use what is known as the margin. This is basically the distance that occurs between the hyperplane and the nearest data point from either of the two classes. Then you will be able to get some numbers that can help you out. These numbers may be closer together, but they will point out which hyperplane is going to be the best.

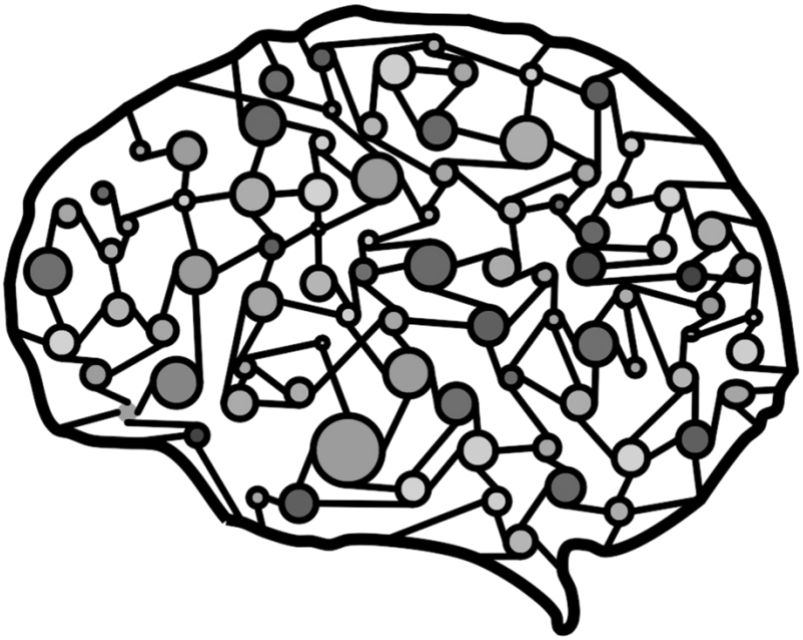
As you can see, we just did an example above, and it is a great one to show you how to work with the SVM algorithm and get the most out of your machine learning project. When you would like to take a look at the different points of data that you have available, and you notice that there is actually a good margin that is able to point out separation, then this algorithm is the best one for you. In addition, you will find that this model is effective and it helps you to increase any of the time that you are working on this kind of project.

As a programmer, it is likely that you are not going to use this model all of the time. But even with this fact, working with this technique can help you to really do a lot with machine learning. It helps you to use a subset of your points for training so that you can figure out where the connections are and how to make them work.

While there are several benefits that come with using this method, depending on what kind of data points and what kind of project you are working with, there will still be some times when SVM may not work for you. If you are using a set of data

that is really large, this model runs into issues being as accurate as you would like. The training time that comes with these bigger data sets is high, and you may not be able to get the information that you want as quickly as you would like.

Chapter 9: Using Neural Networks



The next topic that we are going to take a look at with machine learning is going to be the neural networks. These are going to also fall into the category of unsupervised machine learning. These networks are the ones that are often used because they will ensure that you can look through your data and notice if there are any patterns present in the information. This is going to be done at different levels, and it is much more effective and faster than what the human eye can do on its own.

When we look at the neural networks, each of the layers that you go through will get the algorithm to stop and see if there is some kind of pattern found in the image that it is looking through. If the network isn't able to find a new pattern once they go down another layer, then it will start taking the necessary steps to help it move on to the next layer. This process continues through one layer after another, until all of the layers are created. The program, if it is doing its job in the right manner,

is going to be able to give you a good prediction back about what is inside the image that you scanned.

There are a few things that could possibly happen when you enter this point, based on how the program is working. If the algorithm was able to go through the process above, and it did a good job at sorting through the layers, it will then provide you with a prediction. If the program is right in its prediction, the neurons of this system, just like the neurons in the brain, are going to become stronger.

The reason that this works so well is that the program decided to work with artificial intelligence, which allowed it to make strong associations between the object and the patterns that it found. The more times that the system is able to look at a picture and come back with the right answer, the more efficient it is going to be the next time that you use it.

To find out how this works will require us to look a bit closer at how the neural networks work together. For example, let's say that you have set out to create a new program that is able to hold onto a picture, which will be your input. And then it can look through the various layers of that picture until it figures out that the image inside is a car.

If you have gone through and coded this in the right way, the program is able to make the prediction that the picture is that of a car. The program will be able to come up and present you with this prediction based on some of the features that it knows, from past experience, belongs to a car. It could look at the headlights, the placement of the doors, the license plate and more to come up with this prediction.

When you are looking at your skills in conventional coding and what all is available to you through this, then this process is going to be a difficult one to do. You will

find that with machine learning, and the neural networks algorithm, you will be able to get this to work without too many problems.

But how do we get this algorithm to work? To make this happen, you need first to make sure the program has an image of a car to compare the newer image to. The neural network will take this learning picture and look it over. It would be able to start out with the very first layer. In this case, this is going to be the edges on the outside of the car. From there, the program would go through other layers that are going to help the neural network learn whether there are any characteristics that are unique to the picture and will tell it that a car is in the picture.

If the program has worked with trial and error and it is good at doing this job, it is going to make the right prediction. And the more pictures of cars that you provide to this program, the better it is going to get at finding the car and predicting it, and the more minute details it will notice.

Depending on the type of picture that you are working with, there could be a lot of potential layers that come with this algorithm. The good news here is that the more details and the more layers that your neural network is able to find, the better accuracy it comes with when it predicts the car and what kind of car even. If the neural network is accurate and does a good job with the predictions, it is able to learn from these lessons. It is going to remember what it learned as it went through the layers and can store that information to use later on. If it does need to look at another similar picture in the future, it will be able to make a very quick prediction in the process.

Any time that a programmer wants to work with the neural network algorithm, you will often be working with things like face recognition software and other similar projects. When this happens, all of the information that you and the program need

won't be available ahead of time. But you are able to use this method in order to teach the system the best way to recognize the right faces in the image. You can also use this one to help with different types of animals, to define models of cars, and so much more.

As you can imagine reading through this chapter, there are a lot of different advantages that come with this particular model when working on machine learning. One of the advantages that you are going to notice is that you can utilize these methods without having to control the statistics of the algorithm. Even if you need to use it without the statistics being available, the neural network will still be able to finish the work for you. The reason that this ends up working so well is that both the dependent and the independent variable are going to be nonlinear.

There are a few times when you will not want to work with this method. One of the main reasons that programmers would choose not to go with a neural network is that it is another one of those models that has a high computing cost to get the information. For some of the smaller businesses who are interested in working with machine learning and doing this kind of technology, the cost is just going to be too much time, computational power, and money and they will need to look at some other algorithms instead.

Chapter 10: The Random Forest Algorithm

When you start to learn about machine learning, there are going to be many instances where you hear about decision trees and random forests. Both of these are important algorithms that you should learn how to use, and you will find that they often work together to help you get the right results with your data sets. In this chapter, we are going to being our exploration with some decision tree information to learn more about them—then, we will move on to the random forests and how they compare and work with the decision trees.

First on the list is the decision trees. These are really efficient tools for data if you would like to compare and contrast two choices, especially if those two choices are very different. You can then use the information that you were able to gather in order to pick out the right one to follow and in the process grow and improve your business. When you see a presentation of the various options presented to you, you

will be able to utilize the algorithm for a decision tree in order to see the possible outcomes. This is a great way to make predictions and come up with the best decisions for your needs.

The neat thing about decision trees is that there are a few different ways that you are able to work with them—many of those in machine learning like to work with these decision trees for random or categorical variables. However, there are going to be times when you can use decision trees to help with problems of classification. To make sure that you do this process accurately and you have a good decision tree, you will need to take up all of the sets of data that you plan to work with, and split them up into at least two sets. And in each of these sets, the data needs to be similar. Once you are done with that, the sets can be sorted more to help you figure out what is important and what isn't.

So, this can bring up the idea that this is a hard algorithm to work with. To make certain that you are doing the decision trees in the right manner and that they are going to work for you, you first need to check out this example. For this one, we are going to take a look at a class that has 60 people inside of it. There are three variables, all that are considered independent of each other, that we need to consider about these students. And the independent variables are going to include class, height, and gender. In addition, as you look over the information about the students, you see that half of the class doesn't like soccer and doesn't like to spend their time with this sport.

Now that we have a bit of information to go from, it is time to see how the decision tree can help us with this. Given this information about the students who like to play soccer, you decide that it is time to create a new model that will help you figure out which of these students likes soccer, and which ones don't. This is possible with the decision trees.

In order to help us make this kind of model, the decision tree needs to look at all of the students, and then find a good way to divide them up into the groups where they belong. The variables that need to be used here would include the height, gender, and class from before. The goal of this one is that when everything is done, you will be able to have a homogenous set of students, ones who like to do the same things together.

When you are working with a decision tree, there are going to be several algorithms that can be used along with this. And each of them will make it possible to split up the data that you have. Working with the right algorithm will help you divide things up into subsets that you can work with, and then they will produce good outcomes that are as homogenous as possible so that you can make the right decisions for your needs. Remember that you can add in more subsets if the situation needs it. For this particular example though, we are working with two which includes the students who play soccer and the ones who don't like soccer.

This is a simplified version of this idea, but there are times when you will be able to work with the decision trees to go through some complex data and sort out the points based on their differences and their similarities. These decision trees are going to provide you with the data that you need, and then you are able to take that data and make some informed and smart decisions for your business.

Yes, in the past business owners would use the knowledge that they had about the industry and their own intuition to help them make decisions about how to act in the future. But decision trees are often going to make the work easier, and are more accurate than what you are going to see with traditional decision-making forms. The decision tree can help you to look through more information, can sort it out

easier, and can make it easy to compare the different options rather than trying to do it all on your own.

Starting with the Random Forests

There are also going to be times when a decision tree just isn't quite what you need with your project. This is often going to happen, and when it does, you may want to take a bit of time to work with the algorithm that is called a random forest algorithm. These algorithms are going to be popular for you to work with, so if you would like to do machine learning on a regular basis, or you plan to really go into the field of data science, then this is a machine learning algorithm that you need to learn more about.

Since these random forest algorithms are so popular and well-known, you can guess that the algorithm can be used to help out with a lot of different challenges here. For example, if you would like to work with tasks that can look through the data and explore it a bit, like dealing with information when the data is missing, or treating for any information outliers that are present, then the random forest is going to be the number one algorithm to use for this.

There are a few times when you are doing some machine learning process, and then the random forest will need to be brought out. This is due to the fact that the random forest is going to be perfect when it comes to providing you with some good results—and oftentimes, they are able to complete the job of other algorithms or do the job better. Some of the different benefits and uses of these random forests that will help you to see the best results will include:

1. Any time that a computer programmer is trying to do their own sets for training, they will find that all of the objects that are on the inside of the set can be generated in a random manner. And there are times when it can be

replaced if your random tree thinks that this is something necessary and important for your needs.

2. If there are M input variable amounts, then this means that the `m<M` will be specified right from the beginning. In addition to this, this information is going to be a constant help throughout the project. The reason that we need to remember this is because it means that each tree that you have has been picked at random for their own variable using the M from before.
3. The goal of these trees, no matter how many of them you end up with, is to help you find the split that is the best for the "`m`" variable.
4. As your tree starts to grow more and more, all of them are going to get as big as they can with the data you have. Remember that these are not trees that can take the information out or do any pruning of themselves.
5. The forest that you get from these random trees can be a good resource to use because it is able to predict certain outcomes for you. It is going to take over this job because it takes all of the predictions that you get from the trees that were created, and then it can select the average for the regression, or provides you with the consensus that you will get during a classification challenge.

You will find that these random forests are a good tool for programmers to use when they want to add in a bit more data science to their machine learning. And as you can see from the information above, there is a lot of advantages to working with these random forests, rather than using the other algorithms that are available.

The first big benefit that you will notice with the random forest is that they are able to deal with all machine learning challenges, whether they are regression or classification challenges. Many times, the other machine learning algorithms that

you will work with can be *either* a regression solver or a classification solver—not both of them.

Another great benefit that comes with these random forests is that they are able to work when you need to sort through a large amount of data. And you get to go through and add in hundreds of thousands of variables (if it is needed), and the algorithm is still able to handle the work. Of course, it isn't that likely that these many variables are needed when doing this, but it is nice to know the amount of power that is used with the random forest algorithm.

Just like with all of the other algorithms that we have talked about so far in this guidebook, there are a few negatives when it comes to doing random forests. First, even though the random forest is able to do some regression problems in machine learning, they are not going to be helpful when it comes to the predictions that you should put in when it concerns the training data. It also can't talk about the ranges that are present there either.

What this means is that there are times when the random forest is able to make a few predictions that are helpful to you. And there are some times when you can use these predictions to help you make some amazing business decisions. But this algorithm is still going to come with a few limitations because you won't be able to go past the ranges that you provide the algorithm, and this decreases the amount of accuracy that you are going to get.

Chapter 11: What Are Recurrent Neural Networks?

We spent some time earlier in this guidebook talking about neural networks and how they work to go through the different layers of pictures to provide you with the information that you need concerning what is in that picture. Now, we are going to take this a bit further and talk more about an algorithm that is known as recurrent neural networks.

In order to help us get started with this one, we need to take a closer look at the human brain. When we look at the human brain, it is understood that we aren't going to restart the way that we think, going all the way back to nothing, each second. You are able to learn something, and then use that knowledge and build upon it all of the time. Whether that knowledge is from your childhood or five minutes ago, you don't have to start from nothing each time this happens. You are able to keep on learning, building upon each experience that occurs in your life, in order to increase your own knowledge.

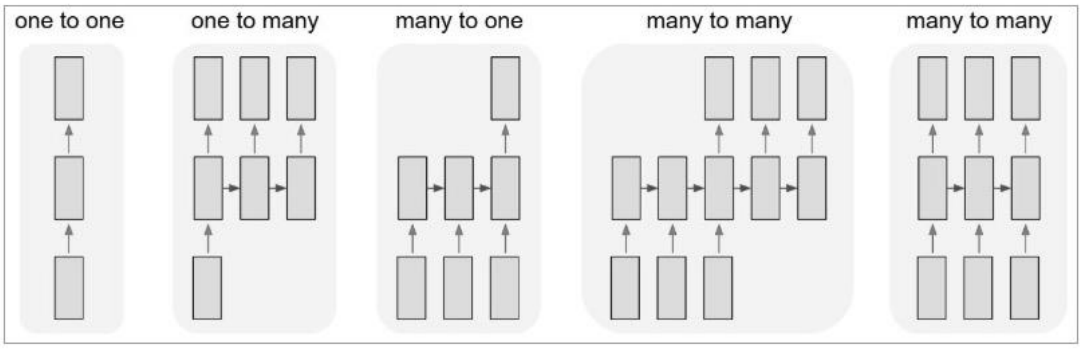
When we look at the traditional neural network with machine learning that we looked at earlier, this process isn't able to do the same thing as the brain. And this can sometimes be a big shortcoming based on the project that you want to work with. For example, if you would like to do some classification for the kind of event that occurs during a movie, it is hard to do this with the traditional neural network. This is because you wouldn't be able to remember what happened in the move earlier through that algorithm, and you would run into trouble knowing when different things happen.

This can seem like it is going to cause issues with your project severely, but that doesn't have to be the case. The recurrent neural networks that we will explore in this guidebook are meant to be used in order to address this exact issue that shows up. These are going to be networks that contain loops inside, which allows the information to stick around, rather than leaving. In this kind of method, the loop is there to allow information to pass from one part of the network on over to the next part as well. A recurrent neural network can be a similar idea as having more than one copy of the exact same network, and each message is sent over to the right spots.

The nature of this chain is there in order to reveal that the recurrent neural networks are going to be related very closely with sequences and lists. They are going to be the natural architecture of the neural network to use for this kind of data. And you will find that in your programming, you are able to work with these quite a bit. For example, over the past few years, programmers found success when they were able to apply these ideas to a variety of problems in machine learning including with speech recognition, translation, captioning, and language modeling.

Now, while there are some benefits that come with this process, it is important to realize that there are going to be some limitations that come with this kind of network. The biggest one is that the API is going to have a lot of different constraints put onto it. These API's are only going to be able to accept fixed vector sizes as their input, and then they are only able to produce a fixed-sized vector as the output.

This is a big problem, but it is just one of the problems that you are going to need to deal with. You may also find that these models are going to perform this kind of mapping with a fixed number of steps computationally, which equals the number of layers that will show up in this particular model. The main reason that these options are going to add in some excitement and something new to the program is that they will allow the programmer to operate their work over a sequence of vectors. This can usually happen with both the output and with the input.



Now, the chart above is going to help us to understand more about these recurrent neural networks. Each of the rectangles that you see there will be our vectors, and the arrows are there to show us the functions. When we look at the vectors for input, they are the red ones. And then the vectors for output are going to be blue. Those green vectors are going to be the state of RNN (Which we will explore in a minute). So, when we go from the left of this chart over to the right, we are able to see the following:

1. In the first one, we are looking at what is known as the vanilla mode of processing. This is going to be the processing method that won't use the RNN at all. This is going to include a fixed input and a fixed output. We would also call this one image classification.
2. For the second part, we are looking at the sequence output. This will be what we see with image captioning. This is going to be where you take an image and then the system is able to provide you with an output of a sentence describing it.
3. Sequence input is going to be what we see with the third part of the chart. This one is going to be seen as a sentiment analysis that is able to show us a given sentence and then helps us to classify this in the proper manner, either as a sentiment that is positive or a sentiment that is negative.
4. The sequence input and output. You will then be able to move on to the fourth box. This one is going to help us get closer to what we are looking for here. This is going to be similar to what we see with a machine translation. When the RNN is able to read out our sentence in English, and then it takes that information and provides you with a sentence then you reach the fourth part.
5. In the last box, we get to the final part of this process. This is going to be where the input and output of the sequence can be synced together. The classification of the video is going to help us to label the frames that we have—if we decide to go that far.

Notice that when we go through each of these different parts, there are not going to be any kinds of constraints put on the length of the sequences that we have to specify ahead of time. This is due to the fact that this transformation, which is shown with the color of green, is fixed, and we are able to use it as many times as are needed to finish up this project.

What Is the RNN?

We took a moment to mention the idea of the RNN in the previous section, but now we need to take some time to explore what all of this is about. In this example here, we are going to work to train the RNN character level language mode. What this ends up meaning is that we will need to provide the RNN is a large amount of text, and then we will go through with the model of the probability of distribution of the next character that is going to show up in a sequence of previous characters. This ends up being a good thing because it will make sure that we are able to get some kind of text, even if the system is just able to do one character at a time.

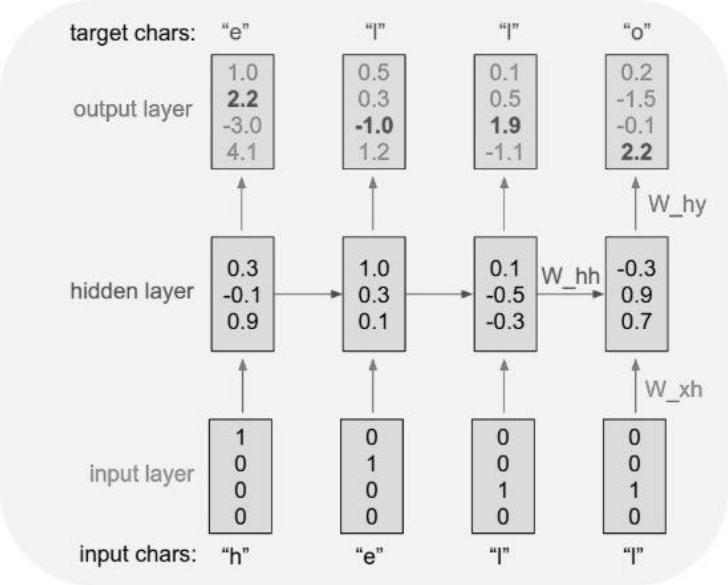
As an example, suppose that we had a really limited vocabulary to work with, and just four possible letters `(helo)`. With this information, we want to train the RNN to do a training sequence of "hello" instead. This training sequence is going to be a source of four separate training examples. This may sound a bit complex when we are first getting started, but we will take this step by step to help you see how it works. Some of the things to consider to get this started includes:

1. The probability of getting `"e"` should be just as likely to occur as getting the letter `"h"`.
2. `"l"` should be likely in the context of `"he"`.
3. The `"l"` should also be likely if the system is given the context of `"hel"`.
4. "o" should be likely if the other sequences have happened and the context of `"hell"` is in place.

Now, we are going to encode each of the characters that occur in the vector working with the "l of k" in coding (this is going to mean that we will use all zeroes except for a single one at the index of the character in the vocabulary). We will then be

able to feed them into the RNN, doing so one at a time, using the function for step to make this happen.

Once all of this has gotten into place, we are going to observe a sequence of 4-dimensional output vectors, with one dimension showing up for each character, which we can then interpret as the confidence that the RNN is able to assign right now to each of the characters that come up in the next sequence at a time. Let's take a look at a diagram that can show up when it is time to run this kind of sequence.



Out of this, we may use an example of seeing that in the first time step, the RNN would see the character of "h", it was able to assign some confidence to this of 1.0, to the next letter turning into "h", 2.2 to getting the letter "e", 3.0 to "l", and 4.1 to "o". Since the training data that we are using (which is the string of hello that we talked about above) had the next right character being "e", we want to increase the confidence—or the green color—and then decrease how confident it is in all of the other letters, which are going to be shown in the red above.

The next thing that we will need to do is go through another step and figure out the character target that we want to have at each of the four steps that we would like this network to assign its confidence to. This can be a bit time consuming to do, but it is going to be helpful because it ensures that we get the right confidence levels to ensure the system will put the right letters in at the right time.

Since we are going to work with this RNN feature, and it is going to consist of differentiable operations, we will be able to use the back-propagation algorithm. This is going to include an application that is recursive of the chain rules that you should remember from calculus. We will then take this information and use it to help us figure out the right direction, moving the weights as we need to help us increase the scores on the targets that we want, which are shown in the green and the bold numbers above.

Once we are done doing this work, we will need to go through and do an update of our parameter. This is going to be the thing that we need to help nudge the weights just a bit towards the direction that we would like. If you are able to take some time to check the inputs that are fed into the RNN end up the same after each update on the parameter, you will find that the scores of the correct characters are going to be higher and that the ones that are not right will go down by a little.

Now, it is possible that you will need to go through and do this process more than ones, and sometimes you will need to do it often, to ensure that it is going to work for you. The number of times that you need to make this happen is going to depend on the complexity of the system. Since the example that we are going to use here is the “hello’ code, it won’t take that much to get it done.

But since machine learning can be pretty complex and will need a lot of programming, it is possible that this is going to be complicated enough that you

have to go through and do it many times. You have to be prepared to repeat this process until you are able to get the network to converge properly, and you can get the predictions to be consistent with the data for training that you are using.

Another option to work with here is a technical explanation. You are able to do this once you have the classifier that is considered standard for Softmax. This can sometimes be referred to as the cross-entropy loss, on every output vector at the same time. If you are able to do this, you are basically training the RNN with a mini-hatch stochastic Gradient Descent, and you may find that you need to add in RMSProp or Adam to make the updates that you do as stable as they can be here.

If you are able to take some time to do all of this, you will start to notice that the first time that you add in your character as an input of "l", the target is going to be "l". But if you go through and do this again, it is going to be "O". The RNN is going to need to have some help because the input is not going to be enough all on its own. This means that it needs to have a connection that is recurrent to help keep track of the context that will make this task as achievable as possible.

After you do the iterations a few times, you will get to testing—it is time to go through and add in the right kinds of characters to the RNN algorithm. As you are training, you should be able to see the distribution and how it will look if you make a certain prediction. The computer programmer needs to be able to take a sample from the given distribution and feed it right back into the algorithm to tell you which letter is going to show up next. Continue to go through with this process again and again until you are able to get the right letter that you want, in the right order.

As you can see, there are going to be many different things that are possible when you decide to work with the algorithm RNN, and many of these are impossible to

do if you are working with some of the other algorithms available through machine learning. Using the RNN algorithm is going to open up a lot of doors and will help you to handle more machine learning situations in the process as well.

Chapter 12: Linear Classifiers

As you go through and work with some machine learning, especially when it comes to supervised learning, you may find that two of the most common tasks that are going to be required here will include both the linear regression and the linear classifier. The linear regression is helpful in some situations because it is able to predict the value of your data—then, the linear classifier is useful because it focuses on the class. While both of these are useful in machine learning, we are going to spend our time looking at the linear classifier and some of the steps that you will be able to use when you bring this up in machine learning.

You will find out quickly that when you are working on machine learning, these kinds of classification problems are very prevalent, and they will take up a minimum of 80 percent of the tasks that you will do in machine learning. The aim

of classification is to predict how probable it is that each class is going to happen given the inputs that you decide to put in, the label (the label is going to be the dependent variables here), and the class.

If your dependent variable or the label only comes in with two classes to work within the beginning, then you know that the algorithm that you are working with is a binary classifier. If you would like to work with a classifier that has more than one class, this means that it will be able to tackle any of the labels with three or more classes.

Let's look at an example of this one. Many of the classification problems that are also considered binary are able to predict how likely it is that your customer will come back, after making one purchase, and purchase again. But if you would like the system to make a prediction about the type of animal that you have placed into an image, you will instead work with a classification problem that is known as multiclass. This is because there will be more than two types of animals that can show up in the picture.

Measuring Your Linear Classifier Performance

Now, we need to take a look at the linear classifier a bit more and how you are able to measure how well it is able to perform. Accuracy is important with any of the learning algorithms that you work with, and it is one of the best places to start. The performance overall of this classifier is going to be measured using the metric of accuracy—this is how important it is for you.

When we talk about accuracy, it is the measurement of whether the algorithm is able to collect the proper values that you have, and then it is able to divide that number by the total number of observations that are present. Looking at an example of this, if you have an accuracy value that is set at 85 percent, this means

that your algorithm is going to be right 85 percent of the time, but then it is going to be wrong the other 15 percent. Of course, if you are working with numbers and data, you want to make sure that the accuracy is as high as possible.

As you take a look at the accuracy, you should also note that there is a shortcoming with this metric, and this is never more apparent than when you are looking at a class of imbalance. A set of data that ends up not being balanced is going to occur when the number of observations that show up isn't equal in all of the groups that you have.

To understand this, let's say that you are doing a classification challenge of a rare event with the function of logistics. You would need to think about the classifier that you would like to use, and that will include estimating how many patients died when they were in contact with a disease. In the data, you see that five percent of the infected patients were going to die.

From this information, you would be able to train your chosen classifier to make sure that it is able to predict the number of deaths is going to occur for those with the disease. Then you can go through and use the metric of accuracy in order to make an evaluation of how that clinic or hospital is performing. With this example, the classifier is able to go and look at the information. Maybe it predicts that there are going to be no deaths for the set of data as a whole. This is going to be accurate about 95 percent of the time, which gives it some margin of error.

The next thing that we need to focus on here is something known as the confusion matrix. This is going to be a better way to take a look at how well the classifier is able to perform compared to the accuracy that you were able to do above. When you decide to bring in the confusion matrix, you will be able to get a visual about how accurate the classifier is by comparing the actual and the predicted classes.

The confusion matrix that is binary is going to consist of squares. If you decide to work with this kind of confusion matrix, there are a few different parts that come with it including

1. TP: This is going to be known as the true positive. This is going to contain all of the predicted values that were correctly predicted as an actual positive.
2. FP: This is going to be the false ones—or the ones that were predicted in an incorrect manner. They were usually predicted as positive, but they were actually negative. This means that the negative values show up, but they had been predicted ahead of time as positive.
3. FN: This is a false negative. This is when your positive values are predicted as negative.
4. TN: this is going to be the true negative. These are the values that were predicted in a correct manner and were predicted as actual negative.

When a computer programmer takes a look at this confusion matrix to use it, they are going to get a lot of information to use. The confusion matrix is able to help you get a nice clear look at the predicted class, and the actual class, so that you can compare and contrast what is going on with the data that you have.

Are Sensitivity and Precision Important?

The confusion matrix is one of the best things that you are able to work with when it comes to looking through the linear classification and understanding the information that is in your data. But when you do this kind of matrix, you will find that a lot of information will come your way. These matrices are also going to give you some insights when it comes to a true positive and a false positive in the information that you have. However, even though this is a great thing to work with, there are still some cases when it would be better to go with a more concise metric to help you understand the information at hand.

The first thing that we need to look at here with the metric that you use is precision. The precision metric is important because it shows us how accurate the positive class is going to be. This may sound confusing, but it means that this precision metric is going to provide you a measure of how likely the positive class prediction is going to be correct in the long run. In order to do this, you can use the formula below to make this easier:

```
Precision = TP / (TP + FP)
```

The maximum amount, or score, that you are going to end up with here is one. And this is going to show up the classifier and be correct when you have a positive value. While precision, and knowing how precise the metric is ahead of time can be super important to the success that you get, it is not going to be all that you need to look for, mainly because it is able to ignore the class that is negative. This metric is something that you would be able to use, but it is more helpful if you pair it up with what is known as the recall metric. The recall is also going to be known as the true positive rate, or the sensitivity rate.

From this point, we also need to have a good idea of the sensitivity that comes with this learning algorithm. This sensitivity can be important because it will let you know the ratio of positive classes that your algorithm is able to figure out correctly. The metric is a good way to model and take a look at a positive formula that will help you work with this algorithm and figure out its sensitivity will include the following:

```
Recall = TP / (TP + FN)
```

Using TensorFlow and Linear Classifiers Together

Earlier in this guidebook, we took a look at what the TensorFlow library is all about and why it would be a useful option to work with. Now, we are going to take a look at this idea and discuss some of the ways that you are able to use this library, especially when it comes to a linear classifier algorithm. The first thing that we are going to be able to do to help combine the two of these is to use a data set that is a census.

The whole aim of doing this is to help us use the variable in a census data set to help us come up with a prediction of the income of those people who are going to participate in this data. Note that the income for this example is going to be known as a variable that is binary.

For this example, we are going to set the variable that is binary at one of the income of the individual, and it is set to be above $50,000. However, if the income ends up being under this dollar amount, then we will need to write out the variable that is binary as 0. The data set that we are able to work with to make this all happen can be written out in eight variables that are categorical and include the following:

- Native country
- Sex
- Race
- Relationship
- Occupation
- Marital status
- Education
- Place of work

And on top of this, we are going to take a look in this at six of the continuous variables. These are going to include:

- `Hours_week`
- `Capital_loss`
- `Capital_gain`
- `Education_num`
- `Fnlwgt`
- `Age`

Once we have this information, you can open up the TensorFlow library in order to figure out what the probability is. This probability is going to help us figure out which customers are able to fit into each group (which ones make more than $50,000 and which ones earn less than $50,000).

In this example, you are going to separate each of your customers into two groups. The first group will be the individuals who listed themselves as making more than $50,000. And the second group is going to be individuals who will make under $50,000. When the individuals are separated out into these two groups, you will then be able to look more at their background information, and figure out some information about them like their race, sex, where they work, where they live, and anything else that you would like about them. This can provide some valuable insights for a business to learn more about their customers and how they may purchase things in the future.

Many businesses like to do this because it allows them to learn more about their customers, and ensures that they are able to figure out who to advertise too. It can

also be useful when it comes to customers who come back and make purchases more than once. The business would be able to use this information to figure out how likely it is that a first-time customer is going to come back, and can change up their marketing and advertising to reach these kinds of people more often.

Discriminative Models and Generative Models

At some point, you will need to create some parameters that go with your linear classifier. And when it is time to work with these parameters, there are two classes or methods that can help you with this. These are pretty broad, and they are known as either the generative model or the discriminative model.

Methods that come with the generative model are going to be functions that look at the conditional density. The examples of these types of algorithms that you may use with machine learning will include the Linear Discriminant Analysis, which will assume the Gaussian conditional density models, or you can work with the classifier that relies on the Naïve Bayes algorithm.

The other method that we brought up above is going to be the discriminative model. These are going to be important because they are going to work to make sure that any output that you get from the program is as high in quality as it can be, especially when you work with the training set. It is possible that when you are doing with the training that you will need to add in some additional terms, but they can cost more and may be able to perform the regularization of the final model that you get. There are a few options that the computer programmer is able to choose from when they work with discriminative training, and these options will include:

1. Logistic regression: This one is important because it is able to show us the likelihood estimation of linear classifiers. It is going to do this with the assumption that the observed training set is going to be generated through

a binomial model and it is going to really depend on the kind of output that the classifier provides to us.

2. Perception: This is a type of algorithm that you would like to use in some of your machine learning because you will use it to fix up any errors that may occur in your training set, which makes it more accurate overall.
3. Support vector machine Remember this one from before? It can work with your discrimination model as well. This is going to be the type of algorithm that you will be able to use and maximize quite a bit. You can use examples that are able to hose up in the hyperplane of the decision and in the training set to help with accuracy.

Despite the name that comes with it, the LDA is not going to be one of the options that come with the discriminative models, at least in this method. However, the name is going to make sense if you are able to compare it to the other algorithms that fit into this model. You will find that this LDA algorithm is able to fit more in with the supervised machine learning method, and it will instead be there to work on the labels of your data. But then the PCA will be more of an unsupervised machine learning algorithm, and it will go through and, on purpose, ignore all of the labels that you have present there.

In many cases, you will be able to work with discriminative training, but as you do, you will be able to get an accuracy that is much higher compared to the conditional density functions. But, if you do work with those later models, handling the data that is missing in that set is going to be easier. So, as you can see, you have to give and take a bit here. You have to pick whether you want a better accuracy in your information, or if you are going to want to handle some of the missing points, and go from there.

Chapter 13: What Else Can I Do with Python Machine Learning?

This guidebook has taken some time to look through the different algorithms and models that you are able to use when you work with machine learning, as well as how you are able to work with the Python coding language at the same time. These are a nice mixture of supervised learning, machine learning, and more to help you see results in a short amount of time.

With that said, there are even more algorithms that you are able to use when it comes to Python machine learning. There are so many great algorithms that are going to come with this kind of technology and coding, and they can be used in a wide variety of situations based on the kind of project that you would like to work on here. Some of the different options that you can work with when it comes to

Python machine learning, in addition to the algorithms that we have already discussed in detail in this guidebook, include:

Naïve Bayes

The first method that we will talk about inside this chapter is known as the Naïve Bayes algorithm. In order to help us learn how this one will work, it is time to bring out a bit of imagination. To help us with this, we are going to pretend that we are in a project that has some challenges of classification. But at the same time, you want to make sure that you can come up with a new hypothesis that will work. You will also need to figure out a design that will allow you to add in new discussions and features based on how important each variable is in this model.

This may seem like it is a lot of work to do, and like we are trying to do a lot with just one type of algorithm, it is something that we can do with this option. Once you spend a bit of time collecting the information, you will probably end up with at least a few of the shareholders and other important investors who would like to see a model of what you plan to do and produce. And often they don't want to wait until you are all done with the work to see all of this.

As a programmer, you are going to run into a dilemma here. First, the information is not all done. And even with the information that is done, you may think that the information is too complex for someone who doesn't work in data analysis to understand, especially when it is in the beginning stage. How are you supposed to make things work for your shareholders so that they know what is going on—but make a model that isn't complete—and is easy enough to help them understand what is going on? This can be hard to do.

Often when it comes to data analysis, you are going to have thousands of points, and sometimes hundreds of thousands of points, that you are going to need to show

up on your model. There could even be some new variables that show up as you are working through the training and the testing. With all of this information going on—and being in the testing stage—how are you able to take this information and present it to the shareholders quickly and in a way that is easy for them to understand what is going on?

The good thing to remember here is that there is a nice algorithm, the Naïve Bayes, that will work and help you during this early stage of the model. It is designed to present all of the information that you need, while still keeping things simple enough for anyone, even someone who isn't a data scientist, to understand.

The Naïve Bayes model is meant to be easy for you to put together, and it is sometimes used to help you get through really large sets of data in a way that is simplified. One of the advantages of working with this model is that though it is simple, it is sometimes better to work with compared to the other, more sophisticated models that you can work with.

As you learn more about how to work with this algorithm, you will start to find that there are more and more reasons for you to work with it. This model is really easy to use, especially if you are a beginner to the world of deep learning and machine learning. You will also find that it can be really effective when it is time to make some predictions for our data sets and what class they should end up in. This makes it easier for you to keep things as simple as possible during the whole process. Even though the Naïve Bayes algorithm is really simple to work with, you will find that it is able to perform really well. In fact, when compared to some of the higher-class algorithms that are out there, and some of the ones that seem to be more sophisticated, this one is going to perform the best.

However, even though there are a lot of benefits that come with the Naïve Bayes method, you need to be careful because there are a few negatives that come with it. The first negative is that when you work with an algorithm that is set with categorical variables, you need to make sure that the data you are testing hasn't already gone through a data set for training. You may find that this algorithm is going to run into some issues when it comes to making accurate predictions, and often the data sets that it assigns information to will be based more on probability than anything else.

There are several different options that you are able to work with that are going to make this process a bit easier and will help to solve the issues that show up here, it is sometimes a bit confusing for a beginner who hasn't been able to explore and work with machine learning. Of course, there are a lot of things to enjoy about the Naïve Bayes algorithm, but there are some drawbacks that you need to be aware of.

Remember, the Naïve Bayes is going to be able to help you do a lot of things when it comes to your presentation, but it is pretty simple. It is used as a way to show the shareholders the information that you want to present to them, but it isn't going to be able to compare the information, and it isn't able to provide the predictions that you would like—or be able to get—with some of the other options.

Working with a Regression Algorithm

While working with the Naïve Bayes algorithm is a great option to go with, another algorithm that you may be interested in is going to be known as the regression analysis. This is a type of algorithm that is going to be the one you want to look at when you would like to see if there are some relationships showing between your predictor and dependent variables.

Programmers will see how this technique works well when they want to check out whether there is a casual relationship that shows up between the forecasting, with the different variables they want to work with, or even the time series modeling that you have in place. The point of working with this algorithm is that it will help you to grab all of the information found in your data set and then try to get it to fit onto a curve or a line, at least as much as possible. Of course, depending on the type of data points that you are working with, it may not work out the way that you will plan. But it is a good place to start, and it will help you to see if there are any commonalities between the points of data that you have.

Many times, a company will choose to work with this kind of algorithm because it helps them when it is time to make predictions. The company will then use these predictions in order to increase their overall profits. In some cases, you can use it to help you estimate how much the sales of a company are going to grow, while still being able to base it on the conditions of the economy right now.

One thing that a lot of data scientists like about this one is that they are able to add in pertinent information that they want to use. You can add in any information that is going to help you get started with this regression algorithm, including current and past economic information. This will be used inside the learning algorithm to help you figure out how the company is likely to grow in the future. To make this one work the way that you need, you have to make sure that the company inputs the right information to give them the right prediction.

One example of this is that a company is able to use this kind of machine learning algorithm to figure out if the company is not only growing, but also if it is growing at a similar rate, or faster or slower, compared to other companies in the same industry. This information can then be used to help you to make some predictions for how the company is going to do in the future, and they can then look for some options that will help them to do better if that is needed.

There are actually a few different algorithms that you can use that are considered regressions, and you have to know your data, and what you want to do with it, in order to help you figure out which one of those regression algorithms you would like to go with. While there are a lot of different options, the most common of the algorithms of regression that you are able to use with this include:

- Linear regression
- Polynomial regression
- Logistic regression
- Ridge regression
- Stepwise regression

As you can see, working with the regression algorithms are going to have a few different benefits that come with them. To start, you will see that these algorithms make it easy for anyone using the information to see what relationship is present, if any, between the dependent variables and the independent variables. This algorithm is also able to show what kind of impact will happen if you try to add in a new variable or change up another kind of variable that is in your data set.

Even though there are several benefits to this method, there are a few things to avoid when working on the regression algorithm. The biggest shortcoming that you will quickly notice is that you aren't able to use this algorithm to help out with any classification problem that comes up. The reason that classification problems and the regression algorithm don't always work together is that this particular algorithm tries to overfit the data many times. So, if you do try to add in different constraints here, you will find that the whole process is going to get tedious pretty quickly.

Clustering Algorithms

While we did stop and look at a few of the ideas behind clustering earlier in this guidebook, we also need to explore this a bit more and look at the different clustering options that a machine learning programmer has available, and even more about how these clustering options are going to benefit you when sorting through the data. These algorithms are going to fit in with unsupervised machine learning—you will be able to use them in a way that helps the program learn things all on its own.

Any time that you decide to work with a clustering algorithm, you need to make sure that the data is set up in a way that is simple and easy for the computer or the system to work with. This kind of method is going to take care of some of the data that you present to it, and then make sure that the clusters will come together right.

To work with this one, before you even decide to turn on the program, you will have a nice benefit of picking out how many clusters you would like to have present. You should aim to have at least two clusters, but often there is going to be more based on the amount of information that you have present for you. These clusters are then going to be able to hold onto the information that is found in your data set. The number of clusters will relate directly to how much data you are trying to sort through, and the kind of information that you would like to get when this algorithm is done.

For example, if your goal is to take a look at your data and sort out the information that you have for male customers and the data that you have for female customers, you may be fine with having just two clusters. But if you are interested in learning more about the different age groups of your customers, then you may want to divide it up between more than those two clusters. Once you have told the

algorithm how many clusters you would like to work with, it will take all of your data points and divide them up amongst the clusters that fit them the best.

The nice thing that a programmer will like about this kind of algorithm is that it will take on the responsibility of doing a lot of the work for you. This is because even though you have to figure out the clusters, the algorithm will go through and fit the data points into the clusters that you chose. To keep things organized here, we are going to call all of the main clusters that you picked out in the beginning (such as the cluster for male and the cluster for female) the cluster centroids.

So, any time that you are looking at one of the clusters, and you notice that there are quite a few points that fall inside of it, it is safe to assume that those data points are going to be similar in one way or another. There is going to be some attribute (and that attribute will vary based on the kind of cluster that you created and the data points that are inside) that show that they each are the same in some way.

Once you have been able to form up these original clusters, you are then able to take each of these and divide them up into some more sets of clusters if this works for your goals. This is often done to help dive deeper into the information, and to learn more about the customers, or the data, as possible. You can do this, dividing up the clusters as many times as you would like to gain the knowledge and insights that you are looking for here.

There are a few reasons that programmers and data scientists like to work with one of these clustering algorithms when they are doing some machine learning. The first reason is that the computation that you do thanks to the algorithms of clustering can be easy to use, are efficient when it comes to costs compared to some of the other machine learning algorithms, and you will enjoy the depth of knowledge that you are able to get with this. If you want to work on some

challenges that come with classification, then using one of the algorithms for clustering is going to be the easiest and the most efficient way to look at this information and to get the work done.

With all of the benefits that we have talked about, you do need to use a bit of caution with this one. These clustering algorithms are going to be useful, but you have to read through the information to make predictions about where to take your company next. This algorithm is not set up to make the predictions for you. And if you are not able to set up the centroids or the clusters in the proper manner, it is possible that you are not going to get the project to work the way that you would like, and you will make the wrong predictions.

Working with the Markov Algorithm

The next algorithm we are going to take a look at is going to fit into the category of unsupervised machine learning. This is known as the Markov algorithm. This is a good algorithm to work with because it is going to gather all of the chosen data that you place into it and then tries to translate it in a way that is able to work with any coding language that you choose. This may work with the Python coding language if that is your choice, but you can pick one of the other coding languages of your choice.

One thing that you will like with this algorithm is that the computer programmer is able to choose the rules that are going to go with it ahead of time so that the algorithm will take the direction that you wish it to. Many computer programmers are going to work with this algorithm because they are able to come up with the rules, rather than being forced to use the rules that are handed to them. You can use this benefit in order to take a whole string of data and make sure that it is useful to the job or the project that you are working about.

Another thing that you may like about this Markov algorithm is that you are able to work with it in several ways, rather than being stuck with just one method. One option to consider here is that this algorithm works well with things like DNA. For example, you could take the DNA sequence of someone, and then use this algorithm to translate the information that is inside that sequence into some numerical values. This can often make it easier for programmers, doctors, and scientists and more to know what information is present, and to make better predictions into the future. When you are working with programmer and computers, you will find that the numerical data is going to be much easier to sort through than other options of looking through DNA.

A good reason why you would need to use the Markov algorithm is that it is great at learning problems when you already know the input you want to use, but you are not sure about the parameters. This algorithm is going to be able to find insights that are inside of the information. In some cases, these insights are hidden, and this makes it hard for the other algorithms we have discussed to find them.

There are still some downfalls to working with the Markov algorithm. This one can sometimes be difficult to work with because you do need to manually go through and create a new rule any time that you want to bring in a new programming language. If you only want to work with one type of programming language on your project, then this is not going to be a big deal. But many times, your program will need to work with several different languages, and going in and making the new rules a bunch of times can get tedious.

What Is Q-Learning?

We haven't spent a lot of time taking a look at what is known as reinforced machine learning in this guidebook. But they can be incredibly useful when you are working with the ideas of machine learning in your projects as well. And the first type of reinforced learning that we are going to take a look at here is known as Q-learning.

With this algorithm, you will be able to use it along with a project that needs temporal difference learning. As you go through some of your learning types in machine learning, you may notice that this one is what programmers may call an off-policy kind of algorithm. It is known as this because it isn't able to go through and learn the action value function that you would get with other options.

This kind of algorithm is useful since a computer programmer is able to use it regardless of the kind of function they want to be able to create with their set of data, you have to take the time to go through this method and list out the specifications that you need. This is important to ensure that the learner or the user will pick out the right course of action that you would like. This does add in a few more steps to the model compared to some of the other methods that you use, which is a drawback to consider. But, because of the efficiency that comes with this model and how well it works, it is often worth the effort and the time to do these steps.

After you have been able to go through with this algorithm and find the action value function that will work the best for your data points, it is time for you to create what is going to be known as the optimal policy. How are we supposed to be able to construct this with the Q-learning algorithm? The best way to get started with this is to use the actions that you think will come in at the highest value, regardless of the state that has been chosen to do this one in.

Depending on the kind of data that you want to go through, and the results that you are hoping to get, there could be a number of great advantages that come with using the algorithm in machine learning for Q-learning. One of the benefits of this is that you won't have to go through all of the effort or the time that is needed to put in the models of the environment so that the system is able to compare the means. You will be able to compare a few, and often a lot, of actions together and compare how they are going to be together. In addition, you can use any kind of

environment that you would like to with this one, and still get the same results and be able to make the predictions that you would like.

Of course, there is a negative that can come with this one. The main issue is that you will need to go through a few more steps to make this kind of learning happen. This process, because you have to write out the rules that you want to use, and write out the course of action that makes sense for your goals, there are a few more steps compared to some of the other algorithms that we talked about in this guidebook. If you are in a hurry and don't care about having specific rules in place, then this is not that big of a deal.

As we have taken the time to explore in this guidebook, there are a lot of things that you are able to do when it comes to working with machine learning, especially when you are able to implement the Python coding language with it. We have looked at a lot of the different algorithms that you are able to bring up that utilize both, and even some of the Python codes that you are able to use along with each one to help you see the best results.

Whether you are working with supervised machine learning, unsupervised machine learning, or reinforcement machine learning, the Python code is able to help you get these various algorithms up and running. This ensures that you are able to make predictions, figure out the right course of action, create some great programs, and so much more—with the help of machine learning. Moreover, these final few algorithms discussed in this chapter will be able to help you get even more out of the whole process of machine learning.

Conclusion

Thank you for making it through to the end of *Python Machine Learning*! Let's hope it was informative and able to provide you with all of the tools you need to achieve your goals—whatever they may be.

The next step is to start using some of the different techniques and tools that we are able to work through in order to utilize Python Machine Learning. The field of machine learning is growing like crazy. Many computer programmers are interested in learning more about how this works, as well as how they are able to use it, in order to see the best results and to help them create new technology into the future. Think of it—a few years ago, there were no such things as the Alexa device (and other similar software from other companies) that could listen to your commands and respond—and that is all thanks to machine learning!

In this guidebook, we didn't just take a look at machine learning—we also looked at how the Python coding language can be used in order to make this all come to fruition for us. Python, as we explored in this guidebook, is a powerful yet easy programming language that we are able to work with—and when it is combined

together with machine learning, you are going to be able to get some of the best results possible.

This guidebook took some time to look at both the Python language and machine learning, as well as some of the different algorithms that you will be able to use with both of these. By the time you get to this point, you should be well-versed in some of the different algorithms that you are able to work with, as well as how they are going to benefit you based on the machine learning project you would like to work with!

When you are ready to learn more about how the Python coding language and machine learning work together and when you want to learn how to make Python machine learning work for your needs, then make sure to check out this guidebook to help you get started!

Finally, if you found this book useful in any way, a review on Amazon is always appreciated!