Python Machine Learning

A Step-by-Step Guide to Scikit-Learn and TensorFlow (Includes a Python Programming Crash Course)

Table of Contents

Introduction

Congratulations on buying *Python Machine Learning* and thank you for doing so.

The following chapters will discuss the things that you need to know to take machine learning and use it in your business or on your next project. And when you combine the different ideas that come with machine learning, and some of the different algorithms, with the Python coding language and the different libraries that come with it, you will find that it is possible to really get some of the complicated tasks done with ease.

This guidebook is going to start out with a good introduction to machine learning to help us understand what it is about, some of the options that you can do with machine learning as a beginner, the reasons to use or even learn about machine learning, and how machine learning and artificial intelligence are the same and how they are different. This gives us a general introduction to what this process is about and how we will be able to use it as we progress through this guidebook.

From there, we are going to explore a bit about the Python language with a crash course in coding in this language. For those who have never been able to learn Python, or who want to jump right into machine learning without all of the studying of a new language along the way, this part is the one for you. We are going to look at some topics like what Python is and how to download it on the different operating systems out there, how to write conditional statements, how to raise and manage your own exceptions, the OOP and functions, and even some of the other basic parts of a Python code.

In the third section of this guidebook, we are going to take a look at our very first Python library and what you can do with it when it comes to machine learning. We will explore a bit about Scikit-Learn and what this library can do, before diving into some of the supervised and unsupervised machine learning that you are able to do with this kind of library.

To finish out this guidebook, we are going to explore our second Python library of TensorFlow. There are some neat things that you can do with machine learning when it comes to TensorFlow that you are not able to do with any other coding library, so it is definitely a section you will want to check out. Inside we are going to explore the TensorFlow library, how to work with High level and

are going to explore the TensorFlow library, how to work with high-level and low-level APIs, and how to handle estimators.

There is so much that you can do when we talk about machine learning, and the Python coding language makes it that much easier for everyone to get started. when you are ready to learn more about Python machine learning and how to get started with some of your own projects today, make sure to check out this guidebook to help you out!

Part 1: An Introduction to Machine Learning

Chapter 1: What is Machine Learning

The first topic that we need to take a look at in this guidebook is machine learning. This is basically a process where you are trying to teach a computer or another machine how to use its own experiences with a particular user, and some of the things it has seen in the past, to help it perform even better in the future. There are a lot of examples of how this can work, such as voice recognition devices, and even with search engines.

As we go through this guidebook, you will find that there are a lot of different methods and algorithms that you can use with machine learning in order to get the machine to learn, but the one you choose really depends on the kind of results you want to get and the project that you decide to work with.

Machine learning is going to be a method of data analysis that is able to automate the process of building analytical models. It is also a branch of artificial intelligence that is going to be based on the whole idea that a system is able to learn from the data it is presented, it can identify the patterns that are there, and it is even able to make its own decisions without a lot of intervention from humans in the process.

Because of all the new computing technologies that are out there, machine learning, as we know it today, is not really the same as the machine learning that we see in the past. It was born out of a form recognition for patterns and the idea of how a computer is able to actually learn, without a programmer there, to ensure it performs a task specifically. Researchers who were interested in some of the things that we are able to do with artificial intelligence wanted to also see if their machines were able to learn from data that it was fed.

The iterative aspect that comes with this machine learning should be seen as an important programming tool because as we expose any of the models we create from this learning to new data, the model is then able to adapt on their own and independently. The machine is going to be able to learn what has happened to it in the past, and the examples it was given, in order to make accurate and reliable predictions in the future.

In recent years, there has been a resurgence in the amount of interest that is out there with machine learning thanks to a few different factors. In particular, some things like Bayesian analysis and data mining are growing in popularity as well, and in the process, machine learning is going to be used more now than ever

before.

All of these things mean that it is now easier and faster in order to automatically produce models with machine learning. And these models are now able to analyze bigger and more complex data, while also delivering results faster and results that are more accurate, even when this is done on a very large scale. And because all of this is able to come together and build models that are more precise, and organization is going to set itself up for identifying profitable opportunities better than before, while also avoiding more of those unknown risks ahead of time. This all comes together to help a company to become more competitive in the market.

There are a few things that need to come together in order to make sure that the system you use in machine learning is actually good. Some of these will include:

1. Ensemble modeling
2. Scalability
3. Iterative and automation processes
4. Algorithms, a good combination of basic and advanced ones
5. Data preparation capabilities.

The neat thing about working with machine learning is that almost every industry is able to use it. And it is still relatively new when it comes to the world of technology, so even the amazing things that have been done with it so far is just the beginning, and it is believed that this kind of technology is going to be able to do even more things in the process.

Machine learning is likely to grow quite a bit as time goes on. Right now, a lot of companies are using it in order to figure out what the data they are receiving is telling them, to figure out how they are able to make better business decisions over time, rather than having to make the decisions on their own, and to find some of the patterns that are hidden in the data, and that a human would not be able to go through.

But this is just the start of what we are able to do when it comes to machine learning. There are a ton of other applications, and what we are able to do with this right now is just the beginning. As more people and developers start to work with machine learning and start to add in some of the Python languages with it, it is likely that more and more applications are going to be available as well.

Most of the industries that are out there that are already working with [illegible]

Most of the industries that are out there that re already working with large amounts of data are going to be able to recognize the kind of value that they would get with using the technology that comes with machine learning. By being able to actually get through this data and glean some good insights from it, and being able to do this close to real-time, the company is then able to work in a more efficient manner in order to gain a big advantage of others in their same industry.

And this is the beauty of working with machine learning. We are able to do things that may have seemed impossible in the past are possible now with the help of machine learning. Businesses that are handling more data than ever before are finding the value of working with machine learning to help them get their work done. They can get through this information faster than would be possible with a person looking through it on their own and can give them that competitive edge over others.

There are a lot of different companies that will be able to benefit from a program that can run on machine learning. Some of the different industries that are already using this kind of technology will include financial services, government, health care, retail, oil and gas, transportation, and more.

Machine learning is similar to artificial intelligence that is going to allow a computer to learn, similar to what we are seeing with the human mind as well. With a minimal amount of supervision from a person, the machine will be able to automate a lot of tasks, find the information that you want, and get to some insights and predictions that you may not be able to find in other methods on your own. And this guidebook is going to spend some time looking at how you are able to do this type of machine learning with the help of the Python coding language so you can start some of your own projects in no time.

Chapter 2: The Different Types of Machine Learning

Now that we have had a chance to learn a bit about machine learning and how it can work well for your needs, it is time to take a step back and focus on some of the different ways that you can work with machine learning. There is more than just one algorithm for machine learning out there that you are able to choose based on what kind of project you choose to work with. But these algorithms can be sorted out into three main categories to help us understand how they work a bit better, and when we are likely to use them for our needs.

The three main types of machine learning that you are able to use include supervised machine learning, unsupervised machine learning, and reinforcement machine learning. Each of these will work in a slightly different way in order to make sure that the computer or another machine knows how to learn and collect the information that is needed. So, let's take some time to explore the different types of machine learning and how they work!

Supervised

The first type of machine learning that we are going to explore is the idea of supervised machine learning. This type of learning is going to happen when you are able to choose an algorithm that is going to learn the response that is the correct one based on the input that the user gives to it.

There are several ways that supervised machine learning can do this. It can look at examples and other targeted responses that you provide to the computer. You could include values or strings of labels to help the program learn the right way to behave.

This is a simple process to work with, but an example to look at is when a teacher is teaching their students a new topic, and they will show the class examples of the situation. The students would then learn how to memorize these examples because the examples will provide general rules about the topic. Then, when they see these examples, or things that are similar, they know how to respond. However, if an example is shown that isn't similar to what the class was shown, then they know how to respond as well.

As you go through some of the work that comes with machine learning, you will run into a variety of algorithms that fit under the umbrella of supervised machine learning. Some of the most common types that you will use though are known as

random forests, decision trees, regression algorithms, and KNN.

Unsupervised Machine Learning

Once you are done taking a look at some of the supervised machine learning that you are able to do, you may notice that there are a few times when this is not going to work for your needs. Supervised machine learning works in some cases, but it is not going to work as well for some of the other problems that come into play. This is where you will be able to look at unsupervised machine learning and see where this is able to fill in some of the blanks.

Unsupervised learning is the type that will happen when your algorithm is able to learn either from mistakes or examples without having an associated response that goes with it. What this means is that with these algorithms, they will be in charge of figuring out and analyzing the data patterns based on the input that you give it.

Now, there will also be a few different types of algorithms that can work well with unsupervised machine learning. Whichever algorithm you choose to go with, it is able to take that data and restructure it so that all the data will fall into classes. This makes it much easier for you to look over that information later. Unsupervised machine learning is often the one that you will use because it can set up the computer to do most of the work without requiring a human being there and writing out all the instructions for the computer.

A good example of this is if your company wants to read through a ton of data in order to make predictions about that information. It can also be used in most search engines to give accurate results.

Unsupervised machine learning is used in a lot of the different programs and projects that you want to use that come with machine learning because of all the power and more that is behind it. Some of the techniques that you can enjoy with unsupervised machine learning, and that you are most likely to use with this kind of learning include neural networks, clustering algorithms, and the Markov algorithm.

Reinforcement Machine Learning

And the third type of machine learning that we need to focus on is known as reinforcement machine learning. This one is going to work in a manner that seems similar to an unsupervised machine, but instead, it focuses on the idea of

true and false to help it to learn how to behave. This one works a little differently than the other two options, but this is going to make it perfect for some of the projects that you want to explore.

So, whenever you decide to work with reinforcement machine learning, you are working with an option that is like trial and error. Think about when you are working with a younger child. When they do some action that you don't approve of, you will start by telling them to stop, or you may put them in time out or do some other action to let them know that what they did is not fine. But, if that same child does something that you see as good, you will praise them and give them a ton of positive reinforcement. Through these steps, the child is learning what is acceptable behavior and what isn't.

To keep it simple, this is what reinforcement machine learning is going to be like. It works on the idea of trial and error, and it requires that the application uses an algorithm that helps it to make decisions. It is a good one to go with any time that you are working with an algorithm that should make these decisions without any mistakes and with a good outcome. Of course, it is going to take some time for your program to learn what it should do. But you can add this to the specific code that you are writing so that your computer program learns how you want it to behave.

The different algorithms that you will use with reinforcement learning are not going to be as prevalent as with the other types of learning, and there are not as many of them as we talked about above. But you can still use a few different algorithms that fit under the umbrella of reinforcement learning, including SARSA and Q-learning.

These are the three main types of machine learning that you are able to work with. The idea behind them is that you will be able to do a lot of different projects based on what your end result should be. Learning the different algorithms and working with them, like we will as we progress through this guidebook, can make it easier to get some of the results that you want.

Chapter 3: How Does Machine Learning Compare to AI

One thing that we need to spend some time working on and understanding before we move on is the difference between Artificial Intelligence and Machine learning. Machine learning is going to do a lot of different tasks when we look at the field of data science, and it also fits into the category of artificial intelligence at the same time. But we have to understand that data science is a pretty broad term, and there are going to be many concepts that will fit into it. One of these concepts that fit under the umbrella of data science is machine learning, but we will also see other terms that include big data, data mining, and artificial intelligence. Data science is a newer field that is growing more as people find more uses for computers and use these more often.

Another thing that you can focus on when you bring out data science is the field of statistics, and it is going to be put together often in machine learning. You can work with the focus on classical statistics, even when you are at the higher levels, sot that the data set will always stay consistent throughout the whole thing. Of course, the different methods that you use to make this happen will depend on the type of data that is put into this and how complex the information that you are using gets as well.

This brings up the question here about the differences that show up between machine learning and artificial intelligence and why they are not the same thing. There are a lot of similarities that come with these two options, but the major differences are what sets them apart, and any programmer who wants to work with machine learning has to understand some of the differences that show up. Let's take some time here to explore the different parts of artificial intelligence and machine learning so we can see how these are the same and how they are different.

What is artificial intelligence?

The first thing we are going to take a look at is artificial intelligence or AI. This is a term that was first brought about by a computer scientist named John McCarthy in the 1950s. AI was first described as a method that you would use for manufactured devices to learn how to copy the capabilities of humans in regard to mental tasks.

However, the term has changed a bit in modern times, but you will find that the

However, the term has changed a bit in modern times, but you will find that the basic idea is the same. When you implement AI, you are enabling machines, such as computers, to operate and think just like the human brain can. This is a benefit that means that these AI devices are going to be more efficient at completing some tasks than the human brain.

At first glance, this may seem like AI is the same as machine learning, but they are not exactly the same. Some people who don't understand how these two terms work can think that they are the same, but the way that you use them in programming is going to make a big difference.

How is machine learning different?

Now that we have an idea of what artificial intelligence is all about, it is time to take a look at machine learning and how this is the same as artificial intelligence, and how this is different. When we look at machine learning, we are going to see that this is actually a bit newer than a few of the other options that come with data science as it is only about 20 years old. Even though it has been around for a few decades so far, it has been in the past few years that our technology and the machines that we have are finally able to catch up to this and machine learning is being used more.

Machine learning is unique because it is a part of data science that is able to focus just on having the program learn from the input, as well as the data that the user gives to it. This is useful because the algorithm will be able to take that information and make some good predictions about the future. Let's look at an example of using a search engine. For this to work, you would just need to put in a term to a search query, and then the search engine would be able to look through the information that is there to see what matches up with that and returns some results.

The first few times that you do these search queries, it is likely that the results will have something of interest, but you may have to go down the page a bit in order to find the information that you want. But as you keep doing this, the computer will take that information and learn from it in order to provide you with choices that are better in the future. The first times, you may click on like the sixth result, but over time, you may click on the first or second result because the computer has learned what you find valuable.

With traditional programming, this is not something that your computer can do on its own. Each person is going to do searches differently, and there are

millions of pages to sort through. Plus, each person who is doing their searches online will have their own preferences for what they want to show up. Conventional programming is going to run into issues when you try to do this kind of task because there are just too many variables. Machine learning has the capabilities to make it happen though.

Of course, this is just one example of how you are able to use machine learning. In fact, machine learning can help you do some of these complex problems that you want the computer to solve. Sometimes, you can solve these issues with the human brain, but you will often find that machine learning is more efficient and faster than what the human brain can do.

Of course, it is possible to have someone manually go through and do this for you as well, but you can imagine that this would take too much time and be an enormous undertaking. There is too much information, they may have no idea where to even get started when it comes to sorting through it, the information can confuse them, and by the time they get through it all, too much time has passed and the information, as well as the predictions that come out of it, are no longer relevant to the company at all.

Machine learning changes the game because it can keep up. The algorithms that you are able to use with it are able to handle all of the work while getting the results back that you need, in almost real-time. This is one of the big reasons that businesses find that it is one of the best options to go with to help them make good and sound decisions, to help them predict the future, and it is a welcome addition to their business model.

Part 2: Your Python Crash Course

Chapter 4: What is Python and How to Set It Up On Your Computer?

If you are looking to learn a new coding language, then look no further than the Python coding language. This is considered one of the most popular options for coding out there mainly because you are able to use it on almost any platform, and it is pretty easy for a beginner to learn how to use while adding in a ton of power to do the different programming options that you need.

The nice thing about Python is that even with all of the power that comes with it, you will see that it has been designed with beginners in mind. So, if you have not been able to work with any kind of coding in the past, you will still be able to learn how to work with Python and even do some of the machine learning algorithms that we will talk about later on. Python is also known as open-source, which means you can download the coding language, along with all of the other parts that are needed without having to pay for them. Add in that there is still a dedicated group of developers who update and work to improve the program and you have one of the best programming languages out there to work with.

One thing that we have to remember when it comes with Python is that even though it is free and you are able to get started with it easily, there are a few extensions and libraries that you can add to this language that is going to cost a bit. These still work well with this Python language, but because they are developed by a third-party developer, they will cost a bit. But you get to choose whether you want to use those or not and it is perfectly fine to just work with the basic and open-sourced parts of Python.

As a beginner, you are going to find that working with the large library that comes with Python can make your life so much easier. You are going to enjoy that it is easy to start with and that there are a lot of functions and more found in the library. This helps you to do more with your codes, helps to keep things organized, and more. The basic library will do a lot of different things that you can do with the simple Python library, but for some of the technical things that we will do with machine learning as we move through this guidebook, you will need to download a few other libraries to go here. The two main ones we will look at include TensorFlow and Scikit-Learn, but there are other options that work well for helping you increase the capabilities of Python.

This coding language is also all about the classes and the objects. We will talk

about this a bit more as we progress through this book, but this really makes coding easier for you. It ensures that when you call up a part of the code, as long as you name it and call it up in the right away, it is going to show up the way that you want. This may have been a struggle for some beginners in other coding languages, but this problem is solved with the help of the Python code.

Python can work with other languages. Not only are you able to turn on the Python language and use all of the capabilities that come with it on your computer, but you can also combine it together with some other coding languages to really enhance some of the capabilities that you see on there. Python is able to do a lot of different things, but there are a few points where it may fall a bit short, or that other coding languages are going to do better. Adding it together with one of these other coding languages can ensure that your program is written the way that you want.

Python is already being used in a lot of different programs already. In fact, some of your favorite programs may already be using Python to help them run. You will find that Python is on many website and games and other common programs, and you may not have even noticed to begin. As we go through this guidebook, you may be pleasantly surprised at how great these programs work, even though a lot of the codes are simple to read and write.

Before we go any further with Python, we need to take some time to learn how to set up the Python program on the different operating systems that you are going to use. Python is going to work with any operating system that you want to work with including Windows, Mac OS X, and Linux so you will be able to download it and get it to work based on whichever is your preference.

To start, you are able to download the Python program from a few different sources. But the method that is the easiest is going to be www.python.org . This one is set up to have all of the files that are already needed to get the code to work right away after the download and will make sure that you get all of the files, the interpreter, the IDLE, and the compiler that is needed. You can also choose to download from another location if that works the best for you, but you should check to see which files are included and if you need to go through and download some more files.

So, let’s go through and look at some of the different steps that you need to take in order to download Python onto the various operating systems that you want to use. First, we are going to look at how to get Python set up on a Windows operating system. This is a popular operating system that programmers are going

operating system. This is a popular operating system that programmers are going to work on, but since Windows has its own coding language available, you will have to take the manual method in order to install Python on the system.

The good news is that this really only takes a few steps, and it is pretty easy to work with. It won't take long before you are able to get the Python program on your computer and you can see it working in no time. Once the Python program and all of the files are put on it are set up on a Windows operating system, there won't be any problems that you need to worry about. It is not going to interfere with anything on the system, and the Windows coding language isn't going to cause problems either.

Once you are ready to install the Python language so it is ready to work on your computer, and with the Windows operating system, you will first need to make sure that the right environment and variables are in place to ensure that you are able to run the scripts for Python from the command prompt that is there. The other steps that are needed to get the Python language, and all of its files, set up on a Windows operating system includes:

1. The first step that we need to do here is to head on over to the download page for Python and grab the installer that is listed under the Windows operating system. You are able to pick out the version of Python that is the best for you, but many programmers choose to download the version that is newest at the time. You also need to decide if you want the 32-bit or 64-bit version of Python based on the operating system type that you are working with.
2. Once you have been able to grab the installer from Windows for Python, it is time for you to click on it so that you can do the Run as Administrator. As you go through and do this, the system is going to provide you with two options to pick out from, and you can pick the one that works for you. For this, click on "Customize Installation."
3. The next screen that comes up is going to have a lot of boxes to check on. You need to make sure that you select all of the ones that fall under Optional Features and then click to go to the next page.
4. While you are still here, you can also check out the location where you want to install this Python. Once that folder is then picked out, and you can click to install. This is going to take a bit of time to get the installation so have some patience with it. Once that install is all done, you can then close out of this part.
5. The next thing that we need to do is set up the PATH variable that

works with this system so that you have all of the directories that will include packages and other components that are necessary to use later on. The way that you get all of this set up is going to use the steps below;

a. Open up your Control Panel. If you are not certain where this is, click on your taskbar and type in “Control Panel”. Click on the little icon that shows up when you do this.
b. When you get the Control Panel to show up, you can search for “Environment” and then click on Edit the System Environment Variables. When this is done, you can then click on the button labeled “Environment Variables.”
c. At this point, you can go to the section that is listed for User Variables. Here you can either decide to create a new PATH variable, or you can edit the PATH variable that is already in place.
d. If there isn’t a variable for PATH on the system as you are looking, then it is time for you to create your own. To do this, click on New. Give it a name, one that works for the PATH variable you are choosing, and then place it into the chosen directory. Click to closer yourself from the Control Panel at this time and then go to the next step.

6. When you get to this point, you can open up that Command Prompt again. You can do this by clicking on your Start Menu, then clicking on Windows System, and finally on Command Prompt. Type in the word “python”. This will be enough to load up the interpreter of Python for you.

Once the steps above are done, you can then go back to our system and open up the Python language. You will then be able to use it in any manner and work with some of the codings that we will do in this guidebook. It takes a few minutes to go through the steps above, but you will find that this is one of the best ways to get it set up and it only takes a few minutes to get it all done.

The next thing we need to look at is how to download the Python files on a Linux operating system. This one is also going to work well with Python, and since there are a lot of people who are using this operating system, it is a good way to learn how to code in Python as well.

Now, the first step that we need to take here is to see which version of Python 3

Now, the first step that we need to take here is to see which version of Python 3 is available on our system. To do this, you can just open up a command prompt on Linux and then go with the code below:

$ python3 - - version

If you are on using a version of Ubuntu that is a bit newer, then it is a simple process to install Python 3.6. you just need to use the commands below:

$ sudo apt-get update
$ sudo apt-get install Python3.6

If you are relying on an older version of Ubuntu or another version, then you may want to work with the deadsnakes PPA, or another tool, to help you download the Python 3.6 version. The code that you need to do this includes:

$ sudo apt-get install software-properties-common
$ sudo add-apt repository ppa:deadsnakes/ppa
# suoda apt-get update
$ sudo apt-get install python3.6

The nice thing about working with this one is that if you do choose to work with the variety of distributions that come with Linux, you can also download the Python 3 program on it to help you out. You can go through these seems steps no matter which distribution you choose to work with. You can also stick with the steps above to install any version of Python onto your system that you would like, so if you want to go with an older version, such as Python 2, that is easy to work with as well.

Now, we can move on here and look at how to get this language into an Apple computer and on Mac OS X. This system is set up to work well with Python, and it is going to already have Python 2 programmed on it. You can go through and double-check to see if this version is present on your system or not. You will find that Python 2 is going to work just fine for a lot of the programming that you want to do with Python so if you want to make things easier, you can go through and just work with this.

However, it is common that a lot of programmers want to work with Python 3, or one of the newer versions that come with Python, and they want to update this. This is pretty easy to work with. The first step to take here is to uninstall the Python 2 version so that you won't end up with some problems with two

versions of this coding language on your computer. Then you can go to www.python.org in order to pick out the exact version of Python that you want to be able to add to the computer.

Being able to run both the shell and the IDLE with the Python language is going to depend on which version of the program you decide to work with, as well as what preferences are there when you write out the code. The two biggest commands that you are going to use the most often to help make sure that the shell and IDLE applications start-up when you want will vary based on the version you use, and they are:

- For Python 2.X just type in “Idle”
- For Python 3.X, just type in “idle3”

As we talked about a bit before, when you take the time to download and install this Python 3 on the Mac operating system, you will need to install the IDLE so make sure that is there, and you can install it as a standard application inside of your Applications folder of course. To help you to start up this program using your desktop, you just need to go into the folder, double click on the application for the IDLE, and then you can wait for it to download.

And that is as simple as it is! If you are able to follow these steps, you will be able to get the Python code on your system, and it is going to be ready to work for you and write some of the codes that we will discuss in this section, as well as in some of the other sections as we move into machine learning as well.

Chapter 5: Some of the Basic Parts of Your Code

With the download of the Python language done and taken care of, it is time to move on to some of the basic parts that come with writing code in Python, along with some of the different benefits that come with the different parts so we have a better idea of how amazing working with Python can be for us. We will start out this section looking at some of the parts that you are most likely to see when you work on a Python code and can make it easier to move on to some of the more complicated things that we do in the following chapters, especially when it comes to machine learning.

You will find that when it comes to using the Python code, there are a lot of different things that you are able to do. And the work that you decide to put into your code often is only limited by the kind of program that you would like to write out. making sure that you have some of the basics down, and gaining a good understanding of how this all works can help out later when you work on machine learning and some of the more complicated codes that you wish to do later on.

The keywords

The first part of the Python code that we need to pay attention to is the keywords. Any coding language that you choose to go with is going to have these keywords, and they are seen as important and reserved because they help tell the compiler the right actions to take to complete the code. These are special because they are a command for the compiler to follow. If you place them in the wrong part of the code or use them in the wrong way, then the code is not going to provide you with the results that you would like.

As you are taking a look at some of the keywords that come with Python, it is important that you learn how to use them in the proper manner. You do not want to make a mistake of adding them to the wrong part of the code. Doing this can lead to a lot of error messages that you have to then try to sort through. As you start working with the code a bit more, you will start to see what we mean by keywords a bit more and where you can use them to get the compiler to act in the manner that you want.

How to name an identifier

The next topic that is important to look at when working in the Python language

is how to name your identifiers. If you want these to work well and the program to behave, then you need to make sure that the proper naming method is used each time. Using the naming process the wrong way can end up with frustrations and a lot of rewriting of the codes. So, with this in mind, let's take a look at the proper steps that you have to take in order to get the identifiers named in the right way.

There are actually quite a few different types of identifiers that you can find in the Python language, but they are going to come in different names. You may find them called things like classes, functions, entities, and variables. Any time that you name one of these identifiers, even when they are under different names, the rules are going to be the same so you won't have to change up how you do it each time. this can make life a bit easier.

This brings us to the idea of naming the identifiers and learning which rules you have to follow to make this happen. First, you need to take some caution concerning the name that you give to the identifier. There are a ton of names that are available, and you can choose, for the most part, the name that you want. You get the choice of working with letters, both the uppercase and the lower case, and any number. The underscore symbol and any combination of the previous will work as well.

But there are a few restrictions to keep in mind with this as well when you start naming your identifier. First, it is not allowed for you to name any identifier with a number, and the name should not have any spaces that come with it. Naming the identifier something like 5kids or 5 kids would get you an error, but naming it fivekids or five_kids would be just fine. And keep in mind that you should never use a keyword as the name of one of your identifiers or the compiler is going to get confused.

When you come up with the name that you want to give to that identifier, make sure that you remember what it is. It may follow all of the rules that you need, but if you are not able to remember the name when it is time to execute the code or pull out that identifier later on, then there can be some issues. If you call it the wrong thing or you spell it differently, then there could be an error or the compiler is going to get confused.

With the rules above in mind, if you are able to pick out a name to go with your identifier, and you make sure that it actually fits in with the work that you are doing for that part of the code, and you follow the few rules that are above, then naming them will be easy and you won't run into any problems with it.

naming them will be easy and you won't run into any problems with it.

The Statements in Python

Another topic that we need to focus on for a moment is the statements that come in the Python language. Statements are simple because they are just some sentences that we are able to tell the compiler to add to the screen for others to read. They are simply a string of code that you are first able to write out, and then the compiler will take it and list it out on the screen based on what kind of code that you have.

When you are working with the various statements that are available on Python, as long as they are written out in full sentences and in the right spot of the code, the compiler will have no problem reading through them, and the message you would like will show up on the screen. You can choose to have the statements at any size that you would like as long as it makes sense for the code that you are writing out.

Comments

We can't finish going through the basics of a Python code without looking at what the comments are all about. You can definitely write out any code that you want without having any comments, but these are useful for helping us to understand how to work with different parts of the code and can explain what is going on in a certain part of the code, and even to leave a little message to someone who is looking it over without interrupting the code at all.

These comments are going to be really helpful in many cases because you can add them in to make it easier for a programmer or someone else reading through the code to have a better idea of what is going on in that code. But it is not going to make any changes in how the code performs. The comments are going to keep things organized, such as naming a certain part of the code so that it works more efficiently, can help explain what is going on in one part of the code compared to another, and can ensure that everyone is on the same page, without causing the code to pause or have errors.

Creating one of these comments can be pretty simple in the Python code. And you can add in as many of these as you would like. It is often recommended that you keep these to a minimum so that they don't mess up the code or make it look too convoluted. In order to make up some of your own comments in Python because you just need to work with the # sign in front of the comment that you

want to write out. The comment can be as long as you want, and you can add in as many as you would like as long as the # sign is in front of it, so the compiler knows not to use that part.

Bringing out the variables

It is also important to spend some time looking at the different variables that are present in the code, and see how they work to add something important to the code as well. These are going to be more common in the code than a lot of beginners may think and the main reason that we need to focus on these variables is that they can help to store up some of the different values that you try to place in the code. This is one of the best ways to make sure any line of code that you try to write is going to be easy to read, will stay organized, and will execute in any manner that you would like.

One of the things that you are going to like the best with these variables is that even with all of the work that they provide, they are still going to be easy to work with. All that you need to do to ensure that a value is assigned correctly to a variable is to put in the equal sign right in between the variable and the value. With that sign in place, the compiler will take on the rest of the work for you. You can choose to add in any kind of variable that you want in here, just double check that you have the equal sign in place first.

Another option to work with is to assign more than one value to the same variable at a time. If you just make sure that there is an equal sign that is linking both of these values back to the variable, then the compiler is going to know exactly what it should do. There are a lot of examples of how this can work in a Python code, and it is simple to do, so just make sure that the right value is hooked up with the right variable, and you should be ready to go.

The Operators

And the last thing that we are going to take a look at in this chapter is the idea of the operators. These are another small, and often easy, part of the code that can make a big difference in how the code runs. There are many types of operators that you are able to work with including those that will assign names to the identifiers you are using, ones that can help with simple mathematics, some that compare two statements to see if they are the same or different, and more. These operators help the compiler know exactly what you would like it to do depending on the part of code you are at.

As you take a look through some of the different codes that we will work on in this guidebook, a lot of different operators are going to show up. And often you will use them without even realizing what you are doing at the time or realizing that you are working with the operators. But it would be almost impossible to do any kind of coding if you were not able to add in the operators along with some of the other parts.

These are just a few of the basic parts that you are able to work on when it comes to creating your own codes in Python. These may seem simple, and you may wonder why you would need to use these in the first place, but there are so many times that you are going to see these basics show up in the code that you are trying to write, and having a strong working knowledge of them can make coding in Python so much easier.

Chapter 6: How to Write Your Own Conditional Statements

Now that we have some idea of the basics that come in a Python code, it is time to learn a few of the different tricks that you are able to do with writing your own codes as well. And the first place we are going to start is with the conditional statements. As you are writing codes, you may wish that you could, at times, set up a program and get it to behave in the manner that you would like all of the time. it would be nice to, ahead of time, guess each and every answer that the user is going to provide to the program, but of course, we know that this is impossible.

Let's say that you are working on a code where you want the program to ask the user what their favorite color is. It would take forever, and be a waste of time, to go through and write out each and every color that is available throughout the world, and it is highly likely that you would still miss some. The code would be a mess, and you would probably cut your losses and never want to code again. This is where the conditional statements can come into the game.

When you want to be able to write out some code that has the capability to make some decisions for you without you being there, based on certain conditions that you are able to set up ahead of time, then this is when you bring out the decision control statements or the conditional statements. These can be helpful any time that you are allowing the user to put in an answer on their own, rather than having a menu of listing available for them. This helps the program know what steps it needs to take based on the conditions that are set, and the answers that the user provides.

Before we get too far into this, we have to look at the fact that there are three types of conditional statements that you are able to work with. The three conditional statements that we can use in Python include the elif statement, then the if else statement, and also the most basic kind known as the if statement. Let's take some time to explore each of these and see how they work in helping to make your code perform in the proper manner.

The first conditional statement that we are going to look at is the if statements. This is a simple example of a conditional statement, and many programmers want to focus more on the if else statements. But learning this one can provide us with some of the foundations that come with these kinds of statements so we can

use these properly.

When we are using the if statement, we will find that it relies on the idea that the answer we are given from the user is either going to be true or false based on the conditions that you set ahead of time. if the answer from the user does match up with your chosen conditions, it is true. If it doesn't, then it is false. Of course, this isn't to say that the user is always giving the wrong answer when it is false; it just means that they are not giving the answer that meets with the conditions. If the user does add in an answer that is seen as true and the computer sees this, then the next step is that they are going to get the information that you were able to add to the code. But if the answer is seen by the compiler and it is seen as false, then the program will just end because it really has no idea of what it is supposed to do next.

To get a better idea of how the if statement is going to look and how it is able to work inside of your code, take a look at an example of the if statement that we have below:

```
age = int(input("Enter your age:"))
if (age <=18):
        print("You are not eligible for voting, try next election!")
print("Program ends")
```

Let's explore what is going to happen with this code when you put it into your program. If the user comes to the program and puts that they are younger than 18, then there will be a message that shows up on the screen. In this case, the message is going to say "You are not eligible for voting, try next election!" Then the program, as it is, is going to end. But what will happen to this code if the user puts in some age that is 18 or above?

As we work with the if statements, we will see that when our user puts in an age that is higher than 18, then nothing is going to happen. The if statement is not going to have anything there to catch the answers that are above 18 in this example. The user has to put in that their age is younger than 18 for the if statement to work so at this point, the program is stalled.

You can see where this is going to end up causing some problems along the way. It is likely that you are going to set up a program where you want the user to put in any information that they would like to this question, with the program proceeding no matter which answer is presented. The if statement is not able to do this, but the if else statement, and even the elif statement that we will talk

do this, but the if else statement, and even the elif statement that we will talk about next, can be important to making this happen.

First, we have to explore what the if else statement is able to do for us. These are going to follow the idea as we found while working with the if statement, but it can take things a bit further by solving the problems that come with the if statement on its own. This one ensures that you are able to give the user response, no matter what age group they add to the program.

Going with the same kind of example that we used before, you are going to use the if else statement in order to allow the user of your program to add in any age that they want, while still making sure that they get a result with that age as well. This one allows you to put in response to your user no matter what their age is. an example of how you can use the if else statement in the correct manner with Python includes:

```
age = int(input("Enter your age:"))
if (age <=18):
        print("You are not eligible for voting, try next election!")
else
        print("Congratulations! You are eligible to vote. Check out your local polling station to find out more information!)
print("Program ends")
```

As you can see, this really helps to add some more options to your code and will ensure that you get an answer no matter what results the user gives to you. You can also change up the message to say anything that you want, but the same idea will be used no matter the answer that the user gives.

You have the option to add in some more possibilities to this. You are not limited to just two options as we have above. If this works for your program, that is just fine to use. But if you need to use more than these two options, you can expand out this as well. For example, take the option above and expand it to have several different age groups. Maybe you want to have different options come for those who are under 18, those that are between the ages of 18 and 30, and those who are over the age of 30. You can separate it out in that way, and when the program gets the answer from the user, it will execute the part that you want.

Of course, this example is not meant to limit you, and there are a lot of different ways that you are able to use the if else statement to help you get the most out of

the programs that you are writing. For example, you could create a part of a program where you would allow the user to choose their favorite color. You could spend a bit of time and list out six colors that you think are the most commonly picked, and add some responses to this. The user is still going to have the option of picking one of those six colors you listed, or they can pick another color that is not on the list.

Now, if your user does choose their favorite color as something that is on your list, then they will get whatever result you put with it. However, if they pick out a color that is not on your list which is possible with all of the color options, then the else part of this statement, which is the catchall here, is going to show up for them.

This else part of the code is super important when it comes to this kind of statement, so you have to make sure that this is added to the code. In many cases, when you make the conditional statement, it is almost impossible to think up each and every answer that someone is going to put into the code, and this would just be a waste of time to try. The else part of this code helps to catch up with any of the answers that the user puts in that you didn't include, and ensures that everyone still gets a response.

With this in mind, it is also time to take a look at the third type of conditional statement known as the elif statement. This one is going to use some of the ideas that have been brought up in the other two conditional statements, but it uses them in a slightly different way. You can recognize this one more like a menu option for people to choose which option they want from it, and you can make the menu as long or as short as you would like.

There are a ton of different codes in Python that you are able to work with that can utilize the elif statement, and it often depends on what you are trying to write out. The most common way that you would see this kind of statement is when you are working on a game when there is a menu of choices that show up on the screen for you to choose from. These elif statements can be used at any time that your program wants to provide more than one or two options for the user to pick from.

To better understand how these elif statements are going to work, here is a good example of the syntax that comes with these statements:

if expression1:

statement(s)
elif expression2:
statement(s)
elif expression3:
statement(s)
else:
statement(s)

The above is going to just contain the syntax that you will need to use when you want to create your own elif statement in your code. You are able to add to this or take away based on how many options you would like to provide to your user. Just take the syntax then, and add in the information that should go with each part.

To help us to see how we would actually use the elif statement, rather than just the syntax that goes with it, we need to look at an example. The example below is a good one to see how the elif statement is going to work so take a look over it, and add it to your compiler to get some practice.

```
Print("Let's enjoy a Pizza! Ok, let's go inside Pizzahut!")
print("Waiter, Please select Pizza of your choice from the menu")
pizzachoice = int(input("Please enter your choice of Pizza:"))
if pizzachoice == 1:
        print('I want to enjoy a pizza napoletana')
elif pizzachoice == 2:
        print('I want to enjoy a pizza rustica')
elif pizzachoice == 3:
        print('I want to enjoy a pizza capricciosa')
else:
        print("Sorry, I do not want any of the listed pizza's, please bring a Coca Cola for me.")
```

With this example, the user can get to this part of the code and will see the different choices they can use to come up. They can choose which type of pizza they want, or if they just want a drink, and then click or type in that number. We just put in a few options, but you can add in as many or as few options as you want when creating an elif statement.

The conditional statements are going to be really useful in some of the codes that you decide to write with Python. Knowing how to make these work, and studying the way that they can add to the different codes that you create can be

studying the way that they can add to the different codes that you create can be super important to ensure you get the results that you want. Take some time to practice these conditional statements, so you are ready to use them when working on Python machine learning.

Chapter 7: OOP and Functions

One of the things that makes Python easy to use for beginners is that it is considered an OOP, or object-oriented programming, language. This is going to make coding a bit easier for us because it shows how all of the different parts of the cod we try to write will be similar to a regular object, just like we can find with all of the objects that are around us in real life. There are no abstract ideas in here, which is going to ensure that we are better able to find and use what we want without things getting lost.

There can be a lot of different objects inside of your code, and you want to make sure that they are organized in a manner that makes them easy to find, without parts getting lost along the way. The way that Python handles this is to allow programmers to build classes. One of the best ways to think about these classes is as boxes that are able to hold onto the various objects that are found inside of your code.

The classes can be for pretty much anything that you want, and they can hold as many or as few objects as you would like. The trick is that when someone takes a look into one of the classes that you create, they should understand how the objects in that class go together. You could have a class of all blue objects, a class of all types of vehicles, or a class of animals. Just as long as it makes sense why the objects are placed into the same class at that time.

This idea of objects and classes working together is relatively new. Some of the older coding languages do not have this kind of feature, which made them much harder to work with, and many beginners struggled because things didn't stay where they should, and the objects would move around on them. Python helped to fix this problem, though there are other modern languages that use this idea as well.

The classes and objects help us to keep things as organized as possible and will ensure that when you place an object in one place, that object is going to stay there throughout the code. This makes it easier for a beginner to learn more about how to code, without getting frustrated about why things are not working the way that they want.

The Functions

Before we end this chapter, we need to take a quick look at the idea of functions

and how these work in the codes that we are writing. Functions are going to be a set of expressions, and are sometimes known as statements even though they work in a slightly different manner. They are going to come to us in two different methods, and they can be kept anonymous or have a name based on the way you decide to use them. As you use them in your code, you can use them in a manner that is like how values are used, even similar to numbers and strings, but they are going to come with some special attributes to distinguish them from other parts of the code.

The functions that we use are going to be pretty special because they are diversified and you are able to see them show up in a lot of different types of codes that you decide to write. And you will see that there are a few different choices that a programmer is able to work with when they want to create these functions, including:

- __doc__: This is going to return the docstring of the function that you are requesting.
- Func_default: This one is going to return a tuple of the values of your default argument.
- Func_globals: This one will return a reference that points to the dictionary holding the global variables for that function.
- Func_dict: This one is responsible for returning the namespace that will support the attributes for all your arbitrary functions.
- Func_closure: This will return to you a tuple of all the cells that hold the bindings for the free variables inside of the function.

Working with functions allows you to do a variety of things in your code, such as taking one argument and passing it over to another function or another part of the code if you would like. Any function that is then able to come in and take on a brand new one as the argument is going to be known as a higher-order function inside of the code. Learning all of the different types of functions can be important in helping you write out the codes that you want.

You can find a lot of examples online about how to create classes and functions. A small example for a first test could be the following:

```
class Contact:
    def __init__(self, name, number):
        self.name = name
        self.number = number
```

```
    def printContact(self):
        print(self.name + " --  " + self.number)

contact = Contact("Simon", "888-888-888")
contact.printContact()
```

This class has two members, a constructor (__init__) that is responsible for building objects and a method, *printContact* , to print the value of properties: name and number. Using this class, an object named “contact” has been created.

Chapter 8: How to Raise and Manage Your Own Exceptions in the Code

Another topic that we need to explore at this point is how to raise and even manage some of the exceptions that we need to inside of the code. As you start to create some more of the codes that you would like to work with, there are already going to be a few exceptions that the program recognizes automatically and that it is going to use to help keep things running smoothly. In this case, these are going to be found inside the regular Python library and can show up for you without any work on your own, outside of managing them a bit if you want.

A good example of this would be when one of your users would like to work on the code and divide by zero. In Python and most other programming languages, this is not something the user is allowed to do. You are also able to raise some of your own exceptions to make sure that the user isn't able to go through and do certain actions, based on what your particular program is designed to complete and work on.

Now, let's split these up and get a better idea of what we are working with here. The first type of exception that we want to be able to focus on here is the ones that the Python compiler is able to recognize on its own. This could be something as simple as adding in a statement that is not recognized, using a keyword in the wrong place, or even misspelling along the way so that the compiler isn't sure what it is looking for in the first place.

As the programmer working on code in Python, it is always a good idea to know a few of the exceptions that the Python library already recognizes. This is going to make it easier to know what you can add to the code, and when you should expect one of these exceptions to turn up. The keywords, and some main exceptions, that can come up in the Python code as you do your work will include the following:

- Finally—this is the action that you will want to use to perform cleanup actions, whether the exceptions occur or not.
- Assert—this condition is going to trigger the exception inside of the code
- Raise—the raise command is going to trigger an exception manually inside of the code.
- Try/except—this is when you want to try out a block of code and

then it is recovered thanks to the exceptions that either you or the Python code raised.

The first thing that we need to focus on here is looking at how you could use the exceptions and their keywords above in your own code. When the ones above, the automatic ones, do show up in code, you need to do some preparation and know what you can do to make them easier to understand, and easier to work with. If you are creating a new code, and you notice that some kind of issue is showing up, or if the program seems to be off and you want to see what is wrong, it is possible to take a look to see if the compiler is raising up a new exception. The reason for this is because your program has had a chance to take a look at that code and then figure out what needs to happen next.

The good news here is that often when you encounter an issue like this, it is nothing big and won't take a lot of time for you to fix it up. For example, if your goal was to bring up a file, and you simply gave it the wrong name, either when you are trying to call it back up or when you first went through and named it, then your compiler is going to go through and raise a new exception. The program was then able to look through your code and noticed that you were doing some action that it is not set up to help you with until you write the spelling in the correct way.

A good way to start to look at how these exceptions can work is to do an example and see the steps that are needed. We can type this in, and then see what happens when the compiler raises the exception on us. To help out with this kind of example, open up the Python compiler and type in the code that is below to get started:

```
x = 10
y = 10
result = x/y #trying to divide by zero
print(result)
```

The output that you are going to get when you try to get the interpreter to go through this code would be:

```
>>>
Traceback (most recent call last):
    File "D: \Python34\tt.py", line 3, in <module>
    result = x/y
```

```
ZeroDivisionError: division by zero
>>>
```

When you take a look at this example, your compiler is going to bring up an error, simply because you or the user is trying to divide by zero. This is not allowed with the Python code so it will raise up that error. Now, if you leave it this way and you run the program exactly how it is, you are going to get a messy error message showing up, something that your user probably won't be able to understand. It makes the code hard to understand, and no one will know what to do next.

A better idea is to look at some of the different options that you can add to your code to help prevent some of the mess from before. You want to make sure that the user understands why this exception is being raised, rather than leaving them confused in the process. A different way that you can write out this code to make sure that everyone is on the same page includes:

```
x = 10
y = 0
result = 0
try:
        result = x/y
        print(result)
except ZeroDivisionError:
        print("You are trying to divide by zero.")
```

As we can see when we work with this kind of code is that when we put it into the Python compiler, it is going to be similar to the first example that we see with these exceptions, and it is going to work in pretty much the same way. the difference that we are going to see is that we changed up the information at the way end, and we made sure that the message that shows up here is something that the user is going to see when the user does raise this exception.

Instead of the user working with the exception and getting a long error code that they will not understand at all and can just make them confused, they will get a message that makes more sense. In the example that we did above, they are going to get the message that says "You are trying to divide by zero" show up on the screen. Now, you don't have to go through and do this kind of step if you don't want too, but it can definitely make your code more user-friendly overall.

How to define your own exceptions in the code

Creating Your Own Exceptions

As we went through the examples of coding for exceptions above, we were just spending our time looking at the exceptions that are automatically recognized by the Python library and compiler to start with. This is going to help us out because we can make the program more user-friendly and enjoy that the messages we provide are more personalized, rather than seeing a string of words and a code that no one else is able to understand.

This is just one level of working with the exceptions though. It is possible to take this to the next level and create some of our own exceptions. The type of exception that you will want to create will depend on the kind of program that you wish to work on, but there are so many ways that you are able to use this to your advantage, and the limits are just going to be your own imagination along the way.

For example, you may be working on a code and decide that the users should only be able to input certain numbers, and then the others are not allowed. This may work when you are creating a game for others to play. Or you could have an exception come up if you only want to let the user try to answer three times. Once the user has gone through all the guesses that they are allowed, the compiler will then raise one of these exceptions to tell the user they won't be allowed to guess again.

These exceptions are unique just to your code, and if you don't write them into the code, the compiler will just keep going, without recognizing that it is supposed to stop. You can add in any kind of exception that you want to this message, using a similar idea that we went with before. The code that you can do to make this happen includes:

```
class CustomException(Exception):
def_init_(self, value):
        self.parameter = value
def_str_(self):
        return repr(self.parameter)

try:
        raise CustomException("This is a CustomError!")
except CustomException as ex:
        print("Caught:", ex.parameter)
```

When you finish this particular code, you are done successfully adding in your

When you finish this particular code, you are done successfully adding in your own exception. When someone does raise this exception, the message “Caught: This is a CustomError!” will come up on the screen. You can always change the message to show whatever you would like, but this was there as a placeholder to show what we are doing. Take a moment here to add this to the compiler and see what happens.

There are many times when you are coding when you want to be able to handle and manage the exceptions that come up in the code. And the more coding that you do, especially when you do some of the more advanced codings that are in this guidebook. Make sure to take some time on a few of these codes above and practice writing them in your compiler to get a better feel for how they work, and to really get a good understanding for handling and raising exceptions in Python.

Part 3: What is Scikit-Learn?

Chapter 9: What is Scikit-Learn?

Now, it is time to get into some of the fun stuff that we are able to do when it comes to Python machine learning. Now that we have some of the basics down and we understand some of the topics above, it is time for us to look at our very first Python library that we can use as we go through some of the machine learning algorithms in the following chapters.

But first, we need to take a look at what Scikit-Learn is all about so we know how we can use it later. There are actually quite a few things that you are going to enjoy when it comes to the Scikit-Learn environment and the library that it brings to Python. This is often one of the first extensions of Python that programmers are going to add in, especially when they want to explore machine learning because it has a lot of the tools and other parts that can help out with these machine learning programming. Any time that you want to work on machine learning of any kind with Python, you will find that it is imperative that you have a good understanding of this library.

To start, we need to take a look at some of the background that comes with the Scikit-Learn library. This particular library for Python was developed in 200. Later on, the company started to grow a lot more and made some important changes. In fact, it has grown so much that right now, it has 30 contributors who are active, and a few paid sponsorships that help it to keep going, including ones from Google, the Python Software Foundation, and INRIA. This is good news for anyone who would like to program with the language because it means that while the library is still free to use, there are still developers working on maintaining it and improving it all the time.

With that in mind, it is likely that you have at least a few questions about this library and what you are able to do with it. First, this library is going to make it easier for a programmer to do machine learning because a ton of the supervised and unsupervised algorithms are inside this library. Plus, the algorithms have had the necessary adjustments done on them to help make sure they work in the Python environment, making it easier to do Python machine learning along the way.

With Scikit-Learn, the licensing comes under a permissive simplified BSD license, and many of the distributions of Linux are going to be able to use it in the process if you want to as well. It is also going to be built up thanks to the help of the SciPy library, so this helps out as well when it comes to ease of use.

The stack of resources that you are able to find with this one is larger and really goes to supporting machine learning in the process. The options you have for what is found in the Scikit-Learn library 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.

There are a few different modules and extensions that you are going to be able to use with SciPy, but they are all going to come together and be known as SciKits. This is why the module that is going to provide is with all of the machine learning algorithms that we are going to be using here are called the Scikit-Learn library. And each of them can come together to help us really get a lot out of the machine learning that we want to do.

This library is designed to make machine learning with Python as easy as possible. There are already a ton of different algorithms that come with this one, and they can easily be used inside of the Python library, making your job a lot easier in the process. Now let’s take a look at some of the different algorithms that you are able to use when it comes to the Scikit-Learn library and why this is actually able to help you to work with machine learning a bit better.

Chapter 10: Supervised Learning with Scikit-Learn

Now that we have had a chance to look at Scikit-Learn and some of the neat things that this library can provide to us, it is time to take a look at some of the different things that you can do with the algorithms that are supervised machine learning with this library. There are a lot of different types of learning algorithms that you are able to use with Scikit-Learn, and these can help you get a good start with machine learning. Some of the different types of machine learning that you are able to do with the Scikit-Learn library includes:

Support Vector Machines or SVMs

The first type of supervised learning that we are going to look at with Scikit-Learn is known as support vector machines or SVMs. These are going to be a set of learning methods that are supervised and can be used with detecting outliers, regression problems, and classification problems. There are a lot of different reasons that we would want to work with support vector machines, including:

1. They are going to be effective when you have spaces that are high dimensional.
2. They can still be effective when you have more dimensions than you do a number of samples to work with.
3. The SVM is going to work with a few training points to make any decision you need, which is going to be known as the support vectors, allowing the machine to be as efficient with its memory as possible.
4. There is a lot of versatility. There are functions of kernels that you are able to specify for the decision function. There are some of the common kernels that can be provided; however, the programmer is going to have the ability to create their own custom kernels in the process.

The SVMs are not going to work for every machine learning that you decide to do, though, despite how great they are. There are a few disadvantages that come with using the SVMs, and one of these is that if you have the features that you have to be higher in number than how many samples you have, you have to make sure that you do not overfit by choosing Kernel functions and regularization term is crucial. In addition, you will find that the SVMs, though you can do a lot with them, can't give you any kind of estimate when you are looking for probability. Instead, you can calculate the estimates with the help of

looking for probability. Instead, you can calculate the estimates with the help of expensive five-fold cross-validation. The cost of doing this can often get much higher than it is worth and sometimes a company will choose to go with a different kind of algorithm to get their work done.

With that in mind, we need to take a look at how we are going to be able to work with a classification problem that works with the support vector machine. For this one, you need to make sure that you have Scikit-Learn, Matplotlib, Pandas, and NumPy ready to go. The first step we have to do when you have all of this set up is to create a new data set to work with. The code that you can do with this one includes:

```
# importing scikit learn with make_blobs
from sklearn.datasets.samples_generator import make_blobs

# creating datasets X containing n_samples
# Y containing two classes
X, Y = make_blobs(n_samples = 500, centers = 2,
                  random_state = 0, cluster_std = 0.40)

# plotting scatters
plt.scatter(X[:, 0], X[:, 1], c = Y, s = 50, cmap = 'spring');
plt.show()
```

What the SVM does is to help you to go with the classes that are created and draws a line here. But you will also notice that it is going to take this a bit further and consider how wide the line will be in this region between the classes. To see how this works, you can try adding the following code to your compiler to see what happens:

```
# creating line space between -1 to 3.5
xfit = np.linspace(-1, 3.5)

# plotting scatter
plt.scatter(X[:, 0], X[:, 1], c=Y, s=50, cmap='spring')

# plot a line between the different sets of data
for m, b, d in [(1, 0.65, 0.33), (0.5, 1.6, 0.55), (-0.2, 2.9, 0.2)]:
    yfit = m * xfit + b
    plt.plot(xfit, yfit, '-k')
```

```
    plt.fill_between(xfit, yfit - d, yfit + d, edgecolor='none',
    color='#AAAAAA', alpha=0.4)

plt.xlim(-1, 3.5);
plt.show()
```

When all of this is in place, we are ready to start importing the data set that we want to use. This is going to be considered one of the intuition parts that come with SVM, and it is done in order to make sure that we are optimizing the linear discriminant model that represents the perpendicular distance that is present in the graph between the sets of data.

With this in mind, we want to be able to take the training data that we have worked on and use it to train our classifier. Before we train this all though and get it to work the way that we want, we first need to make sure that we have imported the cancer datasets, and that we have done this in a way that gives us acsv file that we can use. We will use this kind of file in order to make sure we can train the right features that we need. The code that is needed to make this happens is below:

```
# importing required libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

# reading csv file and extracting class column to y.
x = pd.read_csv("C:\...\cancer.csv")
a = np.array(x)
y  = a[:,30] # classes having 0 and 1

# extracting two features
x = np.column_stack((x.malignant,x.benign))
x.shape # 569 samples and 2 features

print (x),(y)
```

We are getting close to done with this one here, but first, we need to take some time to fit our SVM properly. We want to make sure that the classifier that we are using is able to fit with these points. While some people may find that working with the likelihood models and the math that goes with them, we will

leave those for later. Instead, our goal is to use the scikit-learn library in order to help us get the algorithm, to help by making it into a black box to handle the tasks above. The code that is needed for this, and the part that we will need to use to allow the model to predict any new values when the fitting is done includes:

```
# import support vector classifier
from sklearn.svm import SVC # "Support Vector Classifier"
clf = SVC(kernel='linear')

# fitting x samples and y classes
clf.fit(x, y)

clf.predict([[120, 990]])
```

clf.predict([[85, 550]])

Neural Networks

Another option that we can work with when it comes to supervised machine learning Scikit-Learn is the neural networks. These are going to be a bit different than the other kinds, but because they work similar to the human mind, it is amazing all of the different things that we are going to see them do, and how strong the learning can be.

The neural networks are going to be used in many applications of machine learning because they are great at analyzing and learning the different patterns that you have, simply by going through several layers at the time. The more layers that your neural network is able to go through, the more successful it is going to be with correctly analyzing and learning what the image is. As it goes through the different layers, each time that the neural network starts to see a new pattern in that layer, it can then automatically activate the process to push it into the next layer to look again. You will see that this process will continue going until all of the layers are complete, and then the algorithm will use the information it has in order to predict what it sees inside this image.

Now, we are going to see a few different actions happen at this point. If the algorithm was able to continue through all of these layers, and then make an accurate prediction when it is done, then the neurons are going to be stronger. This is a good thing because it provides a great and strong association between

the patterns and the object. And the stronger these neurons get the easier it is for the algorithm and the system to do this same process the next time that it is presented with it.

To get this algorithm to work the way that you would like, you need to be able to provide your system with some kind of image, in this case, a car. The neural network would take some time to search over the picture, starting along the first layer to see what is there, such as the outside edges of the car. Then it would move on from there to some of the other layers to see if there are other characteristics that should be noted in the picture as it goes through the different layers. If it is successful, the program will go through all of the different layers until it is able to find the little details that say the picture is a car.

It is possible that the neural network is going to find a ton of layers when it comes to working with this one. And this is a good thing. The more layers that the neural network is able to figure out, the more accurate it is going to be when it comes to making predictions. And when the neural network is successful, it is going to be able to remember a lot of the patterns that it sees and will store this knowledge to use next time.

This is one of the options that you would use when you are looking to do some different processes like recognizing the animal in a picture, how to define a car model, facial recognition, and more. Anything that requires the system to recognize what is found in a picture and an image and then brings back the results that you are looking for.

One advantage that a programmer may see when they are doing a neural network algorithm is that you won't have all of the statistical training to go with it to make it easier. Even without having to use all of the statistics, you can use the neural network to help you to find out what relationships are there between your different variables, even when the problem is nonlinear.

There are a few negatives that come with this, though. The biggest issue that comes with the neural network algorithm is that the computational cost is going to be pretty high. This makes it so that while you can get a lot of the results that you want with machine learning, the costs can be so high that it is hard to bring out these neural network algorithms as often as you would like.

Decision Trees

Another option that the programmer is able to use when they are doing things

with supervised learning with the help of this library is going to be the decision trees. These are going to be helpful when it comes to regress and classification. The goal with this kind of algorithm is to be able to create some kind of model that is able to go through and predict the value of a target variable, simply by being able to learn the decision rules that it can infer from the features of your data.

There are a lot of different advantages that come with being able to use these decision trees and the algorithms that come with it. Some of these advantages are going to include:

1. The decision trees are really simple to understand, and they are easy for even beginners to interpret. The trees are a model that can be visualized as well.
2. There isn't a big need to spend a lot of time preparing the data. But it does struggle if there are missing values, so you need to at least check that this is not a problem.
3. The cost of using one of these trees is going to be logarithmic when we are talking about how many different points of data we need to use to make sure our tree is trained in the proper manner.
4. It has the ability to handle categorical and numerical data. Some of the other techniques that you can use, and the other algorithms out there are only going to be able to handle one variable type.
5. It is possible to validate what is going on in this model with some statistical tests. This helps us to make sure that the model is as reliable as possible.
6. It is going to perform well even if you find that some of the assumptions were violated, or we have inconsistent data when we generated that tree.

While there are some benefits that come with using this kind of algorithm, there are also some negatives that we have to watch out for before we choose to go with this kind of machine learning algorithm. For example, sometimes, the learners of decision trees are going to create trees that are too complex and are not able to generalize the data they have all that well. This is a process known as overfitting and can make it hard to get the right data that you want. In addition, these data trees are sometimes seen as unstable because even a small variation in the data could give you a completely different tree than what you had before. And it is possible for a decision tree learning to make a tree that is biased if there are a few classes that dominate. It is, therefore, best to balance out your set of

are a few classes that dominate. It is, therefore, best to balance out your set of data before you decide to fit the decision tree.

The reason that some companies like to work with decision trees is that it allows them to take a look at all of the different possibilities that you are able to see all of the different decisions and the possible outcomes that are going to come with these. This allows you to make the best decisions for your business based on the data that you have.

In addition to working with this kind of library, you will need to install the parts known as graphviz and pydotplus. You are able to install these with your pip and your package manager. Graphviz is going to be a helpful tool to use here because it helps us to draw the graphics that we want with dot files and Pydotplus is going to be the module to help us do this.

Once both of those are installed, we are going to start out this work when we define the code and then collect up any of the data that is needed. Our goal is to make up a decision tree that will determine if someone is a woman or a man based on the inputs. There are three inputs that we will focus on here, including the length of their hair, how tall they are, and what pitch their voice is. First, we want to train out the data, and the code that we need to make this happen includes the following:

```
import pydotplus
from sklearn.datasets import load_iris
from sklearn import tree
import collections

# Data Collection
X = [ [180, 15,0],
     [177, 42,0],
     [136, 35,1],
     [174, 65,0],
    [141, 28,1]]

Y = ['man', 'woman', 'woman', 'man', 'woman']

data_feature_names = [ 'height', 'hair length', 'voice pitch' ]
```

After this is in your compiler and ready to go, we then want to take some time to train our classifier, which is going to be the decision tree, with the data that we

trained. Training may take a bit of time, but remember that it is a necessary part of every supervised learning algorithm, so we have to take the time to do it. The code that we need to make this happen includes:

```
# Training
clf = tree.DecisionTreeClassifier()
clf = clf.fit(X,Y)
```

From here, we need to focus on the visualization of the decision tree. The best code to use for this to make sure that we can visualize the tree would be below:

```
# Visualize data
dot_data = tree.export_graphviz(clf,
                                feature_names=data_feature_names,
                               out_file=None,
                                filled=True,
                                rounded=True)
graph = pydotplus.graph_from_dot_data(dot_data)

colors = ('turquoise', 'orange')
edges = collections.defaultdict(list)

for edge in graph.get_edge_list():
   edges[edge.get_source()].append(int(edge.get_destination()))

for edge in edges:
   edges[edge].sort()
   for i in range(2):
      dest = graph.get_node(str(edges[edge][i]))[0]
      dest.set_fillcolor(colors[i])

graph.write_png('tree.png')
```

These steps are going to help save the visualization that we want under an image format that is known as tree.png, so you are able to find it later. You can then go through and have the compiler execute the code and see what shows up. If you got a decision tree to show up on the screen, then this is a good sign that you have completed the code properly!

Naïve Bayes

It is also possible to work with an algorithm that is known as the Naïve Bayes

It is also possible to work with an algorithm that is known as the Naïve Bayes algorithm. This is going to be a unique one that can simplify what you want to do with some of the codings you have, and it makes it easier to even explain some of the more complex models and things that you want to do in machine learning, even to some people who may not understand how these work as much.

Here we are going to look at how to work on a new classification problem, and your goal is to come up with a new hypothesis and a design that comes with this. Then there is going to be a time when your stakeholders of the company will want to see the model that you are trying to produce with this information. Often, these stakeholders will want to see the information and learn about it, long before the information is even done. This can present a dilemma because you want to be able to show them what your plans are and your goals are, without having a finished product and without having to spend a lot of time and effort on this while everyone else is confused.

When you are first forming your hypothesis, it is likely that you will run into many different points of data that we want to get to work on our model. And then you are also going to have a lot of other variables and things show up in the different training that you do. With all of this information, and the different types of data, how are you supposed to be able to show off this information to people who may not really understand what is going on?

This is where the Naïve Bayes algorithm is going to come into play. It is going to be one of the best ways for you to showcase the model that you are working on, and even do a bit of demonstration of how it works, even when you see that it is at the earliest stages of being developed.

As you get more familiar with this algorithm, you will find that there are a lot of reasons to use it. The Naïve Bayes' model is easy to use and is effective at predicting the class of your test data sets, so it is the perfect choice for someone who wants to keep things simple or who is new to the whole process. Even though this algorithm is simple, it will perform well, and it has proven that it can do better than some of the other higher-class algorithms in some cases.

You do need to be careful with this one though because there are some negatives to using the Naïve Bayes' algorithm. First, when you are working with categorical variables, and you need to test data that hasn't been through the training data set, you will find that this model is not able to make a good prediction for you and will assign those data sets a 0 probability. You can add some other methods that will help to solve this issue, such as the Laplace

estimation, but it can be confusing for someone who is brand new to working in machine learning.

There are a lot of times when you will need to show the stakeholders your project, even when it is not done. And we can imagine how complicated having a project that is the beginning of a machine learning project is to describe to those who are not using it. The Naïve Bayes is a good algorithm to help us get started, can help the stakeholders see what is going on with the process, and can make it so much easier to get their questions answered, and can make it easier to continue on with the project that you are doing.

Working with supervised learning can open up a lot of doors when it comes to all of the different things that you are able to do with machine learning. We will look at unsupervised machine learning in the next chapter, but as you can see, there are already a lot of different options and algorithms that you are able to work with here. Take some time to try out a few of the codes above so you can see how they work, and get familiar with them for your machine learning needs.

Chapter 11: Unsupervised Machine Learning with Scikit-Learn

You can also work with some of the different unsupervised machine learning algorithms with the help of the Sciit-learn library. There are a lot of these algorithms that you are able to explore with the help of this Python library, but some of the most common ones that programmers like to focus on include:

K-Means Clustering and Other Clustering Options

There are a lot of different clustering algorithms that work well in the unsupervised machine learning that you want to learn how to use. Unsupervised learning is going to often come with data that is not labeled, and doing this clustering is going to help us to get it all organized and easier to read. And when we are looking at some of the different clustering algorithms that you can use, the K-Means clustering algorithm is the one that will show up the most. The goal of working with this algorithm, compared to some of the others, is to help us to look through our data and find clusters to see where they land and what they are trying to tell us.

First, we need to explore what the K-means algorithm is all about and why we would want to use it. This is basically a simple unsupervised machine learning algorithm that is going to work by clustering the data into K number of clusters. You are able to choose how many different clusters that you will want to use so that the algorithm knows what it is supposed to do and how to move each of the points that come in your data.

There are a lot of different times when you are able to use this particular algorithm for your own needs. Some of the examples would include anomaly detection, species clustering, clustering languages, news article clustering, clustering gene segmentation data, and image segmentation to name a few of the applications.

If you are interested in working with the K-Means algorithm, there are a few different codes and steps that you are able to use in order to implement this in Python. The set of data that we will choose to use is going to include 3000 entries along with 3 different clusters. This helps us to already have an idea of what our K value is going to be. So, the first step that we are going to take a look at here is going to be importing the right set of data using the code below:

```
%matplotlib inline
from copy import deepcopy
import numpy as np
import pandas as pd
from matplotlib import pyplot as plt
plt.rcParams['figure.figsize'] = (16, 9)
plt.style.use('ggplot')

# Importing the dataset
data = pd.read_csv('xclara.csv')
print(data.shape)
data.head()
```

```
(3000, 2)
```

```
# Getting the values and plotting it
f1 = data['V1'].values
f2 = data['V2'].values
X = np.array(list(zip(f1, f2)))
plt.scatter(f1, f2, c='black', s=7)

# Euclidean Distance Calculator
def dist(a, b, ax=1):
    return np.linalg.norm(a - b, axis=ax)


  # Number of clusters
  k = 3
  # X coordinates of random centroids
  C_x = np.random.randint(0, np.max(X)-20, size=k)
  # Y coordinates of random centroids
  C_y = np.random.randint(0, np.max(X)-20, size=k)
  C = np.array(list(zip(C_x, C_y)), dtype=np.float32)
  print(C)
```

```
  [[ 11.  26.]
   [ 79.  56.]
   [ 79.  21.]]
```

```
  # Plotting along with the Centroids
  plt.scatter(f1, f2, c='#050505', s=7)
  plt.scatter(C_x, C_y, marker='*', s=200, c='g')
```

```
# To store the value of centroids when it updates
C_old = np.zeros(C.shape)
# Cluster Lables(0, 1, 2)
clusters = np.zeros(len(X))
# Error func. - Distance between new centroids and old centroids
error = dist(C, C_old, None)
# Loop will run till the error becomes zero
while error != 0:
   # Assigning each value to its closest cluster
   for i in range(len(X)):
      distances = dist(X[i], C)
      cluster = np.argmin(distances)
      clusters[i] = cluster
   # Storing the old centroid values
   C_old = deepcopy(C)
   # Finding the new centroids by taking the average value
   for i in range(k):
      points = [X[j] for j in range(len(X)) if clusters[j] == i]
      C[i] = np.mean(points, axis=0)
   error = dist(C, C_old, None)


colors = ['r', 'g', 'b', 'y', 'c', 'm']
fig, ax = plt.subplots()
for i in range(k):
      points = np.array([X[j] for j in range(len(X)) if clusters[j] == i])
      ax.scatter(points[:, 0], points[:, 1], s=7, c=colors[i])
ax.scatter(C[:, 0], C[:, 1], marker='*', s=200, c='#050505')
```

Now, this may seem like a lot of code to write, but it is going to show us how to write out a code for the K-means in Python and get it to work. After you have had some time to type it out and then execute to see how it is going to work. You should end up with a cluster with three different clusters where most of your points are going to fit. There will be some outliers that seem to be away from the center that you have, but there are three distinct groups that will hold onto the majority of the points that are in your set of data.

Gaussian Mixture Models

The next thing that we are going to look at is the Gaussian Mixture Model. This one is going to follow the idea of clustering and will be similar to what we are going to find with the K-Means algorithm that we talked about before. This can make things easier for us and can show us just how great clustering can be when it comes to making predictions for a company to do better.

In real life, there are going to be a lot of different sets of data that can then be modeled using what is known as Gaussian Distribution. It is a very intuitive and natural method that assumes that the clusters come from various Distributions that are seen as Gaussian. Or, we will see that this distribution has gone through the model and wants to be able to turn it into not just one type of Gaussian Distribution, but several. This is going to be one of the core ideas that we are going to see with this model.

The best way to see how the Gaussian Mixture Model is going to work, we need to take a look at an example and some coding to help it make more sense. In the example that we are going to use here, we are looking at a set of data from the IRS. Inside of Python, we are able to implement GMM using the GaussianMixture class to make things easier.

One thing to note before we get started is that with this code, it may be difficult to run using an online computer. This is why an offline IDE is often recommended. If you do try it with your online computer, then it is possible that it is going to encounter some problems in the process.

When you are ready, it is time to load up the set of data from iris to your package. To make sure that this is as simple as possible, at least for this example, we are going to focus on just using the first two columns. This is the sepal length and also the sepal width to make it easier. The code that you need to use in order to make this happen includes:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from pandas import DataFrame
from sklearn import datasets
from sklearn.mixture import GaussianMixture

# load the iris dataset
iris = datasets.load_iris()

# select first two columns
X = iris.data[:, :2]

# turn it into a dataframe
d = pd.DataFrame(X)
```

```
# plot the data
plt.scatter(d[0], d[1])
```

When this part is done, we want to be able to take the data that we have and make sure that it is a mixture of a total of 3 Gaussians. That will take just a moment to complete, and then we can move on to do the clustering. This just means that we want to take each of the observations that we use and give them a label. During this process, we want to also make sure that we get the right amount of times that we want the iteration to go through for this kind of function so that it is able to converge. The code that you need to make this happen includes:

```
gmm = GaussianMixture(n_components = 3)

# Fit the GMM model for the dataset
# which expresses the dataset as a
# mixture of 3 Gaussian Distribution
gmm.fit(d)

# Assign a label to each sample
labels = gmm.predict(d)
d['labels'] = labels
d0 = d[d['labels'] == 0]
d1 = d[d['labels'] == 1]
d2 = d[d['labels'] == 2]

# plot three clusters in same plot
plt.scatter(d0[0], d0[1], c = 'r')
plt.scatter(d1[0], d1[1], c = 'yellow')
plt.scatter(d2[0], d2[1], c = 'g')
```

When this part of the code is done, it is time to go through and print off the converged log-likelihood value and the number of times the iteration has to go through so that we are certain the model will be able to converge in the end. The code that you are going to need to use in order to get this one to work includes:

```
# print the converged log-likelihood value
print(gmm.lower_bound_)

# print the number of iterations needed
```

```
# print the number of iterations needed
# for the log-likelihood value to converge
print(gmm.n_iter_)</div>
```

If you have typed this into the compiler the right way, you should get an answer that includes a 7 on the second line. this is going to tell you that you will need to work with 7 iterations here that have to happen to make the model above converge. You can go through and do some more iterations and go past the 7, there is not going to be a big change that we are going to see in this, and it is really just a waste of time to do more.

As you can see, there are a lot of different types of unsupervised machine learning that you are able to use when it comes to the Scikit-Learn library. This one is going to help you to really explore all of the algorithms that come with Scikit-Learn and can make exploring a bit more in the world of machine learning that much easier.

Part 4: The TensorFlow Library

Chapter 12: What is the TensorFlow Library

The next thing that we need to spend some time looking at is the TensorFlow Library. This is another option that comes from Python, and it can really help you to get some machine learning done. This one takes on a few different options of what you are able to do when it comes to machine learning, so it is definitely worth your time to learn how to use this option along with the algorithms that we talked about with the Scikit-Learn library.

TensorFlow is another framework that you are able to work with in Python machine learning, and it is going to offer the programmer a few different features and tools to get your project done compared to the others. You will find that the framework that comes with TensorFlow is going to come from the Google company, and it is helpful when you are trying to work on some models that are deep learning related. This TensorFlow is going to rely on graphs of data flow for numerical computation. And it is able to make sure that some of the different things that you can do with machine learning are easier than ever before.

TensorFlow is going to help us out in many different ways. First, it can help us with acquiring the data, training the models of machine learning that we are trying to use, helps to make predictions, and can even modify a few of the future results that we have to make them work more efficiently. Since each of these steps is going to be important when it comes to doing some machine learning, we can see how TensorFlow can come into our project and ensure we reach that completion that we want even better.

First, let’s take a look at what TensorFlow is all about and some of the background that comes with this Python library. The Brain team from Google was the first to develop TensorFlow to use on large scale options of machine learning. It was developed in order to bring together different algorithms for both deep learning and machine learning, and it is going to make them more useful through what is known as a common metaphor. TensorFlow works along with the Python language that we talked about before. In addition to this, it is going to provide the users with a front-end API that is easy to use when working on a variety of building applications.

It makes it a bit further, though. Even though you are able to work with TensorFlow and it matches up with the Python coding language while you do the coding and the algorithms, it is going to be able to change these up. All of the

applications that you use with the help of TensorFlow are going to be executed using the C++ language instead, giving them an even higher level of performance than before.

TensorFlow can be used for a lot of different actions that you would need to do to make a machine learning project a success. Some of the things that you can do with this library, in particular, will include running, training, and building up the deep neural networks, doing some image recognition, working with neural networks that are recurrent, digit classification, natural language processing, and even word embedding. And this is just a few of the things that are available for a programmer to do when they work with TensorFlow with machine learning.

Installing TensorFlow

With this in mind, we need to take some time to learn how to install TensorFlow on a computer before we are able to use this library. Just like we did with Scikit-Learn, we need to go through and set up the environment and everything else so that this library is going to work. You will enjoy that with this kind of library; it is already going to be set up with a few APIs for programming (we will take a look at these in more depth later on), including Rust, Go, C++ and Java to name a few. We are going to spend our time here looking at the way that the TensorFlow library is going to work on the Windows system, but the steps that you have to use to add this library to your other operating systems are going to be pretty much the same.

Now, when you are ready to set up and download the TensorFlow library on your Windows computer, you will be able to go through two choices on how to download this particular library. You can choose either to work with the Anaconda program to get it done, or a pip is going to work well, too. The native pip is helpful because it takes all of the parts that go with the TensorFlow library and will make sure that it is installed on your system. And you get the added bonus of the system doing this for you without needing to have a virtual environment set up to get it done.

However, this one may seem like the best choice, but it can come with some problems along the way. Installing the TensorFlow library using a pip can be a bit faster and doesn’t require that virtual environment, but it can come with some interference to the other things that you are doing with Python. Depending on what you plan to do with Python, this can be a problem so take that into consideration before starting.

The good thing to remember here is that if you do choose to work with a pip and it doesn't seem like it is going to interfere with what you are doing too much, you will be able to get the whole TensorFlow library to run with just one single command. And once you are done with this command, the whole library, and all of the parts that you need with it, are going to be set up and ready to use on the computer with just one command. And the pip even makes it easier for you to choose the directory that you would like to use to store the TensorFlow library for easier use.

In addition to using the pip to help download and install the TensorFlow library, it is also possible for you to use the Anaconda program. This one is going to take a few more commands to get started, but it does prevent any interference from happening with the Python program, and it allows you to create a virtual environment that you can work with and test out without a ton of interference or other issues with what is on your computer.

Though there are a few benefits to using the Anaconda program instead of a pip, it is often recommended that you install this program right along with a pip, rather than working with just the conda install. With this in mind, we will still show you some of the steps that it takes to just use the conda install on its own so you can do this if you choose.

One more thing that we need to consider here before moving on is that you need to double-check which version of Python is working. Your version needs to be at Python 3.5 or higher for this to work for you. Python 3 uses the pip 3 program, and it is the best and most compatible when it comes to working with a TensorFlow install. Working with an older version is not going to work as well with this library and can cause some issues when you try to do some of your machine learning codings.

You can work with either the CPU or the GPU version of this library based on what you are the most comfortable with. The first code below is the CPU version and the second code below is going to be the GPU version.

pip 3 install – upgrade tensorflow

pip 3 install – upgrade tensorflow-gpu

Both of these commands are going to be helpful because they are going to ensure that the TensorFlow library is going to be installed on your Windows system.

But another option that you are able to use is with the Anaconda package itself. The methods above were still working with the pip installs, but we talked about how there are a few drawbacks when it comes to this one.

Pipis the program that is already installed automatically when you install Python onto your system as well. But you may find out quickly that Anaconda is not. This means that if you want to ensure that you can get TensorFlow to install with this, then you need to first install the Anaconda program. To do this, just go to the website for Anaconda and then follow the instructions that come up to help you get it done.

Once you have had the time to install the Anaconda program, then you will notice that within the files there is going to be a package that is known as conda. This is a good package to explore a bit at this time because it is going to be the part that helps you manage the installation packages, and it is helpful when it is time to manage the virtual environment. To help you get the access that you need with this package, you can just start up Anaconda and it will be there.

When Anaconda is open, you can go to the main screen on Windows, click the Start button, and then choose All programs from here. You need to go through and expand things out in order to look inside of Anaconda at the files that are there. You can then click on the prompt that is there for Anaconda and then get that to launch on your screen. If you wish to, it is possible to see the details of this package by opening the command line and writing in “conda info”. This allows you to see some more of the details that you need about the package and the package manager.

The virtual environment that we talk about with the Anaconda program is going to be pretty simple to use, and it is pretty much just an isolated copy of Python. It will come with all of the capabilities that you need to maintain all of the files that you use, along with the directories and the paths that go with it too. This is going to be helpful because it allows you to do all of your coding inside the Python program, and allows you to add in some different libraries that are associated with Python if you choose.

These virtual environments may take a bit of time to adjust to and get used to, but they are good for working on machine learning because they give you an opportunity to isolate a project, and can help you to do some coding, without all of the potential problems that come with dependencies and version requirements. Everything you do in the virtual environment is going to be on its own, so you

can experiment and see what works and what doesn't, without messing up other parts of the code.

From here, our goal is to take the Anaconda program and get it to work on creating the virtual environment that we want so that the package from TensorFlow is going to work properly. The conda command is going to come into play here again to make this happen. Since we are going through the steps that are needed to create a brand new environment now, we will need to name it tensorenviron, and then the rest of the syntax to help us get this new environment created includes:

conda create -n tensorenvrion

After you type this code into the compiler, the program is going to stop and ask you whether you want to actually create the new environment, or if you would rather cancel the work that you are currently doing. This is where we are going to type in the "y" key and then hit enter so that the environment is actually created. The installation may take a few minutes as the compiler completes the environment for you.

Once the new environment is created, you have to go through the process of actually activating it. Without this activation in place, you will not have the environment ready to go for you. You just need to use the command of "activate" to start and then list out the name of any environment that you want to work with to activate. Since we used the name of tensorenviron earlier, you will want to use this in your code as well. An example of how this is going to look 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

When you get to this point, you will be presented with a list of all the packages that are available to install in case you want to add in a few others along with TensorFlow. You can then decide if you want to install one or more of these packages, or if you want to just stick with TensorFlow for right now. Make sure to agree that you want to do this and continue through the process.

The installation of this library is going to get to work right away. But it is going to be a process that takes some time so just let it go without trying to backspace or restart. The speed of your internet is going to make a big determinant of whether you will see this take a long time or not.

Soon though, the installation process for this library is going to be all done, and you can then go through and see if this installation process was successful or if you need to fix some things. The good news is the checking phase is going to be easy to work with because you can just use the import statement of Python to set it up.

This statement that we are writing is then going to go through the regular terminal that we have with Python. If you are still working here, like you should, with the prompt from Anaconda, then you would be able to hit enter after typing in the word python. This will make sure that you are inside the terminal that you need for Python so you can get started. Once you are in the right terminal for this, type in the code below to help us get this done and make sure that TensorFlow is imported and ready to go:

import tensorflow as tf

At this point, the program should be on your computer and ready to go and we can move on to the rest of the guidebook and see some of the neat things that you can do with this library. There may be a chance that the TensorFlow package didn't end up going through the way that it should. If this is true for you, then the compiler is going to present you with an error message for you to read through and you need to go back and make sure the code has been written in the right format along the way.

The good news is if you finish doing this line of code above and you don't get an error message at all, then this means that you have set up the TensorFlow package the right way and it is ready to use! With that said, we need to explore some more of the options and algorithms that a programmer can do when it comes to using the TensorFlow library and getting to learn how it works with the different machine learning projects you want to do.

Chapter 13: Working with the High-Level APIs

The first thing that we need to take a look at when working with the TensorFlow library is some of the high-level APIs that can help you get the work done. Some of the best high-level APIs that work well with this kind of machine learning library will include:

Keras

The first high-level API that we are going to look at that comes from TensorFlow is known as Keras. This is a high-level neural networks API, and it is written in Python while being able to run on top of other programs like Theano, CNTK, and TensorFlow. It was originally developed so that it would have a focus on enabling fast experimentation. This means that it is going to be able to go from idea to the result as quickly as possible, with the least amount of delay possible. And it does this by making that strong and sturdy research is used. You would want to use Keras if you want to work with a library for deep learning that can:

1. Allow for fast and easy prototyping. This is going to happen with user-friendliness, extensibility, and modularity.
2. Support both types of networks including recurrent and convolutional ones, as well as any kind of combination of both these networks.
3. Run seamlessly with both GPU and CPU.

There are a few different principles that happen with Keras that help to guide it and make sure that it is going to work the way that it should. First off, Keras was designed in order to be user-friendly. This one is an API that has been designed to be used by humans, rather than one that was designed to be used by machines. And because of this shift of focus, it makes sure that the experience of the user is front and center.

To help make sure that it is friendly for a user to work with, you are going to see that it has the best practices in place to reduce the amount of cognitive load, it offers APIs that are simple and consistent, and it is going to minimize how many actions are required by the user for most of the common problems that they are using this API for.

The next guiding principle that comes with Keras is the modularity. The model is going to be understood as a sequence or a graph of a standalone. The

configurable models are going to be plugged in together with Keras, cutting out as many restrictions as they can. In particular, it is going to work with the standalone modules such as regularization schemes, activation functions, initialization schemes, optimizers, cost functions, and neural networks and can help us to combine them together so that the programmer is able to work on any kind of model they need for their project.

Keras is also able to help out because it is easy to extend out. New modules are going to be really easy to add in keras, and you can just use them as new functions and classes. In addition, the existing modules are going to provide us with a lot of the examples that we need. To make sure that we are able to be totally expressive in the new modules that we are creating, we can use keras for advanced research in the process.

And finally, another guiding principle that makes Keras easy to work with and one of the best high-level API's to use with TensorFlow is that it can work with Python. All of the modules are going to be described with the help of a Python code. And since we have already seen how great this code can be, how compact it is, and how easy this coding language is even for beginners, we can see why it is such a good thing to use the Python language along with the Keras API.

Eager Execution

The next thing that we are going to take a look at here is the eager execution that comes with TensorFlow. This is going to be an imperative programming environment for you to use. It is helpful because it is able to come in and evaluate some of the operations that are there right away without delay, without needing to build up graphs in the process. The operations are also going to be able to return concrete values instead of constructing the computational graph that you will be able to run later.

This can be so good for the eager execution because it helps you to get started with the use of TensorFlow and some of the debugging models that come with it, while also reducing some of the boilerplate that comes with it. The eager execution is going to be a machine learning platform that is flexible and great for experimentation and research all at the same time. Some of the things that you are able to see when it comes to eager execution and what it can provide to the programmer includes:

1. An interface that is intuitive: You will be able to structure your code

in a natural manner while using the structures for data in Python. You are going to be able to iterate quickly on small data and small models.
2. Easier debugging. The call ops are going to directly inspect the running models and test some of the changes that happen. You can also see that eager execution is going to use some of the standard tools for debugging from Python in order to get error reporting that happens right away.
3. Natural control flow. You will be able to benefit from using the control flow from Python rather than the control flow of graphs. This is going to help us simplify some of the specifications that happen in dynamic models with this system.

Estimators

Estimators are going to be explored in more detail when it comes to TensorFlow, but it is definitely one of the high-level APIs found in this that can make it easier to do some of the programmings that are needed with machine learning. Estimators are going to help us with a few different actions to make life easier including export for serving, prediction, evaluation, and training.

You get the benefit of working with either a premade estimator to make things easier, or, if none of the pre-made options work for you, you can create your own estimator that is custom and works for your needs. All of the estimators, whether they are custom or pre-made, are going to be considered classes that are based on the class of tf.estimator.Estimator.

There are going to be a few benefits that you are going to enjoy when it comes to using these estimators. Some of the benefits that you are able to get with these estimators are going to include:

1. You are able to run models based on the Estimator on a local host or on a distributed multi-server environment, and you do not have to change up the model. You can also run it on the TPU, GPU, and CPU without needing to go through and recode the model.
2. Estimators are going to help make it easier to share any implementations that you have between model developers.
3. You can develop a great model that has code that is high level intuitive with the Estimators. This is going to make it a lot easier to

create models with the Estimators than it is to create these models with any of the lower-level APIs on TensorFlow.

4. You will find that since the Estimators can be built using the tf.keras.layers, it is easier to customize the process as you want.
5. Estimators are going to build up a graph for you.
6. Estimators are a great place to have safe distributed raining loops. These are going to help you have more control over how and when to do some of the following tasks:
    a. Build up a new graph
    b. Save some of the summaries to the TensorBoard
    c. Create a new checkpoint file and then recover from some of the failures.
    d. Handles the exceptions that come up
    e. Can load up the data that you are using.
    f. Initializes any of the variables that you are using.

Importing data

If you do any kind of work with TensorFlow, you will find that working with the feed-dict option to pass information inside of the TensorFlow library is one of the worst possible ways to make the information go through. This is slow, and it is going to end up with errors more often than not. The best for a programmer to get the data over to the models in TensorFlow is going to be an input pipeline in order to make sure that the GPU you are working with doesn't have to wait for some of the new stuff to come in.

The best part of this is that there is a built-in API inside of the TensorFlow library known as Dataset that can be used to make this task easier. We are going to take some time to work through all of the steps that the library needs so we can make our own input pipeline and then we can also look at some of the codings that are needed to make sure our data is fed into the model in the most efficient manner possible.

Now, we have to look at the three main steps that are needed in order to make sure we can use the Dataset API with TensorFlow. These three steps are going to include:

1. Importing the data. We will be able to do this by creating a Dataset instance with the data we want to use.

2. Create our Iterator. When we use the dataset that we created in order to make an Iterator instance, we can then have this iterate through the whole dataset.
3. Consuming data. Then we can take that iterator that we created in order to get the elements that we need from the dataset to feed the model.

Looking at the steps above, we need to start out with the importing data part of this process. This means that we need to start out by picking out the data that is needed to place inside the dataset. Often you will want to work with an array from NumPy and then pass it over to TensorFlow. The code that you can use to make this happen includes:

 # create a random vector of shape (100,2)
x = np.random.sample((100,2))
 # make a dataset from a numpy array
dataset = tf.data.Dataset.from_tensor_slices(x)

It is also possible to go through and pass more than one array from NumPy. One of the examples of using this can include when you have a few types of data that have been split up between labels and features. Some of the code that you would want to use with this will include the following:

features, labels = (np.random.sample((100,2)), np.random.sample((100,1)))
dataset = tf.data.Dataset.from_tensor_slices((features,labels))

Another method that you are able to use in order to initialize your own dataset is through a generator. You can often be fine using the NumPy from earlier, but this method is going to be pretty useful any time that you are going to work with a type of array that has element lengths that are different, such as a sequence. The code that you are able to use to make this one happen includes:

 # from generator
sequence = np.array([[[1]],[[2],[3]],[[3],[4],[5]]])def generator():
for el in sequence:
yield eldataset = tf.data.Dataset().batch(1).from_generator(generator,
output_types= tf.int64,
output_shapes=(tf.TensorShape([None, 1])))iter =
dataset.make_initializable_iterator()
el = iter.get_next()with tf.Session() as sess:

sess.run(iter.initializer)
print(sess.run(el))
print(sess.run(el))
print(sess.run(el))

After you have been able to initialize the dataset that you want to work with, it is time to go through and create the iterator. This is going to allow us to get our data back when we want, and it can iterate through the dataset while retrieving the values of data in real-time. You will find that while working with this one is going to provide us with four different types of iterators to use. These four iterator types are going to include:

1. One-shot. This is one that is able to iterate through the dataset just one time, and you won't be able to feed in value to it.
2. Initializable. This is something that you are able to change dynamically, calling up the operation for initialization all while making sure that the new data is passed on to the function of feed_dict. This one is nice because it is pretty much like a bucket that you are able to fill up with the stuff you want to use.
3. Reinitialize. This is going to be initialized from a completely different Dataset. It is going to be the most helpful to a programmer when they are working on any kind of training of their set of data where they would like to transform in an additional manner. This could include shuffling and testing a dataset. This is going to be similar to using a crane with a tower in order to go over and select the container that you want to use.
4. Feedable: This kind is going to be used in order to help you figure out which of the iterators to use. It is going to be similar to a tower crane that is able to select which of the cranes in your arsenal then get to go through and pick out the container that will work for you. This one is often not needed because it adds in too many steps.

And finally, we are able to work with the idea of consuming the data that we need. In this one, we need to be able to pass the data to a model to make it work, and all that we need to do to see this happen is to pass on the tensors generated from get_next()

In the little bit of code that we have below, we are going to have a Dataset that is going to come with two arrays from NumPy, and we will still stick with the example that we used in the first section. We need to take notice that it is time to

example that we used in the first section. We need to take notice that it is time to wrap up the function of .random.sample in another array through NumPy so that you can add in a dimension. This means that we need to batch the data that we want to use. The example of a code that we will need to use to make this happen is going to be shown below:

```
# using two numpy arrays
features, labels = (np.array([np.random.sample((100,2))]),
np.array([np.random.sample((100,1))]))dataset =
tf.data.Dataset.from_tensor_slices((features,labels)).repeat().batch(BATCH_SIZ
```

When we are done with this one, it is time to create the iterator that we want to use with this. The code for getting this one done is below:

```
iter = dataset.make_one_shot_iterator()
x, y = iter.get_next()
```

Now we are going to have a few different things come into play here that we are able to use to make sure we get the results that we want in the process. The first part is going to allow us to make our own model. In this example, we are going to work with what is known as a simple neural network.

After we are done with that part, the second part of the code that we are going to work with, the part that says EPOCHS, is going to be how we are able to use these tensors in a direct manner, the ones that the model is going to give us from the function of iter.get_next(), as the input for the first layer. We can then take this information and use it for our loss function and our labels as well. This is basically how we are going to make sure that all of the parts are wrapped in together. The codes that you need to make both of these parts happen will include:

```
# make a simple model
net = tf.layers.dense(x, 8) # pass the first value from iter.get_next() as input
net = tf.layers.dense(net, 8)
prediction = tf.layers.dense(net, 1)loss =
tf.losses.mean_squared_error(prediction, y) # pass the second value from
iter.get_net() as label
train_op = tf.train.AdamOptimizer().minimize(loss)

EPOCHS = 10
BATCH_SIZE = 16
```

```
# using two numpy arrays
features, labels = (np.array([np.random.sample((100,2))]),
np.array([np.random.sample((100,1))]))dataset =
tf.data.Dataset.from_tensor_slices((features,labels)).repeat().batch(BATCH_SIZ
= dataset.make_one_shot_iterator()
x, y = iter.get_next()# make a simple model
net = tf.layers.dense(x, 8, activation=tf.tanh) # pass the first value from
iter.get_next() as input
net = tf.layers.dense(net, 8, activation=tf.tanh)
prediction = tf.layers.dense(net, 1, activation=tf.tanh)loss =
tf.losses.mean_squared_error(prediction, y) # pass the second value from
iter.get_net() as label
train_op = tf.train.AdamOptimizer().minimize(loss)with tf.Session() as sess:
sess.run(tf.global_variables_initializer())
for i in range(EPOCHS):
_, loss_value = sess.run([train_op, loss])
print("Iter: {}, Loss: {:.4f}".format(i, loss_value))
```

This is a lot of coding, but take some time to write it all down and see what results you are able to get in the end. It is a good way to practice some of the Python codings that we talked about earlier and makes sure that we are able to see more when it comes to how to use the TensorFlow library with machine learning as well.

Chapter 14: The Estimators in TensorFlow

Now that we have had some time to look at the high-level APIs that you are able to focus on, it is time to take a quick look at some of the Estimators that you are able to work with when it comes to this library as well. These estimators can provide you with a lot of different benefits, and you are going to enjoy the freedom and more that they provide. Let's dive in so we can get a better idea of how these Estimators work and why you would want to use them in TensorFlow.

The Pre-Made Estimator

These Estimators are going to make it easier for you to work at a level that is much higher conceptually than some of the base APIs in this library. You do not have to spend time creating the computational graphs or even sessions like before because the Estimator is going to take care of all this for you. This makes it easier to do some of the programmings that you want.

The main thing that we need to take a look at here is some of the structural parts of the Estimators program. A TensorFlow program that is going to rely on some of these pre-made Estimators is going to include the four steps below to make things easier:

1. You will write out one or more dataset importing functions. For example, it is possible that you would create your first function so that it is going to import the set that you are using for training, and then the second function is done in order to import the settings for the test. Each of these needs to bring back two objects for you, including:
    a. A Tensor that has at least one label, but can come in with more than one label.
    b. A dictionary where we can take the keys and turn them into the values and names that we want to feature, and which become our tensors. These tensors contain the corresponding data that you need for the features.
2. Define the columns of the feature. Each of the tf.feature_column will identify a feature name, the type, and any of the pre-processing input that you need to use.
3. Instantiate the pre-made Estimator that is relevant.
4. Call a training, inference or evaluation method based on what you want to do with this particular code.

So, this brings up the question of why you would want to work with these pre-made estimators in the first place. These are going to encode some of the best practices to make your work easier and can help you to make sure that the coding gets done right. For someone who wants to write proper codes in Python and machine learning, but who are just getting started, then the pre-made Estimator is the best option to go with.

Checkpoint

The next thing that we need to look at in TensorFlow is the idea of a checkpoint. When you are looking at a Keras doc, you will find that there are several explanations of what this term is all about, but some of the things that you can remember about the checkpoint here include:

1. It is going to be a kind of architecture of a model that will allow you a chance to recreate the model.
2. It is going to be the weights of the model.
3. It could be a type of training configuration that you can use that includes epochs, optimizer, loss, and some other important meta-information.
4. It is the state of the optimizer, which is going to allow us to resume the training right where we left off.

Basically, the checkpoint is going to contain the information that is needed to save your current state in the experiment. This allows you to take a break or check on other parts of the code. Then you can go back and resume the training right where you left it rather than having to check back the whole time or start from the beginning again.

Making your Own Custom Estimator

The next thing that you can consider working on is creating one of your own custom estimators. While many beginners of TensorFlow are going to focus on using the pre-made estimators in order to get the work done and to ensure that they are getting things done in the right manner, there are going to be times when you need to create our very own estimator along the way that is then able to perform way that you want for your program.

At the heart of every type of estimator, whether you are working with a custom or a pre-made estimator, you are going to find the model function. This is going

to be the specific method that is going to build up the needed graphs for all the training, the evaluation, and the predictions that are needed for machine learning. If you happen to be using an estimator that was already made, this means that someone else was able to go through and implement all of the functions that you need for that model to work.

This makes life easier, but when you are working with a custom estimator, you will find that you are going to be the one responsible for writing out the functions for that model all on your own. This takes a bit more time and effort in order to complete, but it is going to be the way things to go the right way with the custom estimator.

When you are working on one of these custom estimators for your own needs in machine learning, there is a certain workflow that is going to be important to make things easier. The workflow that is recommended in most cases for doing this includes:

1. Making the assumption that there is a suitable estimator that is in existence, you will use this to help build up the first model that you have, and then you can also use it to help establish a good baseline to work from.
2. You can then build up and test the overall pipeline that you are working on, including any of the integrity and the reliability of the data using the Estimator that was pre-made for this.
3. If the suitable pre-made Estimator was available, you would want to go through and run a few experiments to see which of the options is going to work the best and give you the best results.
4. If you need it, and you think that it is going to improve your model better than the pre-made estimators are able to do, you can then consider going through and building up your own customer Estimator in the process.

Creating a custom estimator takes some time and effort, and it is not always very easy for a beginner in machine learning to do. This is why many programmers like to spend their time working with some of the pre-made options to help them get started in the process. But when it is needed, and when the other options available are not going to help suit your needs, then working with a custom estimator can be your best choice as well.

Chapter 15: Understanding the Low-Level APIs

In addition to some of the estimators that we talked about before, and the high-level APIs that we talked about a few chapters before, it is also time to fully understand a bit more about the lower-level APIs that make the TensorFlow library work the way that it is meant too. These low-level APIs are just as important to the functioning of the machine learning algorithms that you want to explore, so being able to understand them and how they work can be super important as well. Some of the different low-level APIs that we need to focus on include:

What is a Tensor

The first thing that we are going to take a look at when following the low-level APIs in TensorFlow is the idea of a tensor. This is basically going to be a type of mathematical object, and it is going to be represented by an array of components that are seen as functions of the coordinates of space. As we discussed earlier, Google was able to create its own framework for machine learning, known as TensorFlow, and it is going to use tensors simply because they will allow us to work with neural networks that are highly scalable.

Google was able to surprise the analysts of the industry when it started to open source the machine learning library of TensorFlow, but this may not have been such a good idea because this soon became the go-to machine learning framework out there for developers of all backgrounds. In the beginning, though, TensorFlow was just used internally by Google, and it ends up helping out the company even more if a lot of different developers learn how to use this library.

There are different ways to define one of these tensors. They can be defined as simply a single point or even a collection of isolated points of space, and sometimes it can even be something that is defined over a continuum of different points. When we look at the latter time, the elements of the tensor are going to be functions of positions, and then the tensor comes in and forms its own tensor field. This sounds really complicated, but it simply means that the tensor is going to be defined at all of the points in that space, rather than just at the one point or the one collection of points.

It is possible that the tensor is going to consist of just one single number. If this happens, then the tensor of order zero is going to be the way that it is referred to, or it could be known as a scalar. A good example of a scalar is going to be the

mass that we see in an object. And a scalar field is going to be seen as the density of a fluid as a function of position for example.

Another complicated tensor that we need to focus on is going to be known as the tensor of order one, which is often referred to as a vector instead. Just as the scalar-tensor, or any of the other tensors that you are going to work with, this one could be defined at just one point or several points, and it can vary continuously from one point to another point in order to define the field of the vector. So, if we are in a three-dimensional space that is normal, the vector is going to come in with three components of some kind. Or if you are working with a four-dimensional space or time, then the vector is going to come in with four components and so on with this same idea.

The Variables

The TensorFlow variable is going to be one of the best ways to represent a shared, persistent state manipulated by our program. The variables can be manipulated with the help of the class tf.Variable. This is going to represent the tensor which has a value that is going to be changed when you choose to run ops on it. Unlike the tf.Tensor objects, this variable is going to exist outside of just one call or one session.

Internally, this class is going to be able to store persistent tensor. The specific ops that you are in can be helpful at allowing you to read and modify the values of the tensor. And then you can see that the modifications that you use will show up through a lot of different sessions like this. The reason that this is so important is that if you have several workers who are able to work on your specific session, they will still be able to see the same values any time that they look at the tf.Variable.

The best method that you are able to use in order to create one of these variables is to bring out the function of tf.get_variable. This specific function is going to ask you to take a bit of time to specify the name of the variable that you want to use. Make sure that you are focusing on a variable name that makes sense for what you are doing because this name is going to be brought up again later on and can show up with other replicas to help you access this variable later on. It is also important to name this variable right so you can bring it out when exporting and checkpointing the models.

The tf.get_variable is also going to come into use because it allows the

programmer to go through a variable of the same name that was created earlier and reuse it, making it easier for the programmer to define the models that are going to reuse layers again.

Graphs and Sessions

TensorFlow is going to work with what is known as a dataflow graph in order to help us better represent the computations that we are doing when it comes to the different dependencies that will show up in all of the operations that we work with. What this ends up doing is leading us to a low-level programming model to use. There are a number of steps that we need to use with this one including defining the dataflow graph, then create a session where you are able to run one or more of the parts that come on that graph across a set of devices that can be both local and remote when it is ready.

If you plan to work with TensorFlow a bit with some low-level programming, then this section is going to help you. Some of the Higher Level APIs are going to be useful at times, but they are going to hide some of the details of the sessions and the graphs from the user at the end. But this one helps to bring them out a bit and can help us to see better how the APIs are going to be implemented.

Now we need to take a look at the dataflow graphs and why we would use them. The dataflow is going to be a common model in programming to use when we are doing parallel computing. In one of these graphs, the nodes are going to be there to represent some of the units that we need for computation, and then the edges are there to represent some of the data that was consumed, or some of the data that was produced through computation.

For example, when we are doing a TensorFlow graph, the tr.matmul operation would then be able to go back and correspond with just one of the nodes, one that has two edges that are income, which will be our multiplied matrices here, and would leave us with just one edge outgoing, which is going to end up being the result of the multiplication that we just did.

This is going to be beneficial to use, and the TensorFlow library is going to be able to leverage these benefits when it is ready to execute out the programs that it is ready to use. Some of the ways that the TensorFlow library is going to be able to do this includes:

1. Parallelism: By being able to use explicit edges to help show the

different dependencies that will show up in our operations can ensure that the system is able to see which operations that it is going to be able to work with, and which of these can be done in parallel

2. Distributed execution: By being able to work with the edges that are explicit with this part, we can see that our values are going to flow right in the middle of the operations, it is then possible to use this kind of library to partition the program through a lot of different devices, includes TPUs, GPUs, and CPUs, attached to a lot of different types of machines. The library is going to make sure to insert the necessary coordination and communication between all of these devices.
3. Compilation. The XLA compiler that comes with TensorFlow is going to use the information that comes inside the dataflow graph. This is done in order to generate faster code because it is able to fuse together some of the operations that are adjacent.
4. Portability: And finally, we are going to take a look at the dataflow graph and how it is going to help out by being language independent. This means that you are able to build it up in the Python language, store it in what is known as SavedModel, and then bring it back later as a C++ program if you prefer.

The next kind of graph that we need to take a look at is known as the tf.Graph. This is an important graph because it is going to hold onto two important types of information that are really important to us. The first bit of information is going to include the structure of the graph. The edges and the nodes that show up in the graph will be there, and they indicate how some of the individual operations will be composed together, but this graph doesn't come with a prediction on how you should use them.

You can think about the structure of this graph like assembly code. If you are able to stop and take some time to inspect what is there, it is going to convey some information that is useful, but it is still not going to contain all of the contexts that are conveyed in the source code.

The second kind of information that you will see with the tf.Graph is going to be the graph collection. TensorFlow is going to provide you with a lot of general mechanisms that are going to be used to store collections of metadata in your tf.Graph. The function of tf.add_to_collection will make it easier for you to revert your code back to the object list that has that key. Then you are able to use the tf.get_collection in order to look up all of the objects that have some kind of

the tf.get_collection in order to look up all of the objects that have some kind of association back to the key.

You will find that a lot of the different parts that come with this kind of library will use this facility. So, if you decide to create the tf.Variable, it is going to be added to the collections of trainable and global variables. Then, when you come back later in order to create a tf.train.Optimize and tf.train.Sver, the variables that come with these specific collections are going to be the arguments that you use as a default.

Another thing that we can take a look at when we are working with some of the graphs that come with this library is how to name operations. A tf.Graph is going to define a new namespace for the tf.Opearation objects it contains. In TensorFlow, it is going to automatically choose a new and unique name for each of the operations that are in the graph, but you can override this and can pick out your own names, ones that are descriptive and easier to make the program to debug and read as you need. There are a few ways that you can go through and override the name that TensorFlow decides to name the operation so that it makes more sense for you.

One of the easiest methods to do this though is going to be the function of tf.name_scope. This one is going to make it easier to add in a name scope prefix into all of the operations that you created, as long as it reaches the particular context. The current prefix for the name scope is going to be "/". You can then just add in the right name with this to make sense for your needs. If you find that a specific name scope has been used in that specific context, you can append on "_1" or "_2" and so on to make this work. A good look at how you would do this in coding would be the following:

```
c_0 = tf.constant(0, name="c")  # => operation named "c"

# Already-used names will be "uniquified".
c_1 = tf.constant(2, name="c")  # => operation named "c_1"

# Name scopes add a prefix to all operations created in the same context.
with tf.name_scope("outer"):
  c_2 = tf.constant(2, name="c")  # => operation named "outer/c"

  # Name scopes nest like paths in a hierarchical file system.
  with tf.name_scope("inner"):
    c_3 = tf.constant(3, name="c")  # => operation named "outer/inner/c"
```

```
  # Exiting a name scope context will return to the previous prefix.
 c_4 = tf.constant(4, name="c")  # => operation named "outer/c_1"

  # Already-used name scopes will be "uniquified".
 with tf.name_scope("inner"):
   c_5 = tf.constant(5, name="c")  # => operation named "outer/inner_1/c"
```

Save and Restore

The next thing that we need to take a look at is how to save and restore some of the variables that we are working on in our code. There is likely to be some point while working on a machine learning algorithm where you will want to save and restore the variables that you are working on. This allows you to come back to them later without them getting deleted or lost or you have to come back and restart all of the work that you are doing. So, let's get started.

As you can guess at this point, the variables that show up in the TensorFlow library are going to be one of the best ways for us to represent the shared and persistent state of manipulation that happens in your program. The tf.train.Saver constructor is going to give us the opportunity to add in the restore and save options that you need with that graph for all of the list, or you can choose to work with a more specified list that you can control, which includes the variables that come with the graph. The Saver object is then going to come in handy here because it will then provide us with some methods to run these options while offering us a way to specify the paths that will lead us over to the checkpoint files, which will make it easier to do any writing on them later on.

We will use the Saver option often because it is able to restore any of the variables that we took the time to define in our model earlier. If you plan to load up a model without having a good idea of how to build your graph up, such as when you are writing up a generic program to help load the models, then you need to make sure that you definitely go through saving and restoring these models, so you don't lose the work.

So, the first thing that we are going to want to focus on here is how to save the variables that we are doing with TensorFlow. This can be done when we choose to create a Sver with the function of tf.train.Saver()to make sure that we are then able to manage some of the variables that this model will bring up for us. The following code can show us how to call the tf.train.Saver.save method to ensure that the variables we have, are saved over to the checkpoint files when we want.

```
# Create some variables.
v1 = tf.get_variable("v1", shape=[3], initializer = tf.zeros_initializer)
v2 = tf.get_variable("v2", shape=[5], initializer = tf.zeros_initializer)

inc_v1 = v1.assign(v1+1)
dec_v2 = v2.assign(v2-1)

# Add an op to initialize the variables.
init_op = tf.global_variables_initializer()

# Add ops to save and restore all the variables.
saver = tf.train.Saver()

# Later, launch the model, initialize the variables, do some work, and save the
# variables to disk.
with tf.Session() as sess:
  sess.run(init_op)
  # Do some work with the model.
  inc_v1.op.run()
  dec_v2.op.run()
  # Save the variables to disk.
  save_path = saver.save(sess, "/tmp/model.ckpt")
  print("Model saved in path: %s" % save_path)
```

The next thing that we need to take a look at is how to restore some of the variables that we want to work with inside the code. You will find that the object of the tf.train.Saver is not only going to be able to save up the variables that we want to put into the checkpoint files, but it can come into use when we want to restore the variables.

Before we do this though, we need to note that when we go through and restore some of the variables we want to use, we do not need to stop and do any initializing them ahead of time to make this happen. A good example of the code that we are able to use here to make this happen includes:

```
tf.reset_default_graph()

# Create some variables.
v1 = tf.get_variable("v1", shape=[3])
v2 = tf.get_variable("v2", shape=[5])
```

```
# Add ops to save and restore all the variables.
saver = tf.train.Saver()

# Later, launch the model, use the saver to restore variables from disk, and
# do some work with the model.
with tf.Session() as sess:
  # Restore variables from disk.
  saver.restore(sess, "/tmp/model.ckpt")
  print("Model restored.")
  # Check the values of the variables
  print("v1 : %s" % v1.eval())
  print("v2 : %s" % v2.eval())
```

Working with some of the lower-level APIs that come with TensorFlow in order to help you to write some of the different machine learning codes that you want to create can make things easier. There is just so much that you are able to do with TensorFlow, and missing out on any of it can make machine learning that much harder. Take some time to look at the different things that you are able to do with the lower-level APIs inside of this library.

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.

Python and Machine Learning may be the answer that you are looking for when it comes to all of these needs and more. It is a simple process that can teach your machine how to learn on its own, similar to what the human mind can do, but much faster and more efficient. It has been a game-changer in many industries, and this guidebook tried to show you the exact steps that you can take to make this happen.

Some of the topics discussed in this guidebook concerning Python Machine Learning include:

- What machine learning is all about?
- The different types of machine learning that you can work with.
- How AI and machine learning are the same and how they are different
- A Python crash course that includes how to set up Python on your computer, writing conditional statements, and raising your own exceptions.
- What is Scikit-Learn and how to use this Python library?
- Some of the Supervised and unsupervised machine learning that you can do with the Scikit-Learn library.
- Working with the TensorFlow Library.
- Working on the high level and low-level APIs in this library
- What estimators have to do with TensorFlow and how you can make these work for your needs.

There is just so much that a programmer is able to do when it comes to using machine learning in their coding, and when you add it together with the Python coding language, you can take it even further, even as a beginner.

The next step is to start putting some of the knowledge that we discussed in this guidebook to good use. There are a lot of great things that you are able to do when it comes to machine learning, and when we are able to combine it together with the Python language, there is nothing that we can't do when it comes to

training our machine or our computer.

This guidebook took some time to explore a lot of the different things that you are able to do when it comes to Python machine learning. We looked at what machine learning is all about, how to work with it, and even a crash course on using the Python language for the first time. Once that was done, we moved right into combining the two of these together to work with a variety of Python libraries to get the work done.

If you have ever wanted to learn how to work with the Python coding language, or you want to see what machine learning is able to do for you, then this guidebook is the ultimate tool that you need! Take a chance to read through it and see just how powerful Python machine learning can be for you.