PYTHON FOR DATA ANALYSIS

A Basic Guide For Beginners, To Learn The Language Of Python Programming Codes Applied To Data Analysis With Libraries Software Pandas, NumPy And IPython

OLIVER R. SIMPSON

Python for Data Analysis

A Basic Guide for Beginners to Learn the Language of Python Programming Codes Applied to Data Analysis with Libraries Software Pandas, Numpy, and IPython

By

Oliver R. Simpson

Table of Contents

This eBook is provided with the sole purpose of providing relevant information on a specific topic for which every reasonable effort has been made to ensure that it is both accurate and reasonable. Nevertheless, by purchasing this eBook you consent to the fact that the author, as well as the publisher, are in no way experts on the topics contained herein, regardless of any claims as such that may be made within. As such, any suggestions or recommendations that are made within are done so purely for entertainment value. It is recommended that you always consult a professional prior to undertaking any of the advice or techniques discussed within.

This is a legally binding declaration that is considered both valid and fair by both the Committee of Publishers Association and the American Bar Association and should be considered as legally binding within the United States.

Furthermore, the information that can be found within the pages described forthwith shall be considered both accurate and truthful when it comes to the recounting of facts. As such, any use, correct or incorrect, of the provided information will render the Publisher free of responsibility as to the actions taken outside of their direct purview. Regardless, there are zero scenarios where the original author or the Publisher can be deemed liable in any fashion for any damages or hardships that may result from any of the information discussed herein.

Additionally, the information in the following pages is intended only for informational purposes and should thus be thought of as universal. As befitting its nature, it is presented without assurance regarding its prolonged validity or interim quality. Trademarks that are mentioned are done without written consent and can in no way be considered an endorsement from the trademark holder.

Introduction

Congratulations on purchasing *Python for Data Analysis: A Basic Guide for Beginners to Learn the Language of Python Programming Codes Applied to Data Analysis with Libraries Software Pandas, Numpy, and IPython* and thank you for doing so.

The following chapters will discuss the fundamental concepts of Data Analysis in light of the Python-based data libraries. You will start with a deep dive into the definition of "Big Data," along with its history and importance in the contemporary world. To further your understanding of this term, you will learn about different types of data. In the same chapter, you will learn all about Big Data Analytics and the steps involved in analyzing such large volumes of data. The topic of Data Analysis is incomplete without an understanding of the process of data mining, which can be defined as "the process of exploring and analyzing large volumes of data to gather meaningful patterns and rules." You will not only learn the end-to-end process of data mining but also be informed about its advantages and challenges. The first chapter of the book will conclude with an overview of select Data Analysis strategies.

In the chapter 2 of this book titled, "Fundamentals of Python and Data Analysis Libraries", you will be introduced to the concept of the "Python" programming language followed by a variety of Data Analysis libraries including "Django", "Scikit-Learn", "NumPy", "Matplotlib", "Pandas", "IPython", and "TensorFlow" among others. The majority of the Data Analysis and machine learning models are developed in Python, as it is well suited to develop sophisticated models and production engines that can be directly plugged into production systems. One of the greatest assets of Python is its extensive set of libraries that can help researchers who are less equipped with developer knowledge to easily execute data analysis and machine learning activities.

In chapter 3 of this book titled, "Predictive Modeling, Data Visualization, and Creation of Training Data Set," you will start by getting a brief overview of machine learning and various machine learning algorithms to help you understand the difference between "Big Data Analytics" and machine learning. We will then explore the concept of predictive analysis of data in the context of real-world customer behavior analysis, as practiced by most eCommerce business including "Amazon" and "Netflix." Machine learning models are used to generate predictions and forecasts for the growth of the business. A variety of data libraries that you will learn about in chapter 2 of this book will be used with the "Scikit-Learn" platform to give you a step-by-step walkthrough of how you can create your own predictive data analysis model starting with data exploration and data visualization. Once you have a good understanding of the raw data available for your analysis, you will be able to learn how to process raw data set and generating high-quality training and test data set for your model. This chapter will conclude a brief overview of scatter plots that can be used to visualize the data.

In the final chapter of the book titled "Applications of Big Data Analysis," you will learn how big data and big data analytics are being leveraged by businesses across the industrial spectrum, with a focus on eCommerce, healthcare, and entertainment industry.

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

START

Chapter 1: Introduction to Big Data and Big Data Analysis

Big Data

In 2001, Gartner defined Big data as "Data that contains greater variety arriving in increasing volumes and with ever-higher velocity." This led to the formulation of the "three V's." Big data refers to an avalanche of structured and unstructured data that is endlessly flooding and from a variety of endless data sources. These data sets are too large to be analyzed with traditional analytical tools and technologies but have a plethora of valuable insights hiding underneath.

The “Vs” of Big data

Volume – To be classified as big data, the volume of the given data set must be substantially larger than traditional data sets. These data sets are primarily composed of unstructured data with limited structured and semi-structured data. The unstructured data or the data with unknown value can be collected from input sources such as webpages, search history, mobile applications, and social media platforms. The size and customer base of the company is usually proportional to the volume of the data acquired by the company.

Velocity – The speed at which data can be gathered and acted upon the first to the velocity of big data. Companies are increasingly using a combination of on-premise and cloud-based servers to increase the speed of their data collection. The modern-day "Smart Products and Devices" require real-time access to consumer data in order to be able to provide them a more engaging and enhanced user experience.

Variety – Traditionally, a data set would contain majority of structured data with low volume of unstructured and semi-structured data, but the advent of big data has given rise to new unstructured data types such as video, text, and audio that require sophisticated tools and technologies to clean and process these data types to extract meaningful insights from them.

Veracity – Another "V" that must be considered for big data analysis is veracity. This refers to the "trustworthiness or the quality" of the data. For example, social media platforms like "Facebook" and "Twitter" with blogs and posts containing a hashtag, acronyms, and all kinds of typing errors can significantly reduce the reliability and accuracy of the data sets.

Value – Data has evolved as a currency of its own with intrinsic value. Just like traditional monetary currencies, the ultimate value of the big data is directly proportional to the insight gathered from it.

History of Big Data

The origin of large volumes of data can be traced back to the 1960s and 1970s when the Third Industrial Revolution had just started to kick in, and the development of relational databases had begun along with the construction of data centers. But the concept of big data has recently taken center stage primarily since the availability of free search engines like Google and Yahoo, free online entertainment services like YouTube, and social media platforms like Facebook. In 2005, businesses started to recognize the incredible amount of user data being generated through these platforms and services, and in the same year, an open-source framework called "Hadoop" was developed to gather and analyze these large data dumps available to the companies. During the same period, a non-relational or distributed database called "NoSQL" started to gain popularity due to its ability to store and extract unstructured data. "Hadoop" made it possible for the companies to work with big data with high ease and at a relatively low cost.

Today, with the rise of cutting edge technology, not only humans but machines also generate data. The smart device technologies like “Internet of things” (IoT) and “Internet of systems” (IoS) have skyrocketed the volume of big data. Our everyday household objects and smart devices are connected to the Internet and able to track and record our usage patterns as well as our interactions with these products and feeds all this data directly into the big data. The advent of machine learning technology has further increased the volume of data generated on a daily basis. It is estimated that by 2020, "1.7 MB of data will be generated per second per person." As the big data will continue to grow, its usability still has many horizons to cross.

Importance of Big Data

To gain reliable and trustworthy information from a data set, it is very important to have a complete data set that has been made possible with the use of big data technology. The more data we have, the more information and details can be extracted out of it. To gain a 360 view of a problem, and its underlying solutions, the future of big data is very promising. Here are some examples of the use of big data:

Product development – Large and small e-commerce businesses are increasingly relying upon big data to understand customer demands and expectations. Companies can develop predictive models to launch new products and services by using primary characteristics of their past and existing products and services and generating a model describing the relationship of those characteristics with the commercial success of those products and services. For example, a leading fast manufacturing commercial goods company, "Procter &

Gamble" extensively uses big data gathered from the social media websites, test markets, and focus groups in preparation for their new product launch.

Predictive maintenance – In order to leave project potential mechanical and equipment failures, a large volume of unstructured data such as error messages, log entries, and normal temperature of the machine must be analyzed along with available structured data such as make and model of the equipment and year of manufacturing. By analyzing this big data set using the required analytical tools, companies can extend the shelf life of their equipment by preparing for scheduled maintenance ahead of time and predicting future occurrences of potential mechanical failures.

Customer experience – The smart customer is aware of all of the technological advancements and is loyal only to the most engaging and enhanced user experience available. This has triggered a race among the companies to provide unique customer experiences analyzing the data gathered from customers' interactions with the company's products and services. Providing personalized recommendations and offers to reduce customer churn rate and effectively kind words prospective leads into paying customers.

Fraud and compliance – Big data helps in identifying the data patterns and assessing historical trends from previous fraudulent transactions to effectively detect and prevent potentially fraudulent transactions. Banks, financial institutions, and online payment services like “PayPal” are constantly monitoring and gathering customer transaction data in an effort to prevent fraud.

Operational efficiency – With the help of big data predictive analysis. Companies can learn and anticipate future demand and product trends by analyzing production capacity, customer feedback, and data pertaining to top-selling items and product returns to improve decision-making and produce products that are in line with the current market trends.

Machine learning – For a machine to be able to learn and train on its own, it requires a humongous volume of data, i.e. big data. A solid training set containing structured, semi-structured, and unstructured data will help the machine to develop a multidimensional view of the real world and the problem it is engineered to resolve.

Drive innovation – By studying and understanding the relationships between humans and their electronic devices as well as the manufacturers of these devices, companies can develop improved and innovative products by examining current product trends and meeting customer expectations.

“The importance of big data doesn’t revolve around how much data you have,

but what you do with it. You can take data from any source and analyze it to find answers that enable 1) cost reductions, 2) time reductions, 3) new product development and optimized offerings, and 4) smart decision making."- SAS

The Functioning of Big Data

There are three important actions required to gain insights from big data:

Integration – The traditional data integration methods such as ETL (Extract, Transform, Load) are incapable of collating data from a wide variety of unrelated sources and applications that are at the heart of big data. Advanced tools and technologies are required to analyze big data sets that are exponentially larger than traditional data sets. By integrating big data from these disparate sources, companies are able to analyze and extract valuable insight to grow and maintain their businesses.

Management – Big data management can be defined as "the organization, administration, and governance of large volumes of both structured and unstructured data." Big data requires efficient and cheap storage, which can be accomplished using servers that are on-premises, cloud-based, or a combination of both. Companies are able to seamlessly access required data from anywhere across the world and then processing this is a data using required processing engines on an as-needed basis. The goal is to make sure the quality of the data is high-level and can be accessed easily by the required tools and applications. Big data are gathered from all kinds of Dale sources including social media platforms, search engines, history and call logs. The big data usually contain large sets of unstructured data and the semi-structured data which are stored in a variety of formats. To be able to process and store this complicated data, companies require more powerful and advanced data management software beyond the traditional relational databases and data warehouse platforms.

New platforms are available in the market that is capable of combining big data with the traditional data warehouse systems in a "logical data warehousing architecture." As part of this effort, companies are required to make decisions on what data must be secured for regulatory purposes and compliance, what data must be kept for future analytical purposes, and what data has no future use and can be disposed of. This process is called "data classification," which allows a rapid and efficient analysis of a subset of data to be included in the immediate decision-making process of the company.

Analysis – Once the big data has been collected and is easily accessible, it can be analyzed using advanced analytical tools and technologies. This analysis will provide valuable insight and actionable information. Big data can be explored to make discoveries and develop data models using artificial intelligence and machine learning algorithms.

Types of Data

Now that you understand the concept of big data let us look at different types of data so you can choose the most appropriate analytical tools and algorithms based on the type of data that needs to be processed. Data types can be divided into two at a very high level: qualitative and quantitative.

Qualitative data – Any data that cannot be measured and only observed subjectively by adding a qualitative feature to the object it's called as "qualitative data." Classification of an object using unmeasurable features results in the creation of qualitative data. For example, attributes like color, smell, texture, and taste. There are three types of qualitative data:

- **“Binary or binomial data”** – Data values that signal mutually exclusive events where only one of the two categories or options is correct and applicable. For example, true or false, yes or no, positive or negative. Consider a box of assorted tea bags. You try all the different flavors and group the ones that you like as "good" and the ones you don't as "bad." In this case, "good or bad" would be categorized as the binomial data type. This type of data is widely used in the development of statistical models for predictive analysis.

- **“Nominal or unordered data”** – Data characteristics that lack an "implicit or natural value" can be referred to as nominal data. Consider a box of M&Ms. You can record the color of each M&M in the box in a worksheet, and that would serve as nominal data. This kind of data is widely used to assess statistical differences in the data set, using techniques like "Chi-Square analysis," which could tell you "statistically significant differences" in the amount of each color of M&M in a box.
- **“Ordered or ordinal data”** – The characteristics of this Data type do have certain "implicit or natural of value" such as small, medium, or large. For example, online reviews on sites like "Yelp," "Amazon," and "Trip Advisor" have a rating scale from 1 to 5, implying a 5-star rating is better than 4, which is better than 3 and so on.

Quantitative data – Any characteristics of the data that can be measured objectively are called as "quantitative data." Classification of an object in using measurable features and giving it a numerical value results and creation of quantitative data. For example, product prices, temperature, dimensions like length, etc. There are two types of quantitative data:

- **“Continuous Data”** – Data values that can be defined to a further lower level, such as units of measurement like kilometers, meters, centimeters, and on and on, are called the continuous data type. For example, you can purchase a bag of almonds by weight like 500g or 8 ounces. This accounts for the continuous data type, which is primarily used to test and verify different kinds of hypotheses such as assessing the accuracy of the weight printed on the bag of almonds.

- **“Discrete Data”** – numerical data value that cannot be divided and reduced to a higher level of precision, such as the number of cars owned by a person which can only be accounted for as indivisible numbers (you cannot have 1.5 or 2.3 cars), is called as discrete data types. For example, you can purchase another bag of ice cream bars by the number of ice cream bars inside the package, like four or six. This accounts for the discrete data type, which can be used in combination with a continuous data type to perform a regression analysis to verify if the total weight of the ice cream box (continuous data) is correlated with the number of ice cream bars (discrete data) inside.

Big Data Analytics

The terms of big data and big data analytics are often used interchangeably owing to the fact that the inherent purpose of big data is to be analyzed. "Big data analytics" can be defined as a set of qualitative and quantitative methods that can be employed to examine a large amount of unstructured, structured, and semi-structured data to discover data patterns and valuable hidden insights. Big data analytics is the science of analyzing big data to collect metrics, key performance indicators, and Data trends that can be easily lost in the flood of raw data, by using machine learning algorithms and automated analytical techniques. The different steps involved in "big data analysis" are:

Gathering Data Requirements – It is important to understand what information or data needs to be gathered to meet the business objective and goals. Data organization is also very critical for efficient and accurate data analysis. Some of the categories in which the data can be organized are gender, age, demographics, location, ethnicity, and income. A decision must also be made on the required data types (qualitative and quantitative) and data values (can be numerical or alphanumerical) to be used for the analysis.

Gathering Data – Raw data can be collected from disparate sources such as social media platforms, computers, cameras, other software applications, company websites, and even third-party data providers. The big data analysis inherently requires large volumes of data, the majority of which is unstructured with a limited amount of structured and semi-structured data.

Data organization and categorization – Depending on the company's infrastructure, Data organization could be done on a simple Excel spreadsheet or using many tools and applications that are capable of processing statistical data. Data must be organized and categorized based on data requirements collected in step one of the big data analysis process.

Cleaning the data – to perform the big data analysis sufficiently and rapidly, it is very important to make sure the data set is void of any redundancy and errors. Only a complete data set fulfilling the Data requirements must have proceeded to the final analysis step. Preprocessing of data is required to make sure the only high-quality data is being analyzed, and company resources are being put to good use.

“Big data is high-volume, high-velocity, and/or high-variety information assets that demand cost-effective, innovative forms of information processing that enable enhanced insight, decision-making, and process automation.”
- *Gartner*

Analyzing the data – Depending on the insight that is expected to be achieved by the completion of the analysis, any of the following four different types of big data analytics approach can be adopted:

- **Predictive analysis** – This type of analysis is done to generate forecasts and predictions for future plans of the company. By the completion of a predictive analysis of the company's big data, the future state of the company can be more precisely predicted and derived from the current state of the company. The business executives are keenly interested in this analysis to make sure the company day-to-day operations are in line with the future vision of the company.

For example, to deploy advanced analytical tools and applications in the sales division of a company, the first step is to analyze the leading source of data. Once it believes source analysis has been completed, the type and number of communication channels for the sales team must be analyzed. This is followed by the use of machine learning algorithms on customer data to gain insight into how the existing customer base is interacting with the company's products or services. This predictive analysis will conclude with the deployment of artificial intelligence-based tools to skyrocket the company's sales.

- **Prescriptive analysis** – Analysis that is carried out by primarily focusing on the business rules and recommendations to generate a selective analytical path as prescribed by the industry standards to boost company performance. The goal of this analysis is to understand the intricacies of various departments of the organization and what measures should be taken by the company to be able to gain insights from its customer data by using a prescribed analytical pathway. This allows the company to embrace domain specificity and conciseness by providing a sharp focus on its existing and future big data analytics process.

- **Descriptive analysis** – All the incoming data received and stored by the company can be analyzed to produce insightful descriptions on the basis of the results obtained. The goal of this analysis is to identify data patterns and current market trends that can be adopted by the company to grow its business. For example, credit card companies often require risk assessment results on all prospective customers to be able to make predictions on the likelihood of the customer failing to make their credit payments and make a decision whether the customer should be approved for the credit or not. This risk assessment is primarily based on the customer's credit history but also takes into account other influencing factors, including remarks from other financial institutions that the customer had approached for credit, customer income, and financial performance as well as their digital footprint and social media profile.

- **Diagnostic analysis** – As the name suggests, this type of analysis is done to "diagnose" or understand why a certain event unfolded and how that event can be prevented from occurring in the future or replicated if needed. For example, web marketing strategies and campaigns often employ social media platforms to get publicity and increase their goodwill. Not all campaigns are as successful as expected; therefore, learning from failed campaigns is just as important, if not more. Companies can run diagnostic analysis on their campaign by collecting data pertaining to the "social media mentions" of the campaign, number of campaign page views, the average amount of time spent on the campaign page by an individual, number of social media fans and followers of the campaign, online reviews and other related metrics to understand why the campaign failed, and how future campaigns can be made more effective.

The big data analysis can be conducted using one or more of the tools listed below:

- Hadoop – Open source data framework.
- Python – Programming language widely used for machine learning.

- SAS – Advanced analytical tool used primarily for big data analysis.
- Tableau – Artificial intelligence-based tool used primarily for data visualization.
- SQL – the Programming language used to extract data from relational databases.
- Splunk – Analytical tool used to categorize machine-generated data
- R-programming – the Programming language used primarily for statistical computing.

Big Data Analysis vs. Data Visualization

In the wider data community, data analysis and data visualization are increasingly being used synonymously. Professional data analysts are expected to be able to skillfully represent data using visual tools and formats. On the other hand, new professional job positions called "Data visualization expert" and "data artist" have hit the market. But companies stool need professionals to analyze their data and extract valuable insights from it. As you have learned by now, Data analysis or big data analysis is an "exploratory process" with defined goals and specific questions that need to be answered from a given set of big data. Data visualization pertains to the visual representation of data, using tools as simple as an Excel spreadsheet or as advanced as dashboards created using Tableau. Business executives are always short on time and need to capture a whole lot of details; therefore, the data analyst is required to use effective visualizations that can significantly lower the amount of time needed to understand the presented data and gather valuable insights from the data.

By developing a variety of visual presentations from the data, an analyst can view the data from different perspectives and identify potential data trends, outliers, gaps, and anything that stands out and warrants further analysis. This process is referred to as "visual analytics." Some of the widely used visual representations of the data are "dashboard reports," "infographics," and "data story." These visual representations are considered as the final deliverable from the big data analysis process, but in reality, they frequently serve as a starting point for future political activities. The two completely different activities of data visualization and big data analysis are inherently related and loop into each other by serving as a starting point as well as the endpoint of the other activity. (The concept of data visualization will be explained in detail in chapter 3 of this book.)

Data Mining

Data mining can be defined as “the process of exploring and analyzing large volumes of data to gather meaningful patterns and rules.” Data mining falls under the umbrella of data science and is heavily used to build artificial intelligence-based machine learning models, for example, search engine algorithms. Although the process of “digging through data” to uncover hidden patterns and predict future events has been around for a long time and referred to as “knowledge discovery in databases,” the term “Data mining” was coined as recently as the 1990s.

Data mining consists of three foundational and highly intertwined disciplines of science, namely, "statistics" (the mathematical study of data relationships), "machine learning algorithms" (algorithms that can be trained with an inherent capability to learn), and "artificial intelligence" (machines that can display human-like intelligence). With the advent of big data, Data mining technology has been evolved to keep up with the "limitless potential of big data" and affordable computing power. The once considered tedious, labor-intensive, and time-consuming activities have been automated using advance processing speed and power of the modern computing systems.

“Data mining is the process of finding anomalies, patterns, and correlations within large data sets to predict outcomes. Using a broad range of techniques, you can use this information to increase revenues, cut costs, improve customer relationships, reduce risks, and more.”

– SAS

According to SAS, "unstructured data alone makes up 90% of the digital universe". This avalanche of big data does not necessarily guarantee more knowledge. The application of data mining technology allows the filtering of all the redundant and unnecessary data noise to garner the understanding of relevant information that can be used in the immediate decision-making process.

The Data Mining Process

Most widely used data mining processes can be broken down into six steps as listed below:

1. Business understanding

It is very critical to understand the project goals and what is it that you're trying to achieve through the data mining process. Companies always start with the establishment of a defined goal and a project plan that includes details such as individual team member roles and responsibility, project milestones, project timelines, and key performance indicators and metrics.

2. Data understanding

Data is available from a wide variety of input sources and in different formats. With the use of data visualization tools, the data properties, and features can be assessed to ensure the existing data set is able to meet the established business requirements and project goals.

3. Data preparation

The preprocessing of data collected in multiple formats is very important. The data set must be scrubbed to remove data redundancies and identify gaps before it is deemed appropriate for mining. Considering the amount of data to be analyzed, the data pre-processing and processing steps can take a long time. To enhance the speed of the data mining process, instead of using a single system company, prefer using distributed systems as part of their "database management systems." The distributed systems also provide enhanced security measures by segregating the data into multiple devices rather than a single data warehouse. At this stage, it is also very crucial to account for backup options and failsafe measures in the event of data loss during the data manipulation stage.

4. Data modeling

Applicable mathematical models and analytical tools are applied to the data set to identify patterns.

5. Evaluation

The modeling results and data patterns are evaluated against the project goal and objectives to determine if the data findings can be released for use across the organization.

6. Deployment

Once the insights gathered from the data have been evaluated as applicable to the functioning and operations of the organization, these insights can be shared across the company to be included in its day-to-day operations. With the use of a Business Intelligence tool, the data findings can be stored at a centralized location and accessed using the BI tool as needed.

Pros of Data Mining

Automated decision-making

With the use of data mining technology, businesses can seamlessly automate tedious manual tasks and analyze large volumes of data to gather insights for the routine and critical decision-making process. For example, financial lending institutions, banks, and online payment services use data mining technology to detect potentially fraudulent transactions, verify user identity, and ensure data privacy to protect their customers against identity theft. When the company's operational algorithms are working in coordination with the data mining models, the company can independently gather, analyze, and take actions on data to improve and streamline their operational decision-making process.

Accurate prediction and forecasting

Project planning is fundamental to the success of any company. Managers and executives can leverage data mining technology to gather reliable forecasts and predictions on future market trends and include in their future planning process. For example, one of the leading retail company "Macy's" has implemented demand forecasting models to generate reliable demand forecasts for Mary is clothing categories at individual stores in order to increase the efficiency of their supply chain by routing the forecasted inventory to each store and cater to the needs of the market more efficiently.

Cost reduction

With the help of data mining, technology companies can maximize the use of their resources by smarty allocating them across the business model. The use of data mining technology in planning, as well as an automated decision-making process, results in accurate forecasts leading to significant cost reductions. For example, a major airline company "Delta" implemented RFID chips inside their passengers checked-in baggage and gathered baggage handling data that was analyzed using data mining techniques to identify improvement opportunities in their process and minimizing the number of mishandled baggage. This not only resulted in a cost-saving on the search and rerouting process of the lost baggage but also translated into higher customer satisfaction.

Customer insights

Companies across different industrial sectors have deployed Data mining models to gather valuable insights from existing customer data, which can be used to segment and target customers with similar shopping attributes using similar marketing strategies and campaigns. Customer personas can be created using data mining techniques to provide a more engaging and personalized user experience to the customer. For example, "Disney" has recently invested over a billion dollars in developing and deploying "Magic bands," offering the convenience and enhanced the experience of Disney resorts. At the same time, these bands can be used to collect data on patron activities and interactions with different "Disney" products and services at the park to further enhance the "Disney experience."

"When [data mining and] predictive analytics are done right, the analyses aren't a means to a predictive end; rather, the desired predictions become a means to analytical insight and discovery. We do a better job of analyzing what we need to analyze and predicting what we want to predict."

– Harvard Business Review Insight Center Report

Challenges of data mining

1. Big data

Our digital life has inundated companies with large volumes of data, which is estimated to reach 1.7 MB per second per person by 2020. This exponential increase in volume and complexity of big data has presented challenges for the data mining technology. Companies are looking to expedite their decision-making process by swiftly and efficiently extracting and analyzing data to gain valuable insights from their data treasure trove. The ultimate goal of data mining technology is to overcome these challenges and unlock the true potential of data value. The "4Vs" of big data namely velocity, variety, volume, and veracity, represent the four major challenges facing the data mining technology.
The skyrocketing "velocity" or speed at which new data is being generated poses a challenge of increasing storage requirements. The "variety" or different data types collected and stored require advance data mining capabilities to be able to simultaneously process a multitude of data formats. Data mining tools that are not equipped to process such highly variable big data provide low value due to their inefficiency and analyzing unstructured and structured data together.

The large volume of big data is not only challenging for storage, but it’s even more challenging to identify correct data in a timely manner, owing to a massive reduction in the speed of the data mining tools and algorithms. To add on to this challenge, the data “veracity” denoting that all of the collected data is not accurate and can be incomplete or even biased. The data mining tools are struggling to deliver high-quality results in a timely manner by analyzing high quantity or big data.

2. Overloading models

Data models that describe the natural errors of the data set instead of the underlying patterns are often “over-fitted” or overloaded. These models tend to be highly complex and require a large number of independent media values to precisely predict a future event. Data volume and variety further increase the risk of overloading. A high number of variables tend to restrict the data model within the confines of the known sample data. On the other hand, an insufficient number of variables can compromise the relevance of the model. To obtain the required number of variables for the data mining models, to be able to strike a balance between the accuracy of the results and the prediction capabilities is one of the major challenges facing the data mining technology today.

3. Data privacy and security

To cater to the large volume of big data generated on a daily basis, companies are investing in cloud-based storage servers along with their on-premise servers. The cloud computing technology is relatively new in the market, and the inherent nature of this service poses multiple security and privacy concerns. Data privacy and security is one of the biggest concerns of Smart consumers who are willing to take their business to the company that can promise them the security of their personal information and data. This requires organizations to evaluate their customer relationships and prioritize customer privacy over the development of policies that can potentially compromise customer data security.

4. Scaling costs

With the increasing speed of data generation leading to a high volume of complex data, organizations are required to expand their data mining models and deploy them across the organization. To unlock the full potential of data mining tools, companies are required to heavily invest in computing infrastructure and processing power to efficiently run the data mining models. Big-ticket item purchase-including data servers, software, and advanced computers must be made to scale the analytical capabilities of the organization.

Data Mining Trends

Increased Computing Speed

With the increasing volume and complexity of big data, Data mining tools need more powerful and faster computers to efficiently analyze data. The existing statistical techniques like "clustering" art equipment to process only thousands of input data with a limited number of variables. However, companies are gathering over millions of new data observations with hundreds of variables making the analysis too complicated for the computing system to process. The big data is going to continue to explode, demanding supercomputers that are powerful enough to rapidly and efficiently analyze the growing big data.

Language Standardization

The data science community is actively looking to standardize a language for the data mining process. This ongoing effort will allow the analyst to conveniently work with a variety of data mining platforms by mastering one standard Data mining language.

Scientific Mining

The success of data mining technology in the industrial world has caught the eye of the scientific and academic research community. For example, psychologists are using “association analysis” to capture her and identify human behavioral patterns for research purposes. Economists are using protective analysis algorithms to forecast future market trends by analyzing current market variables.

Web mining

Web mining can be defined as “the process of discovering hidden data patterns and chains using similar techniques of data mining and applying them directly on the Internet.” The three main types of web mining are: “content mining,” “usage mining,” and “structure mining.” For example, “Amazon” uses web mining to gain an understanding of customer interactions with their website and mobile application, to provide more engaging and enhanced user experience to their customers.

Data mining tools

Some of the most widely used data mining tools are:

Orange

Orange is "open-source component-based software written in Python." It is most frequently used for basic data mining analysis and offers top-of-the-line data pre-processing features.

RapidMiner

RapidMiner is an "open-source component-based software written in Java." It is most frequently used for "predictive analysis" and offers integrated environments for "machine learning," "deep learning," and "text mining."

Mahout

Mahout is an open-source platform primarily used for unsupervised learning process" and developed by "Apache." It is most frequently used to develop "machine learning algorithms for clustering, classification, and collaborative filtering." This software requires advanced knowledge and expertise to be able to leverage the full capabilities of the platform.

MicroStrategy

MicroStrategy is a “business intelligence and data analytics software that can complement all data mining models.” This platform offers a variety of drivers and gateways to seamlessly connect with any enterprise resource and analyze complex big data by transforming it into accessible visualizations that can be easily shared across the organization.

Data Analysis Strategies

Data science is mainly used in decision-making by making precise predictions with the use of “predictive causal analytics,” “prescriptive analytics,” and machine learning.

Predictive causal analytics – It can be applied to develop a model that can accurately predict and forecast the likelihood of a particular event occurring in the future. For example, financial institutions use predictive causal analytics-based tools to assess the likelihood of a customer defaulting on their credit card payments by generating a model that can analyze the payment history of the customer with all of their borrowing institutions.

Prescriptive analytics - The "prescriptive analytics" are widely used in the development of "intelligent tools and applications" that are capable of modifying and learning with dynamic parameters and make their own "decisions." The tool not only predicts the occurrence of a future event but is also capable of providing recommendations on a variety of actions and its resulting outcomes. For example, the self-driving cars gather driving-related data with every driving experience and use it to train themselves to make better driving and maneuvering decisions.

Machine learning to make predictions – To develop models that can determine future trends based on the transactional data acquired by the company, machine learning algorithms are a necessity. This is considered "supervised machine learning," which we will elaborate on later in this book. For example, fraud detection systems use machine learning algorithms on the historical data pertaining to fraudulent purchases to detect if a transaction is fraudulent.

Machine learning for pattern discovery – To be able to develop models that are capable of identifying hidden data patterns but lack required parameters to make future predictions, the “unsupervised machine learning algorithms,” such as “Clustering,” need to be employed. For example, telecom companies often use the “clustering” technology to expand their network by identifying network tower locations with optimal signal strength in the targeted region.

basic
PHP
C
T-SQL
C++
Ruby
Perl
C
C++
Java
Delphi
Pascal
Program
Scala
C++
Python
Python
PL-SQL

Chapter 2: Fundamentals of Python and Data Analysis Libraries

Python

Python is a highly useful software programming language, which is rapidly becoming a standard in big data analysis. It is free with open source code and fully standardized across multiple operating systems, including “Windows,” “MacOS,” and “Linux.” Python is touted as an extremely versatile, simple to use and learn the programming language, and ideal for software programming beginners. In the 1980s, Python was developed, by Guido van Rossum at the "Center Wiskunde & Informatica (CWI), Netherlands," as a successor to the "ABC language" and with the capability of managing and interacting with the "Amoeba operating system.” It was launched at the end of the year 1989. In late 2000, Python v2.0 was introduced with a variety of enhancements such as "cycle-detecting garbage collector" and support for "Unicode." In December 2008, Python v3.0 was released. I, which resulted in a complete revision of Python making it more backward-compatible. Several of the main characteristics of Python v3.0 have been reverted to versions "2.6.x" and "2.7.x".

Python-based software programs are extremely readable and tend to be more succinct than similar programs written in other programming languages such as "C" or "Fortran." In addition, Python integrates seamlessly with conventional modules that provide a wide range of features and algorithms for tasks such as "parsing text data, manipulating and finding disk files, reading and writing compressed files, and downloading data from web servers." Python is equipped with capabilities of all the sophisticated methods including object-orientation that are prerequisites for advanced programmers.

Python is rather distinct from "C," "C++," or "Fortran," which require that the source code is first compiled into an executable format prior to the execution. On the other hand, there is no compilation phase in Python, and the source code is processed on a line-by-line basis. This means Python will execute the source code as though it is written as a script. The significant benefit of an interpreted language is that it tends to be extremely versatile; the variables are not required to be indicated in advance, and the program can easily adapt on the fly. However, there is also a drawback, in that statistically-rich programs based on Python report higher execution time in comparison to similar programs based on compiled languages. To address this issue and make Python a preferable option for data analysis, time-consuming subroutines could be compiled in "C" and "Fortran" and imported into Python in a way that resembles Python features.

Data Analysis or Machine Learning Libraries

Data Analysis libraries are sensitive routines and functions that are written in any given language. Software developers require a robust set of libraries to perform complex tasks without needing to rewrite multiple lines of code. Machine learning is largely based on mathematical optimization, probability, and statistics.

Python is the language of choice in the field of data analysis and machine learning credited to consistent development time and flexibility. It is well-suited to develop sophisticated models and production engines that can be directly plugged into production systems. One of its greatest assets being an extensive set of libraries that can help researchers who are less equipped with developer knowledge to easily execute data analysis and machine learning.

Django

According to the Django Software Foundation, “Django is a free and open-source, high-level Python Web framework that encourages rapid development and clean, pragmatic design. Built by experienced developers, it takes care of much of the hassle of Web development, so you can focus on writing your app without needing to reinvent the wheel”. The main objective of Django is to facilitate the development of sophisticated websites that are driven by databases. Its name is credited to the famous guitarist "Django Reinhardt" and was developed in late 2003 by computer scientists at the “Lawrence Journal World” newspaper, Adrian Holovaty and Simon Willison. In July 2005, “Django” was launched under a “BSD license” and was rolled up to the management of the "Django Software Foundation" in June 2008.

This framework promotes reusability and easy plugging in of the component, fewer codes, limited connection, faster development, and the no repetition principle. “Django” extensively uses Python for the development of configuration documents and data models. “Django” can be equipped with an optional administrative interface, which is dynamically developed by introspection and administrative model configurations to allow creating, reading, updating, and deleting files as needed. Several of the widely renowned websites are based on “Django,” such as "Public Broadcasting Service," "Instagram," "Mozilla," "Washington Times," "Disqus," "Bitbucket," and "Nextdoor."

Although the fundamental framework of "Django" has its naming conventions, like naming the objects that can be called and used to generate "views" of the "HTTP responses," it could still be considered a "model-template-view (MTV)" architectural pattern. It comprises of an "object-relational mapper (ORM)" that acts as a mediator between data models ("Python classes") and a relational database ("Model"), a system to process "HTTP requests" using a "web template system" or "View", and a standard expression driven "URL dispatcher" or "Controller".

The underlying framework also contains the features listed below:

- A "standalone and lightweight webserver" to develop and test the websites.
- A system to serialize and validate HTML forms, which is capable of translating between appropriate database storage values and these forms.
- A template system using the principle of inheritance as found in the "object-oriented programming."
- A "caching framework," which is capable of using a variety of caching techniques.
- Support for middleware classes, which are capable of intervening and performing custom tasks at different phases of request processing.
- An "internal dispatcher system" allowing application components to

relay occurrences to one another, through pre-defined signals.

- An internationalization system, which includes translations of various components of "Django" into a multitude of languages.
- A serialization system, which is capable of producing and reading "XML and/or JSON representations" of the "Django" model instances
- A system that allows extension of the template engine functionality.
- An interface to the integrated unit test framework of "Python."
- “Django REST Framework" constitutes a strong and adaptable "Web API construction" toolkit.

Scikit-Learn

“Scikit-Learn” has evolved as the gold standard for machine learning using Python, offering a wide variety of “supervised and unsupervised machine learning algorithms.” It is touted as one of the most user-friendly and cleanest machine learning libraries to date. For example, decision trees, clustering, linear and logistics regressions, and K-means. Scikit-learn uses a couple of basic Python libraries: NumPy and SciPy and adds a set of algorithms for data mining tasks, including classification, regression, and clustering. It is also capable of implementing tasks like feature selection, transforming data and ensemble methods in only a few lines.

In 2007, David Cournapeau developed the foundational code of "Scikit-Learn" as part of a "Summer of Code" project for "Google." Scikit-learn has become one of Python's most famous open-source machine learning libraries since its launch in 2007. But it wasn't until 2010 that Scikit-Learn was released for public use. Scikit-Learn is “an open-sourced and BSD licensed data mining and data analysis tool used to develop supervised and unsupervised machine learning algorithms” build on Python.” Scikit-learn offers various “machine learning algorithms” such as “classification,” “regression,” “dimensionality reduction,” and “clustering.” It also offers modules for feature extraction, data processing, and model evaluation.

Designed as an extension to the “SciPy” library, Scikit-Learn is based on “NumPy” and “matplotlib,” the most popular Python libraries. NumPy expands Python to support efficient operations on big arrays and multidimensional matrices. Matplotlib offers visualization tools and science computing modules are provided by SciPy. For scholarly studies, Scikit-Learn is popular because it has a well-documented, easy-to-use, and flexible API. Developers are able to utilize Scikit-Learn for their experiments with various algorithms by only altering a few lines of the code. Scikit-Learn also provides a variety of training datasets, enabling developers to focus on algorithms instead of data collection and cleaning. Many of the algorithms of Scikit-Learn are quick and scalable to all but huge datasets. Scikit-learn is known for its reliability, and automated tests are available for much of the library. Scikit-learn is extremely popular with beginners in machine learning to start implementing simple algorithms.

Prerequisites for application of Scikit-Learn library

The “Scikit-Learn” library is based on the “SciPy (Scientific Python),” which needs to be installed before using “SciKit-Learn.” This stack involves the following:

NumPy (Base n-dimensional array package)

“NumPy” is the basic package with Python to perform scientific computations. It includes, among other things: “a powerful N-dimensional array object; sophisticated (broadcasting) functions; tools for integrating C/C++ and Fortran code; useful linear algebra, Fourier transform, and random number capabilities.” The predecessor of NumPy called "Numeric" was initially developed by Jim Hugunin. In 2005, Travis Oliphant developed "NumPy" by integrating the functionalities of the "Numarray" into "Numeric" and making additional enhancements to it. NumPy is widely reckoned as an effective multi-dimensional container of generic data in addition to its apparent scientific uses. It is possible to define arbitrary data types. This enables NumPy to integrate with a wide variety of databases seamlessly and quickly. NumPy assists the "CPython reference implementation" of Python, which is a "non-optimizing bytecode interpreter." NumPy can partially address the issue of slow execution of mathematical algorithms, by offering multidimensional arrays, functions, and operators that work effectively on arrays by rewriting the code pertaining to the internal loops using NumPy.

Python bindings of "OpenCV's" commonly used computer vision library uses "NumPy arrays" for data storage and operation. Since pictures with various channels are merely depicted as 3-D arrays, indexing, slicing, or masking with other arrays are highly effective methods to access relevant pixels of the picture. The "NumPy array" as a universal data structure in "OpenCV" for pictures, extracted functionality points, filter kernels, and several other, to simplify the "programming workflow and debugging." The primary objective of NumPy is the homogeneity of the multidimensional array. It consists of an element table (generally numbers), all of which are of the same sort and are indicated by tuples of non-negative integers.

The dimensions of NumPy are called “axes," and the array class is called "ndarray."

These arrays are considered "stridden views on memory." Unlike the built-in list data structure of Python (also a dynamic array), the "NumPy arrays" can be typed uniformly, which means that "all the elements of a single array must be of the same type." Such arrays could also be "views of memory buffers assigned to the CPython interpreter by C / C++, Cython, and Fortran extensions without the need to copy data around," making them compatible with current numerical libraries. The "SciPy package" that incorporates a multitude of such libraries (particularly "BLAS" and "LAPACK") utilizes this capability. NumPy also offers built-in support for "memory-mapped ndarrays."

To develop “NumPy array” from “Python lists” while accessing elements, use the code below:

```
"import numpy as np

a = np.array([1, 2, 3])
print(type(a))
print(a.shape)
print(a[0], a[1], a[2])
a[0] = 5
print(a)

b = np.array([[1,2,3],[4,5,6]])
print(b.shape)
print(b[0, 0], b[0, 1], b[1, 0])"
```

Now, if you would like to index the “NumPy arrays,” you should start with slicing the multidimensional “array” into one dimension with the code below:

```
"import numpy as np


a = np.array([[1,2,3,4], [5,6,7,8], [9,10,11,12]])
b = a[:2, 1:3]
print(a[0, 1])
b[0, 0] = 77
print(a[0, 1]) "
```

This will result in a “sub-array” of the original “NumPy array” but if you would like to generate an “arbitrary array,” you can do so by utilizing "integer array indexing" which enables the generation of arbitrary arrays with the data from another array, as shown in the code below:

```
"import numpy as np


a = np.array([[1,2], [3, 4], [5, 6]])
print(a[[0, 1, 2], [0, 1, 0]])
print(np.array([a[0, 0], a[1, 1], a[2, 0]]))
print(a[[0, 0], [1, 1]])
print(np.array([a[0, 1], a[0, 1]]))"
```

Basic mathematical operations can be applied to arrays, as shown in the code below, and can be found in “NumPy” as “functions” and “operator overloads.”

```
"import numpy as np


x = np.array([[1,2],[3,4]], dtype=np.float64)
y = np.array([[5,6],[7,8]], dtype=np.float64)


print(x + y)
print(np.add(x, y))


print(x - y)
print(np.subtract(x, y))


print(x * y)
print(np.multiply(x, y))


print(x / y)
print(np.divide(x, y))
```

print(np.sqrt(x))"

Matplotlib (Comprehensive 2D/3D plotting)

“Matplotlib" is a 2-dimensional graphic generation library from Python that produces high-quality numbers across a range of hardcopy formats and interactive environments. The “Python script,” the “Python,” “IPython shells,” the “Jupyter notebook,” the web app servers, and select user interface toolkits can be used with matplotlib. Matplotlib attempts to further simplify easy tasks and make difficult tasks feasible. With only a few lines of code, you can produce tracks, histograms, scatter plots, bar graphs, error graphs, etc.

A MATLAB-like interface is provided for easy plotting of the Pyplot Module, especially when coupled with IPython. As a power user, you can regulate the entire line styles, font’s properties, and axis properties through an object-oriented interface or a collection of features similar to the one provided to MATLAB users.

SciPy (Fundamental library for scientific computing)

SciPy is a “collection of mathematical algorithms and convenience functions built on the NumPy extension of Python,” capable of adding more impact to interactive Python sessions, by offering high-level data manipulation and visualization commands and courses for the user. An interactive Python session with SciPy becomes an environment that rivals data processing and system prototyping technologies, including “MATLAB, IDL, Octave, R-Lab, and SciLab.”

Another advantage of developing "SciPy" on Python is the accessibility of a strong programming language in the development of advanced programs and specific apps. Scientific apps using SciPy benefit from developers around the globe, developing extra modules in countless software landscape niches. Everything produced has been made accessible to the Python programmer, from database subroutines and classes as well as "parallel programming to the web." These powerful tools are provided along with the "SciPy" mathematical libraries.

IPython (Enhanced interactive console)

“IPython (Interactive Python)” is an interface or command shell for interactive computing using a variety of programming languages. “IPython” was initially created exclusively for Python, which supports introspection, rich media, shell syntax, tab completion, and history. Some of the functionalities provided by IPython include: “interactive shells (terminal and Qt-based); browser-based notebook interface with code, text, math, inline plots, and other media support; support for interactive data visualization and use of GUI tool kits; flexible interpreters that can be embedded to load into your own projects; tools for parallel computing”. The architecture of "IPython" offers "parallel and distributed computing." IPython" allows the development, execution, debugging, and interactive monitoring of parallel applications, thus the "I (Interactive) in IPython." The underlying architecture can easily separate parallelism, allowing "IPython" to assist with multiple parallelism styles including: "Single program, various information (SPMD) parallelism," "Multiple programs, various data (MIMD) parallelism," "Message passing using MPI," "Task parallelism," "Data parallelism," combinations of these methods and even customized user-defined strategies.

The parallel computing functionality has been rendered optional under the "ipyparallel python package," with the implementation of "IPython 4.0".

“IPython" often derives from "SciPy stack libraries" such as "NumPy" and "SciPy," frequently installed in combination with one of the various “Scientific Python distributions." IPython" can also be integrated with select "SciPy stack libraries," primarily "matplotlib," which produces inline charts upon use with the "Jupyter notebook." For customization of rich object display, Python libraries can be implemented with "IPython-specific hooks." For instance, if used in the context of "IPython," "SymPy" can implement "rendering of mathematical expressions as rendered LaTeX."

SymPy (Symbolic mathematics)

Developed by Ondřej Čertík and Aaron Meurer, SymPy is "an open-source Python library for symbolic computation." It offers algebra computing abilities to other apps as a stand-alone app and/or as a library as well as live on the internet applications with “SymPy Live” or “SymPy Gamma.” “SymPy” is easy to install and test, owing to the fact that it is completely developed in Python boasting limited dependencies. SymPy involves characteristics ranging from calculus, algebra, discrete mathematics, and quantum physics to fundamental symbolic arithmetic. The outcome of the computations can be formatted as the "LaTeX" code. In combination with a straightforward, expandable codebase in a widespread programming language, the ease of access provided by SymPy makes it a computer algebra system with a comparatively low entry barrier.

Pandas (Data structures and analysis)

Pandas provide highly intuitive and user-friendly high-level data structures. "Pandas" has achieved popularity in the machine learning algorithm developer community, with built-in techniques for data aggregation, grouping, and filtering as well as results of time series analysis. The Pandas library has two primary structures: one-dimensional "Series" and two-dimensional "Data Frames."

Some of the key features provided by “Pandas” are listed below:

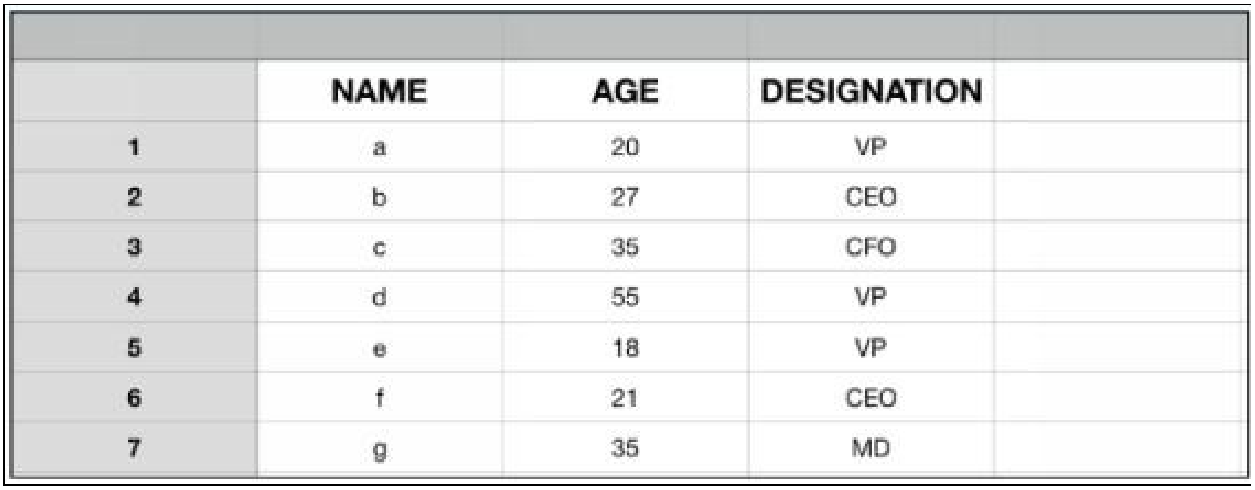
- A quick and effective "Data Frame object" with embedded indexing to be used in data manipulation activities.
- Tools to read and write data between internal memory data structures and multiple file formats, such as "CSV" and text, "Microsoft Excel," "SQL databases," and quick "HDF5 format".
- Intelligent data alignment and integrated management of incomplete data by achieving automatic label driven computational alignment and readily manipulating unorganized data into an orderly manner.
- Flexible reconstructing and pivoting of datasets.
- Smart label-based slicing and indexing of big data sets, as well as the creation of data subsets.
- Columns can be added to and removed from data structures to achieve the desired size of the database.
- Aggregation or transformation of data using a sophisticated "Group By" system enabling execution of the "split-apply-combine" technique on the data.
- Highly efficient merge and join functions of the data set.

- "Hierarchical axis indexing" offers a simple way to work in a low dimensional data structure even with high dimensional data.
- Time-series functionalities, including "date range generation and frequency conversion, moving window statistics, moving window linear regressions, date shifting, and lagging." Also, the creation of "domain-specific time offsets" and the capability of joining time series with no data loss.
- Having most of the underlying code in "Cython" or "C," Pandas boasts high performance and efficiency.
- Python, in combination with Pandas is being used in a broad range of academic and industrial sectors, including Financial Services, Statistics, Neurobiology, Economics, Marketing and Advertising, Online Data Analytics, among others.

The two types of Data Structures offered by Pandas are: “Pandas DataFrame” and “Pandas Series.”

Pandas DataFrame

It is defined as “2-D labeled data structure with columns of a potentially different type”. It has a high resemblance to the Excel spreadsheet, as shown in the picture below, with multiple similar features for analysis, modification, and extraction of valuable insights from the data. You can create a “Pandas DataFrame” by entering datasets from “Excel,” “CSV,” and “MySQL database,” among others.



	NAME	AGE	DESIGNATION	
1	a	20	VP	
2	b	27	CEO	
3	c	35	CFO	
4	d	55	VP	
5	e	18	VP	
6	f	21	CEO	
7	g	35	MD	

For instance, in the picture above, assume “Keys” are represented by the name of the columns, and “Values” are represented by the list of items in that column. A “Python dictionary” can be used to represent this as shown in the code below:

"my_dict = {

'name' : ["a", "b", "c", "d", "e","f", "g"],

'age' : [20,27, 35, 55, 18, 21, 35],

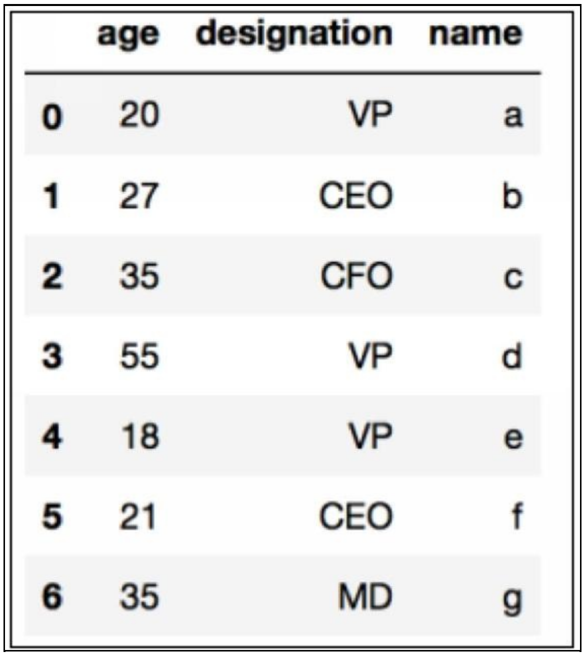
'designation': ["VP", "CEO", "CFO", "VP", "VP", "CEO", "MD"]}"

The “Pandas DataFrame” can be created from this dictionary by using the code below:

"import Pandas as PD

df = pd.DataFrame(my_dict)"

The resulting “DataFrame” is shown in the picture below which resembles the excel spreadsheet:



	age	designation	name
0	20	VP	a
1	27	CEO	b
2	35	CFO	c
3	55	VP	d
4	18	VP	e
5	21	CEO	f
6	35	MD	g

If you would like to define index values for the rows, you will have to add the “index” parameter in the “DataFrame ( )” clause as shown below:

```
"df = pd.DataFrame(my_dict, index=[1,2,3,4,5,6,7])"
```

To obtain “string” indexes for the data instead of numeric, use the code below:

```
"df = pd.DataFrame(
    my_dict,
    index=["First", "Second", "Third", "Fourth", "Fifth", "Sixth", "Seventh"])"
```

Now, as these index values are uniform, you could execute the code below to utilize the “NumPy arrays” as index values:

```
"np_arr = np.array([10,20,30,40,50,60,70])
df = pd.DataFrame(my_dict, index=np_arr)"
```

Similar to “NumPy”, the columns of “DataFrame” are also homogeneous. You can use dictionary like syntax or add the column name with “DataFrame” to view the data type of the column as shown in the code below:

```
"df['age'].dtype    # Dict Like Syntax
df.age.dtype       # DataFrame.ColumnName
df.name.dtype      # DataFrame.ColumnName"
```

You can use the code below to selectively view the record or rows available within the “Pandas DataFrame” by using the “head ( )” function for the first five rows and “tail ( )” function for the last five rows. For example, to see the data’s first three rows, you can use the code below:

```
"df.head(3)   # Display first 3 Rows"
```

Pandas Series

It can be defined as a "one-dimensional labeled array capable of holding data of any type (integer, string, float, python objects)." Simply put, it is like a column in an excel spreadsheet. To generate a “Pandas Series” from an array, a “NumPy” module must be imported and used with “array ()” function as shown in the code below:

```
"# import pandas as pd
import pandas as pd"
```

```
"# import numpy as np
import numpy as np"
```

```
"# simple array
```

```
data = np.array (['m','a','c','h','I','n','e'])"
```

```
"ser = pd.Series(data)
print(ser)"
```

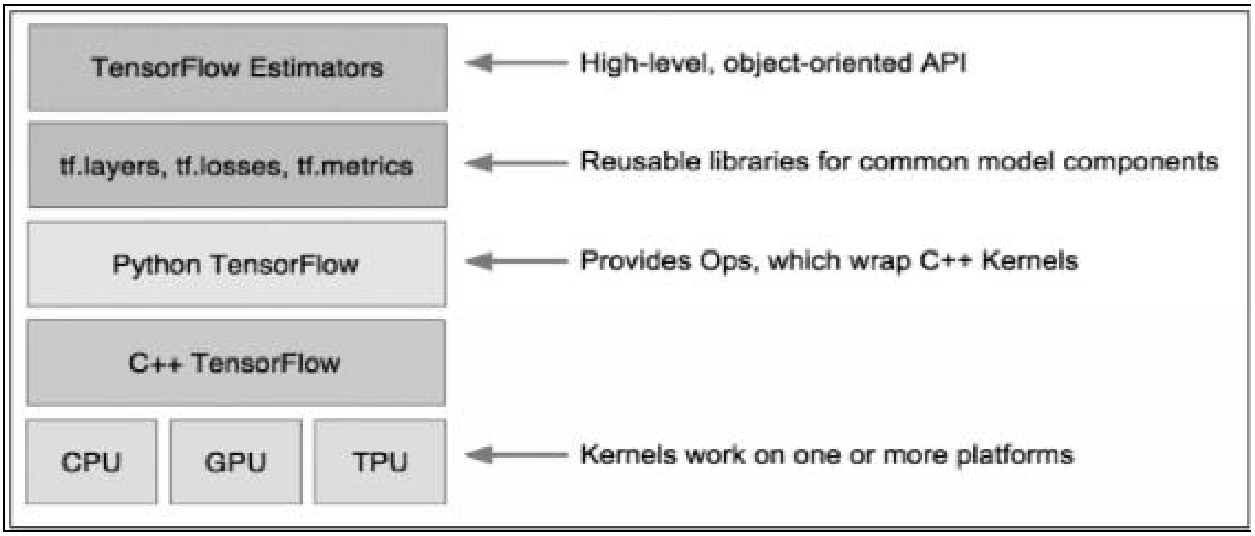
Seaborn (data visualization)

Seaborn is derived from the Matplotlib Library and an extremely popular visualization library. It is a high-level library that can generate specific kinds of graph including heat maps, time series, and violin plots.

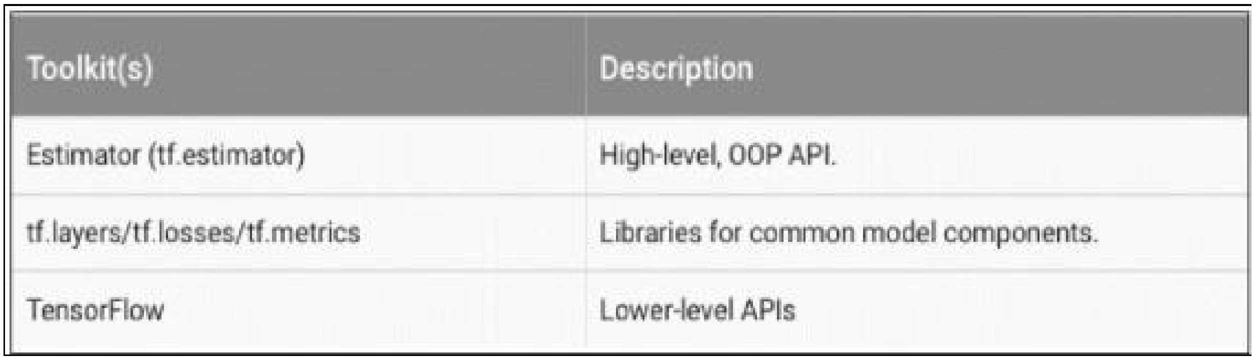
TensorFlow

TensorFlow can be defined as a Machine Learning platform providing end-to-end service with a variety of free and open sources. It has a system of multilayered nodes that allow for quick building, training, and deployment of artificial neural networks with large data sets. It is touted as a “simple and flexible architecture to take new ideas from concept to code to state-of-the-art models and to publication at a rapid pace.” For example, Google uses TensorFlow libraries in its image recognition and speech recognition tools and technologies.

Higher-level APIs such as “tf.estimator” can be used for specifying predefined architectures, such as “linear regressors” or “neural networks.” The picture below shows the existing hierarchy of the TensorFlow tool kit:


TensorFlow Estimators
High-level, object-oriented API
tf.layers, tf.losses, tf.metrics
Reusable libraries for common model components
Python TensorFlow
Provides Ops, which wrap C++ Kernels
C++ TensorFlow
CPU
GPU
TPU
Kernels work on one or more platforms

The picture shown below provides the purposes of the different layers:



Toolkit(s)	Description
Estimator (tf.estimator)	High-level, OOP API.
tf.layers/tf.losses/tf.metrics	Libraries for common model components.
TensorFlow	Lower-level APIs

The two fundamental components of TensorFlow are:

1. A “graph protocol buffer”
2. A “runtime” that can execute the graph

The two-component mentioned above are similar to the “Python” code and the “Python interpreter." Just as the "Python interpreter” can run Python code on several hardware systems, TensorFlow can be operated on various hardware systems, like CPU, GPU, and TPU.

To make a decision regarding which API(s) should be used, you must consider the API offering the highest abstraction level to solve the target problem. Easier to use, but (by design) less flexible, are the greater abstract levels. It is recommended to first begin with the highest-level API and make everything work. If for certain unique modeling issues, you need extra flexibility, move one level down. Notice that each level is constructed on the lower-level APIs. It should thus be quite simple to decrease the hierarchy.

For the development of the majority of Machine Learning models, we will use “tf.estimator” API, which significantly lowers the number of code lines needed for development. Also, “tf.estimator” is compatible with Scikit-Learn API.

Chapter 3: Predictive Modeling, Data Visualization and Creation of Training Data Set

Machine Learning

Artificial intelligence is the manifestation of the idea that machines are capable of human-like intelligence and can mimic human thought processing and learning capabilities to adapt to new inputs and perform tasks without requiring human assistance. Machine learning is integral to the concept of artificial intelligence. Machine learning can be defined as a concept of artificial intelligence technology that focuses primarily on the engineered capability of machines to learn and self-train, by identifying data patterns to improve upon the underlying algorithm and make independent decisions. Machine learning hypothesizes that modern-day computers have an ability to be trained using a training data set that can be easily customized to meet desired functionalities. Machine learning is driven by the pattern recognition technique wherein the machine records and revisit past interactions and results that are deemed in alignment with its current situation. Given the fact that machines are required to process the endless amount of data, with new data always pouring in, they must be equipped to adapt to the new data without needing to be programmed by a human speaks to the iterative aspect of machine learning.

Machines are capable of learning from and self-training by utilizing previous computations and underlying algorithms to produce high-quality decisions and results that are easily reproducible. The concept of machine learning has been around for decades, but recent advancements in machine learning algorithms have made it possible for the machines to process and analyze big data. This is accomplished by the application of advanced and complex mathematical calculations using automation at high speed and frequency.

Advanced computing machines of today are capable of rapidly analyzing the humongous amount of data and delivering faster and more accurate results. Companies that are employing machine learning algorithms have enhanced flexibility to modify the training data set to meet their business requirements and train the machines accordingly. These customized machine learning algorithms allow businesses to identify potential risks as well as growth opportunities. Machine learning algorithms are typically used in collaboration with artificial intelligence technology and cognitive technologies to make the machines highly effective and efficient in processing large volumes of data or big data and produce highly accurate results.

Given the intrinsic link between artificial intelligence and machine learning, going forward to make this book easier to understand, we will be using the terms “machine learning” and “artificial intelligence” interchangeably.

There are four types of machine learning algorithms available today:

Supervised Machine Learning Algorithms

These algorithms are widely used in predictive big data analysis, owing to its capability of assessing and applying the lessons learned from previous iterations and interactions to new input data set. These algorithms can label all their ongoing runs based on the instructions provided to successfully predict and forecast future events. For example, humans can program the machine to label its data points as "R" (Run), "N" (Negative), or "P" (Positive). The machine learning algorithm will then label the input data as programmed and receive data inputs with corresponding correct outputs. The algorithm will run a comparison of its own generated output against the "expected or correct" output, to identify potential improvements that can be made and errors that can be fixed to make the model more accurate and smarter.

By applying methods like "regression," "prediction," "classification," and "ingredient boosting" to well train the machine learning algorithms, any new input data can be fed into the machine as "target" data set to orchestrate the learning program as desired. This "known training data set" jump-starts the analytical process, which is followed by the learning algorithm to produce and "inferred function" that can be used to generate forecasts and predictions for future events based on the output values. For example, financial institutions and banks are heavily dependent on supervised machine learning algorithms to detect fraudulent credit card transactions and predict the likelihood of a potential credit card customer failing to make their credit payments on time.

Unsupervised Machine Learning Algorithms

Companies often run into a situation where data sources required to generate a labeled and classified training data set are unavailable. In these situations, the use of unsupervised machine learning algorithms is ideal. Unsupervised machine learning algorithms are used to identify ways in which machines can create "inferred functions" to elucidate a hidden structure from the pile of the unlabeled and unclassified data set. These algorithms are capable of exploring the data to identify a structure within the mass of information.

Unlike the supervised machine learning algorithms, the unsupervised algorithms fail out to identify the correct output, although they are just as efficient at exploring the input data and drawing inferences as the supervised learning algorithms. These algorithms can be used for identification of data outliers, generation of customized and personalized product recommendations, and classification of text topics using techniques like "self-organizing maps," "singular value decomposition," and "k-means clustering." For example, the identification of customers with shared shopping attributes that can be segmented into specific groups and targeted with similar marketing strategies and campaigns. As a result, the unsupervised learning algorithms are extremely popular in the online marketing space.

Semi-Supervised Machine Learning Algorithms

The semi-supervised machine learning algorithms are highly versatile and capable of utilizing labeled as well as unlabeled data set to learn from and train themselves. These algorithms are a “hybrid” of supervised and unsupervised learning algorithms. Typically, the training data set is composed of predominantly unlabeled data with a small amount of label data. The use of analytical methods including "prediction," "regression," and "classification" in combination with semi-supervised learning algorithms allows the Machine to significantly improve its learning accuracy and training abilities.

These algorithms are widely used in cases where generating labeled training data set from raw data highly resource-intensive and less cost-effective for the company. So to avoid additional personnel and equipment cost, companies use semi-supervised learning algorithms on their systems. For example, the facial recognition application requires a large volume of facial data spread across multiple input sources. The preprocessing, processing, classification, and labeling of the raw data obtained from sources like web cameras, requires a lot of manpower, and thousands of hours of work in order to be used as a training data set.

Reinforcement Machine Learning Algorithms

The reinforcement machine learning algorithms are much more unique in that it learns from the environment. These algorithms perform actions and diligently record the results of each action that would have either been a failure resulting in an error or reward for successful execution. The two main characteristics that distinguish the reinforcement learning algorithms are "trial and error" research method and "delayed reward." The machine repeatedly analyzes input data by using a variety of calculations and sending a reinforcement signal for every correct or expected output, to eventually optimize the end result. Simple action and reward feedback loop are developed by the system to assess, record, and learn which actions were successful and led to accurate results in a shorter period of time. The use of these algorithms allows the automatic determination of ideal behaviors within the constraints of a specific context by the system to, and hence, its capabilities and maximize its performance. As a result, the reinforcement machine learning algorithms are heavily used in the gaming industry, robotics engineering as well as in navigation systems.

Big Data Analytics vs. Machine Learning

“Machine learning,” “Big Data,” as well as “Big Data Analytics” fall under the umbrella of “Artificial Intelligence” technology, which rolls up into the field of “Data Science.” Here are some key differences between machine learning and big data or big data analytics:

Applications

Machine learning is used in the development of advanced recommendation engines and models that can predict and forecast future events by analyzing and existing data. For example, "Google" is using advanced machine learning algorithms and the development of their self-driving cars. Virtual personal and home assistance devices such as "Amazon Alexa," "Google Home," and "Apple HomePod" are all driven by advanced machine learning algorithms, working in close collaboration with artificial intelligence technology.

Big data has much wider applicability and can be used for general research purposes to gather information regarding specific business queries such as collecting sales data, building consumer profiles, and financial research, among other applications across industrial domains.

Learning capability

Machine learning algorithms are self-sufficient and capable of learning from a training data set as well as new input data without human assistance. They are powerful enough to create the foundation required for the system to learn and improve upon itself. Big data analytics is capable of gathering existing data and analyzing it for the identification of emerging patterns that can be used in its decision-making process. It is always powered by human-controlled analytical tools and technologies with no scope of self-learning.

Pattern recognition

Big data analytics utilizes statistical methods like regression, clustering, and classification, to recognize data patterns that can be analyzed to produce logical information.

Machine learning can not only recognize emerging data patterns but apply advanced algorithms on these patterns to learn from the data patterns and maximize the system performance with every successive iteration.

Data volume and Data type

Big data pertains to the humongous amount of unstructured, semi-structured, and structured data. Advanced analytical tools and technologies are required to process these large data sets with a majority of unstructured data to gather valuable information.

Machine learning utilizes relatively smaller data sets that are classified and labeled to serve as guiding instructions for the algorithms to learn and improve upon itself.

Fundamental Purpose

The main goal of machine learning is self-improvement solely on the basis of input data with little to no human assistance. The promising use of these algorithms and the development of smart machines could someday assist in providing answers to daunting challenges facing humanity, such as global warming.

Big data is designed to collect and store the skyrocketing data generated by our evolving digital lives. The big data analytics uses pattern recognition technology to uncover hidden patterns and insight that can be easily lost in this mass of information.

Customer Analytics

According to SAS, customer analytics can be defined as “processes and technologies bad gives organizations the customer insight necessary to deliver offers that are anticipated, relevant and timely.” Customer analytics is at the heart of all marketing activities and is an umbrella term used for techniques such as “predictive modeling,” “data visualization,” “information management,” and “segmentation.”

Importance of Customer Analytics

Customer analytics has evolved as the backbone of the marketing industry. This is a direct result of the advent of "smart consumer" who is more aware and connected to one another than ever before and willing to take their business elsewhere at a moment's notice. The smart customer has seamless access to a variety of information including the best products and services available in the market and where can they find the best deals to make the most of their money. Therefore, companies are required to be proactive and be able to predict consumer behavior when interacting with their products and to be in a position to take the required action to convert the prospective customer into a paying client. To generate more accurate forecasts and predictions of customer behavior, companies must have a solid understanding of their customers' buying habits and lifestyle choices. These near accurate predictions will provide the company with an edge over the competition and help achieve higher conversion rates in their sales and marketing funnel.

One of the best customer analytics solution in the market today is “SAS Customer Intelligence,” which claims to have the following applications:

- Achieve higher customer loyalty and response rates.
- Generate personalized customer offers and messages to reach the right customer at the right time.

- Identify prospective customers with similar attributes and a high likelihood of conversion so the company can reduce costs on their targeted marketing strategies and campaigns.
- Reduce customer attrition by generating accurate predictions on customers that are more likely to take their business somewhere else and developing proactive marketing campaigns to retain them.

“*The insights derived from our new analytics capabilities are allowing us to find the sweet spots that will continue to drive loyalty, profitability, and sustainable growth.”*
- *Carrie Gray, Executive Director for Medium Business Marketing, Verizon*

Predictive Analytics Marketing

According to SAS, predictive analytics is the “*use of data, statistical algorithms, and machine learning techniques to identify the likelihood of future outcomes based on historical data. The goal is to go beyond knowing what has happened to provide the best assessment of what will happen in the future"*. Today, companies are digging through their past with an eye on the future, and this is where artificial intelligence for marketing comes into play, with the application of predictive analytics technology. The success of predictive analytics is directly proportional to the quality of big data collected by the company.

Here are some of the widely used predictive analytics applications for marketing:

Predictive Analysis for Customer Behavior

For the industrial giants like "Amazon," "Apple," and "Netflix," analyzing customer activities and behavior is fundamental to their day-to-day operations. Smaller businesses are increasingly following in their footsteps to implement predictive analysis in their business model. The development of a customized suite of predictive models for a company is not only capital-intensive but also requires extensive manpower and time. Marketing companies like "AgilOne" offer relatively simple predictive model types with wide applicability across industrial domains. They have identified three main types of predictive models to analyze customer behavior, which are:

“Propensity models” – These models are used to generate “true or accurate” predictions for customer behavior. Some of the most common propensity models include: “predictive lifetime value,” “propensity to buy,” “propensity to turn,” “propensity to convert,” “likelihood of engagement,” and “propensity to unsubscribe.”

“Cluster models” – These models are used to separate and group customers based on shared attributes such as gender, age, purchase history, and demographics. Some of the most common cluster models include "product-based or category base clustering," "behavioral customs clustering," and "brand based clustering."

“Collaborative filtering” – These models are used to generate products and services and recommendations as well as to recommended advertisements based on prior customer activities and behaviors. Some of the most common collaborative filtering models include "upsell," "cross-sell," and "next sell" recommendations.

The most significant tool used by companies to execute predictive analytics on customer behavior is "regression analysis," which allows the company to establish correlations between the sale of a particular product and the specific attributes displayed by the purchasing customer. This is achieved by employing "regression coefficients," which are numeric values depicting the degree to which the customer behavior is affected by different variables and developing a "likelihood score" for the future sale of the product.

Predictive Data Analysis using Scikit-Learn library

To understand how Scikit-Learn library is used in the development of the “Predictive Data Analysis” or machine learning model, let us use the “Sales_Win_Loss data set from IBM’s Watson repository” containing data obtained from sales campaign of a wholesale supplier of automotive parts. We will build a machine learning model to predict which sales campaign will be a winner and which will incur a loss.

The data set can be imported using Pandas and explored using Pandas techniques such as “head (), tail (), and dtypes ().” The plotting techniques from “Seaborn” will be used to visualize the data. To process the data Scikit-Learn’s “preprocessing.LabelEncoder ()” will be used and “train_test_split ()” to divide the data set into a training subset and testing subset.

To generate predictions from our data set, three different algorithms will be used, namely, “Linear Support Vector Classification and K-nearest neighbor classifier.” To compare the performances of these algorithms, Scikit-Learn library technique, “accuracy_score,” will be used. The performance score of the models can be visualized using “Yellowbrick” visualization and Scikit-Learn.

Installing Scikit-Learn

The latest version of Scikit-Learn can be found on “Scikit-Learn.org” and requires “Python (version >= 3.5); NumPy (version >= 1.11.0); SciPy (version >= 0.17.0); joblib (version >= 0.11)”. The plotting capabilities or functions of Scikit-learn start with “plot_” and require “Matplotlib (version >= 1.5.1)”. Certain Scikit-Learn examples may need additional applications: “Scikit-Image (version >= 0.12.3), Pandas (version >= 0.18.0)”.

With the prior installation of "NumPy" and "SciPy,” the best method of installing Scikit-Learn is using “pip: pip install -U scikit-learn” or “conda: conda install scikit-learn.”

One must make sure that “binary wheels” are utilized when using pip and that “NumPy” and “SciPy” have not been recompiled from source, which may occur with the use of specific OS and hardware settings (for example, “Linux on a Raspberry Pi”). Developing “NumPy” and “SciPy” from source tends to be complicated (particularly on Windows); therefore, they need to be set up carefully, making sure the optimized execution of linear algebra routines is achievable.

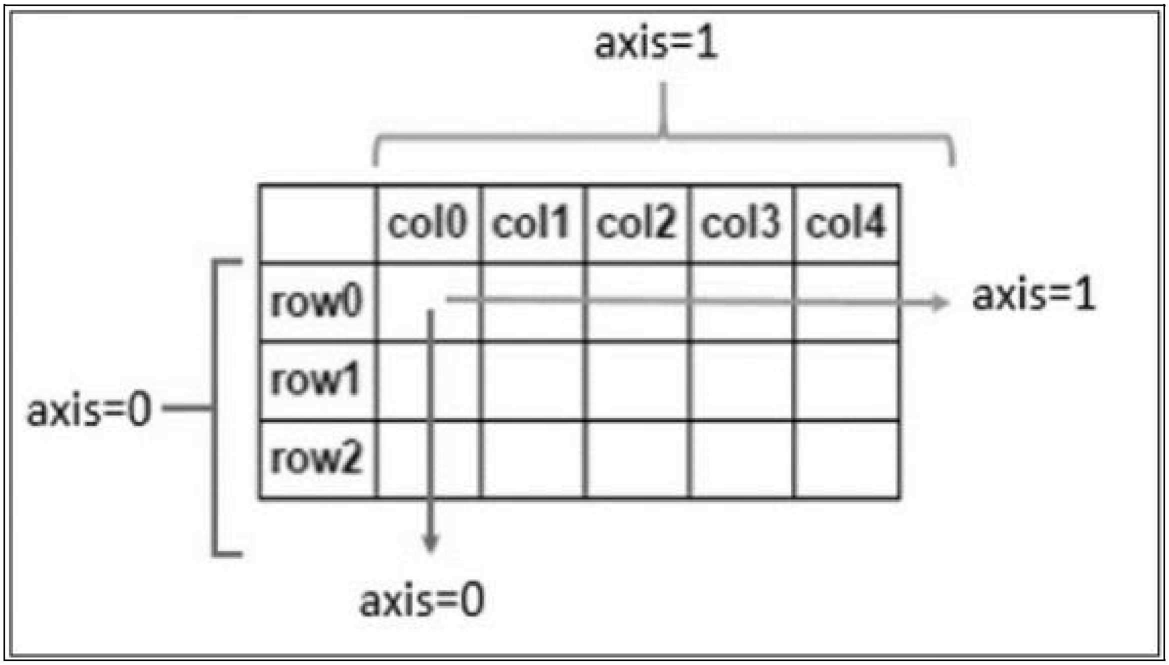
Importing the Data Set

To import the “Sales_Win_Loss data set from IBM’s Watson repository," the first step is importing the "Pandas" module using "*import pandas as pd*.”

Then, we leverage a variable URL as *“https://community.watsonanalytics.com/wp content/uploads/2015/04/WA_Fn-UseC_-Sales-Win-Loss.csv”* so that the data set can be stored and downloaded on the URL.

Now, *“read_csv() as sales_data = pd.read_csv(url)”* technique will be used to read the above "CSV or comma-separated values" file, which is supplied by the Pandas module. The CSV file will then be converted into a Pandas data framework with the return variable as "*sales_data*,” where the framework will be stored.

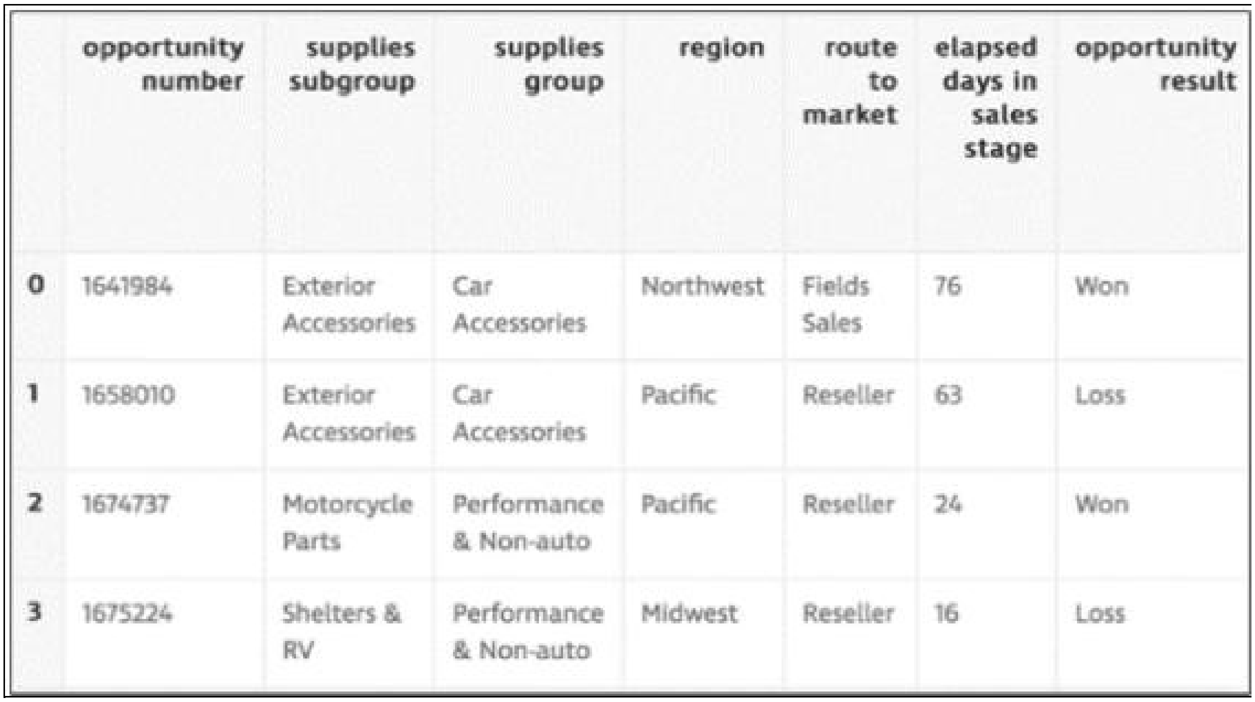
For new ‘Pandas’ users, the *“pd. read CSV()"* technique in the code mentioned above will generate a tabular data structure called “data framework”, where an index for each row is contained in the first column, and the label/name for each column in the first row are the names taken from the data set located in the initial column. In the above code snippet, the *“sales data”* variable results in a table depicted in the picture below.



In the above diagram, the individual column is represented by the “row0, row1, row2,” and the names for the features of the data set or individual columns are represented by “col0, col1, col2".

With this step, you have successfully stored a copy of the data set and transformed it into a "Pandas" framework!

Now, using the *“head() as Sales_data.head()”* technique, the records from the data framework can be displayed as shown below to get a “feel” of the information contained in the data set.



	opportunity number	supplies subgroup	supplies group	region	route to market	elapsed days in sales stage	opportunity result
0	1641984	Exterior Accessories	Car Accessories	Northwest	Fields Sales	76	Won
1	1658010	Exterior Accessories	Car Accessories	Pacific	Reseller	63	Loss
2	1674737	Motorcycle Parts	Performance & Non-auto	Pacific	Reseller	24	Won
3	1675224	Shelters & RV	Performance & Non-auto	Midwest	Reseller	16	Loss

Data Exploration

We can quickly explore the data to understand what information can tell can be gathered from it and accordingly to plan a course of action.

In any machine learning project, data exploration tends to be a very critical phase. Even a fast data set exploration can offer us significant information that could be easily missed otherwise, and this information can propose significant questions that we can then attempt to answer using our project.

Some third-party Python libraries will be used here to assist us with the processing of the data so that we can efficiently use this data with the powerful algorithms of Scikit-Learn. The same *“head()”* technique that we used to see some initial records of the imported data set in the earlier section can be used here. As a matter of fact, *“(head)*” is effectively capable of doing much more than displaying data records and customize the “head()” technique to display only a selected record with commands like *“sales_data.head(n=2)”*. This command will selectively display the first 2 records of the data set. At a glance, it's obvious that string data is contained in columns such as “Supplies Group” and “Region,” while columns such as “Opportunity Resultthe “Opportunity Number,” et cetera. are comprised of integer values. It can also be seen that there are unique identifiers for each record in the' Opportunity Number' column.

Similarly, to display select records from the bottom of the table, the *“tail() as sales_data.tail()”* can be used.

To view the different data types available in the data set, the Pandas technique *“dtypes() as sales_data.dtypes”* can be used. With this information, the data columns available in the data framework can be listed with their respective data types. We can figure out, for example, that the “object” data type is in the column “Supplies Subgroup” and that the “integer data type” is in the column “Client Size By Revenue.” So, we have an understanding of columns that either contains integer values or string data.

Data Visualization

At this point, we are through with basic data exploration steps, so we will not attempt to build some appealing plots to portray the information visually and discover other concealed narratives from our data set.

Of all the available Python libraries providing data visualization features; “Seaborn” is one of the best available options, so we will be using the same. Make sure that python plots module provided by “Seaborn” has been installed on your system and ready to be used. Now, follow the steps below generate the desired plot for the data set:

Step 1 – Transfer the module "Seaborn" with the command *“import seaborn as sns.”*

Step 2 – Transfer the module “Matplotlib” with command “*import* *matplotlib.pyplot as plt."*

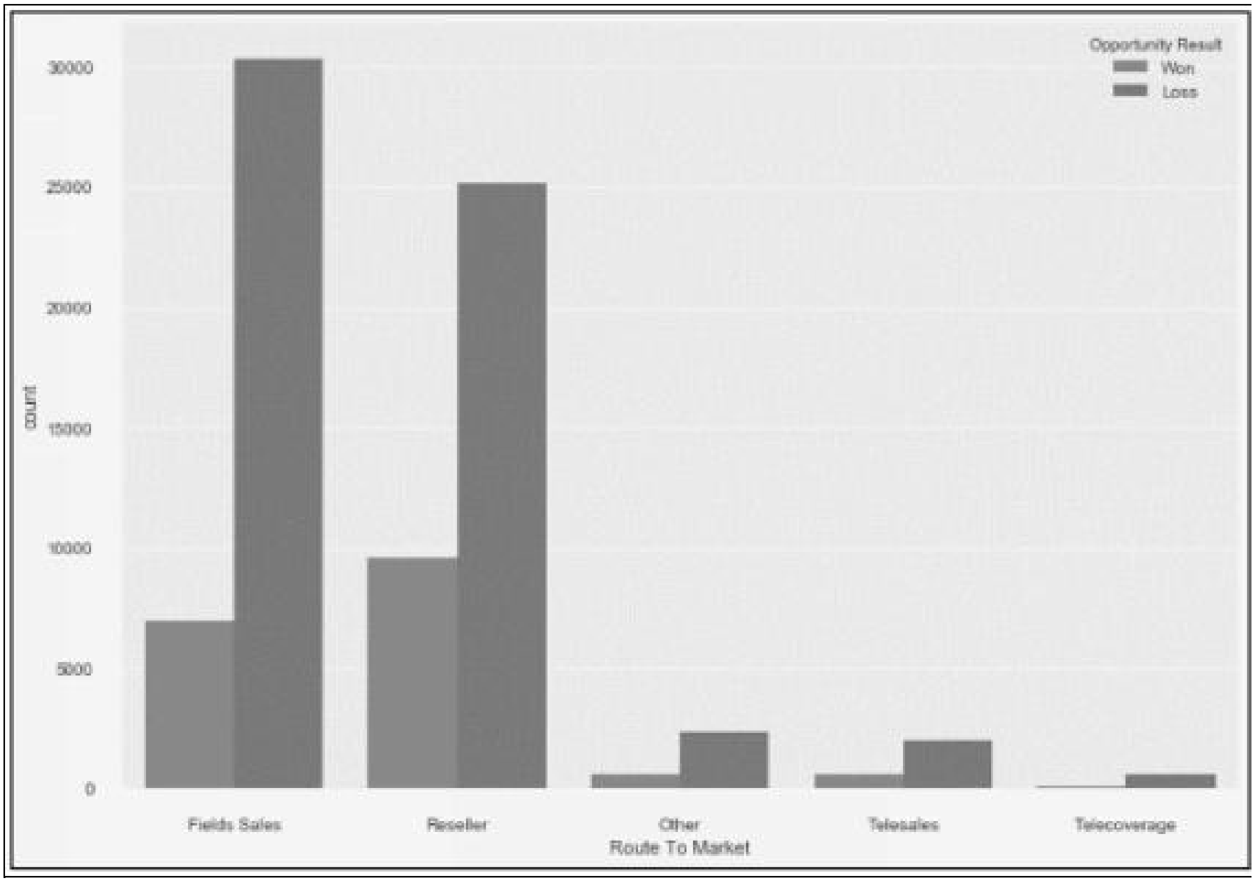
Step 3 - Use the command *"sns.set(style="whitegrid", color_codes=True)"* in setting to white the "background color" of the plot.

Step 4 - To set the "plot size" for all plots, use command *"sns.set(rc= {'figure.figsize':(11.7,8.27)})"*.

Step 5 – To generate a "countplot", use command *"sns.countplot('Route To Market', data=sales_data, hue = 'Opportunity Result')"*.

Step 6 – To eliminate the bottom and top margins, use command *"sns.despine(offset=10, trim=True)"*.

Step 7 – To display the plot, use the command *"plotplt.show()."*



Quick recap - The “Seaborn” and “Matplotlib” modules were imported first. Then the *“set()”* technique was used to define the distinct characteristics for our plots, such as plot style and color. The background of the plot was defined to be white using the code snippet

“sns.set(style= “whitegrid”, color codes= True)”.

Then the plot size was define using command *“sns.set(rc={'figure.figsize':*

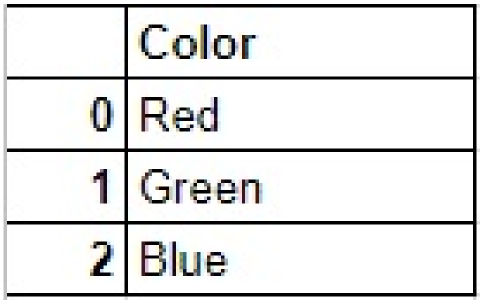
(11.7,8.27)})" that define the plot's size as "11.7px and 8.27px".

Next, the command *"sns.countplot('Route To Market',data= sales data, hue='Opportunity Result')"* was used to generate the plot. The "countplot()" technique enables the creation of a count plot, which can expose multiple arguments to customize the count plot according to our requirements. As part of the first *"countplot()"* argument, the X-axis was defined as "Route To Market" column from the data set. The next problem concerns the origin of the data set, which would be the "sales_data" data framework we imported earlier. The third argument is the color of the bar graphs that were defined as "blue" representing the "won" column and "green" for the "loss" column."

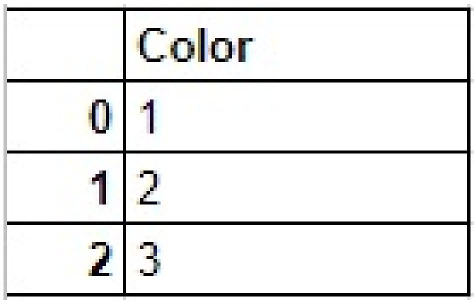
Data Pre-processing

By now, you should have a clear understanding of what information is available in the data set. To transform categorical labels from the data set such as "won" and "loss" into numerical values, we will use the *"LabelEncoder()"* technique.

Let's look at the pictures below to see what we are attempting to accomplish with the *"LabelEncoder()"* technique. The first image contains "color" column with 3 records, namely, "Red," "Green," and "Blue." Using the *"LabelEncoder()"* technique, the record in the same "color" column can be converted to numerical values, as shown in the second image.



	Color
0	Red
1	Green
2	Blue



	Color
0	1
1	2
2	3

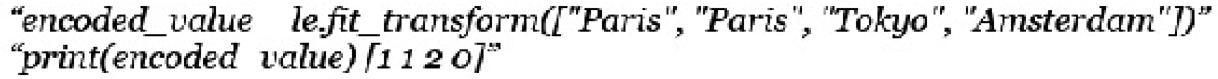
Let's begin the real process of conversion now. Using the *“fit transform()”* technique given by *“LabelEncoder(),”* the categorical column’s label like “Route To Market” can be encoded and converted to numerical labels comparable to those shown in the diagrams above. The function *“fit transform()”* requires input labels identified by the user and consequently returns encoded labels.

To know how the encoding is accomplished, let's go through an example quickly. The code instance below constitutes string data in the form of a cities’ list such as ["Paris," "Paris,” "Tokyo," "Amsterdam"] that will be encoded into something comparable to “[2, 2, 1,3]”.

Step 1 - To import the required module, use the command *“from sklearn import preprocessing.”*

Step 2 – To create the Label encoder object, use command *“le = preprocessing.LabelEncoder()”*.

Step 3 – To convert the categorical columns into numerical values, use command:



```
“encoded_value    le.fit_transform(["Paris", "Paris", "Tokyo", "Amsterdam"])”
“print(encoded  value) [1 1 2 0]”
```

We just transformed our labels for string data into numerical values. The first step was importing the preprocessing module that offers the *“LabelEncoder()”* technique. Followed by the development of an object representing the *“LabelEncoder()”* type. Then, the *“fit_transform()”* function of the object was used to distinguish between the list’s distinct classes ["Paris," "Paris," "Tokyo,” "Amsterdam,"] and output the encoded values of “[1 1 20]”.

The technique of *“LabelEncoder()”* assigns the numerical values with regards to the classes’ initial letter, alphabetically to the classes, for example “(A)msterdam” was assigned code “0”, “(P)aris” was assigned code “1” and “(T)okyo” was assigned code “2”.

Creating Training and Test subsets

To know the interactions between distinct characteristics and how these characteristics influence the target variable, a collection of information must be learned by a machine learning algorithm. We need to split the complete data set into two subsets to accomplish this. One subset will serve as the training data set, which will be used to train our algorithm to construct machine learning models. The other subset will serve as the test data set, which will be used to test the accuracy of the predictions generate by the machine learning model.

The first phase in this stage is the separation of feature and target variables using the steps below:

Step 1 – To select data excluding select columns, use command *“select columns other than 'Opportunity Number', 'Opportunity Result'cols = [col for col in sales_data.columns if col not in ['Opportunity Number','Opportunity Result']]”*.

Step 2 – To drop these select columns, use the command *“dropping the 'Opportunity Number’ and 'Opportunity Result' columns*
data = sales_data[cols]”.

Step 3 – To assign the Opportunity Result column as “target”, use command *“target = sales_data['Opportunity Result']*
data.head(n=2)”.

The “Opportunity Number” column was removed since it just acts as a unique identifier for each record. The “Opportunity Result” contains the predictions we want to generate, so it becomes our “target” variable and can be removed from the data set for this phase. The first line of the above code will select all the columns except “Opportunity Number” & “Opportunity Result” in. These columns are then assigned to the “cols” variable. Then using the columns in the “cols” variable, a new data framework was developed. This is going to be the “feature set.” Next, the column “Opportunity Result” from the *“sales_data”* data frame was used to develop a new data framework called “target.”

The second phase in this stage is to separate the date frameworks into training and testing subsets using the steps below. Depending on the data set and desired predictions, it needs to be split into training and testing subset accordingly. For this exercise, we will use 75% of the data as a training subset, and the rest 25% will be used for the testing subset. We will leverage the *“train_test_split()”* technique in “Scikit-Learn” in order to separate the information using steps and code as below:

Step 1 – To import the required module, use the command *“from sklearn.model_selection import train_test_split”*.

Step 2 – To separate the data set, use command *“split data set into train and test* *setsdata_train, data_test, target_train, target_test = train_test_split(data, target, test_size = 0.30, random_state = 10)”*.

With the code above, the *“train_test_split”* module was first imported, followed by the use of *“train_test_split()”* technique to generate “training subset *(data_train, target_train)”* and “testing subset (*data_test, data_train).”* The *“train_test_split()”* technique’s first argument pertains to the features that were divided in the preceding stage, the next argument relates to the (“Opportunity Result”) target. We are using 30% for this example, although it can be any amount. The fourth 'random state' argument is used to make sure that the results can be reproduced every time.

Building the Machine Learning Model

The “machine_learning_map” provided by Scikit-Learn is widely used to choose the most appropriate ML algorithm for the data set. For this exercise, we will be using “Linear Support Vector Classification” and “K-nearest neighbors’ classifier” algorithms.

Linear Support Vector Classification

“Linear Support Vector Classification" or "Linear SVC" is a sub-classification of

"Support Vector Machine (SVM)" algorithm, which we have reviewed in chapter 2 of this book titled "Machine Learning Algorithms." Using Linear SVC, the data can be divided into different planes so the algorithm can identify the optimal group structure for all the data classes.

Here are the steps and code for this algorithm to build our first ML model:

Step 1 – To import the required modules, use commands *“from sklearn.svm import LinearSVC”* and *“from sklearn.metrics import accuracy_score”*.
Step 2 – To develop an LinearSVC object type, use command *“svc_model = LinearSVC(random_state=0)”*.
Step 3 – In training the algorithm and and generating predictions from the testing data, use command *“pred = svc_model.fit(data_train, target_train).predict(data_test)”*.
Step 4 – To display the model accuracy score, use command *“print (‘LinearSVC accuracy:’, accuracy_score(target_test, pred, normalize = True))”*.

With the code above, the required modules were imported in the first step. We then developed a type of Linear SVC using *the "svc_model”* object with “random_state” as ‘0'. In step 3, the “Linear SVC” algorithm is trained on the training data set and subsequently used to generate predictions for the target from the testing data. The *“accuracy_score()”* technique was used in the end to verify the “accuracy score” of the model, which could be displayed as “LinearSVC accuracy: 0.777811004785”, for instance.

K-nearest Neighbors Classifier

The “k-nearest neighbors (k-NN)” algorithm is referred to as “a non-parametric method used for classification and regression in pattern recognition.” In cases of classification and regression, “the input consists of the nearest k closest training examples in the feature space.” K-NN is a form of "instance-based learning," or "lazy learning," in which the function is only locally estimated, and all calculations are delayed until classification. The output is driven by the fact, whether the classification or regression method is used for "k-NN":

- “k-nearest neighbors classification” - The “output” is a member of the class. An “object” is classified by its neighbors’ plurality vote, assigning the object to the most prevalent class among its nearest “k-neighbors,” where “k” denotes a small positive integer. If k= 1, the “object” is simply allocated to the closest neighbor's class.
- “k-nearest neighbors regression” - The output is the object's property value, which is computed as an average of the k-nearest neighbors' values.

A helpful method for both classification and regression can be assigning weights to the neighbors' contributions to allow closer neighbors to make more contributions in the average compared to the neighbors located far apart. For instance, a known “weighting scheme” is to assign each neighbor a weight of “*1/d”*, where “d” denotes the distance from the neighbor. The neighbors are selected from a set of objects for which the “class” (for “k-NN classification”) or the feature value of the “object” (for “k-NN regression”) is known.

Here are the steps and code for this algorithm to build our next ML model:

Step 1 – To import required modules, use the command *“from sklearn. neighbors import KNeighborsClassifier”* and *“from sklearn.metrics import accuracy_score”.*

Step 2 – In creating the object of the classifier, use command *“neigh = KNeighborsClassifier(n_neighbors=3)”.*

Step 3 – In training the algorithm, use command *“neigh.fit(data_train, target_train).”*

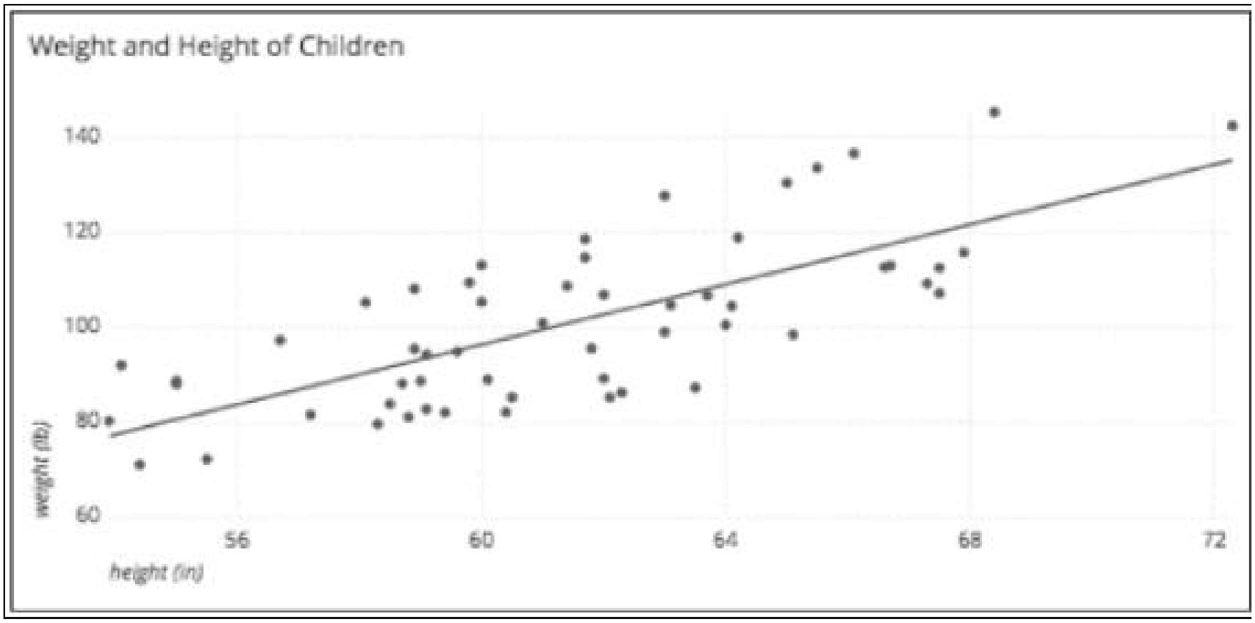
Step 4 – To generate predictions, use command *“pred = neigh.predict(data_test)”.*

Step 5 – To evaluate the accuracy, use command *“print (‘KNeighbors accuracy score:,’ accuracy_score(target_test, pred)).”*

Now that our preferred algorithms have been introduced, the model with the highest accuracy score can be easily selected. In Scikit-Learn, the “Yellowbrick library” can be used, which offers techniques to depict various scoring techniques visually.

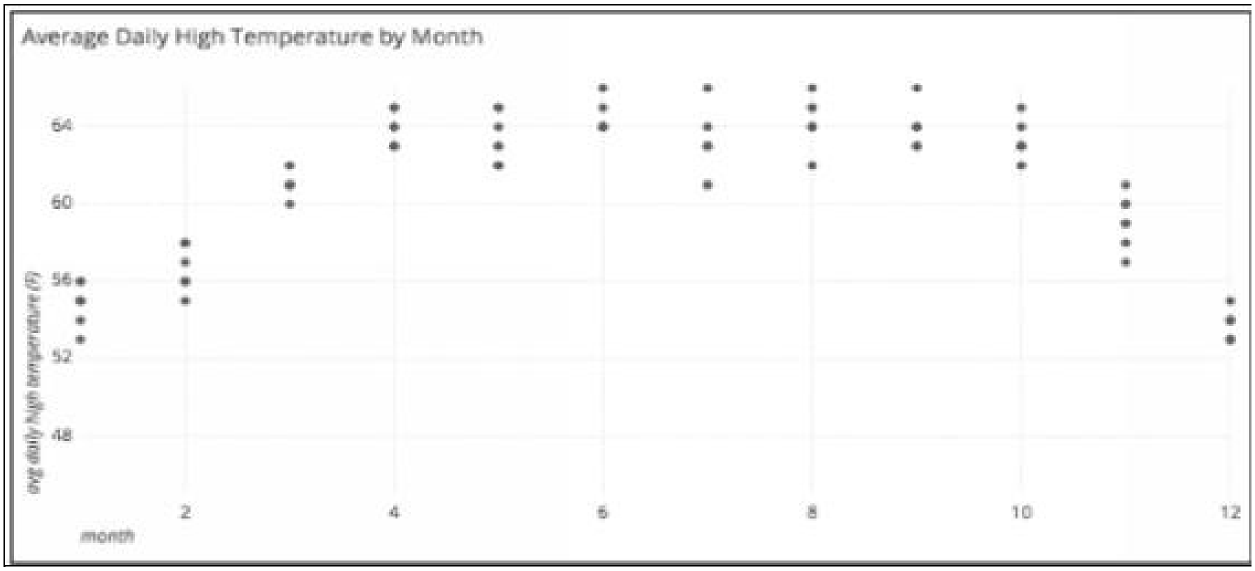
Data Visualization with Scatter Plots

A scatter plot can be defined as "a two-dimensional data visualization that uses dots to represent the values obtained for two different variables-one plotted along the x-axis, and the other plotted along the y-axis." It is also known as a “scatter graph” or "scatter chart." For instance, the "scatter plot" seen in the picture below depicts a fictional set of height and weight measures for children. Each "dot" in the plot is used to represent an individual with measures of their height along the "x-axis" and weight along the "y-axis."

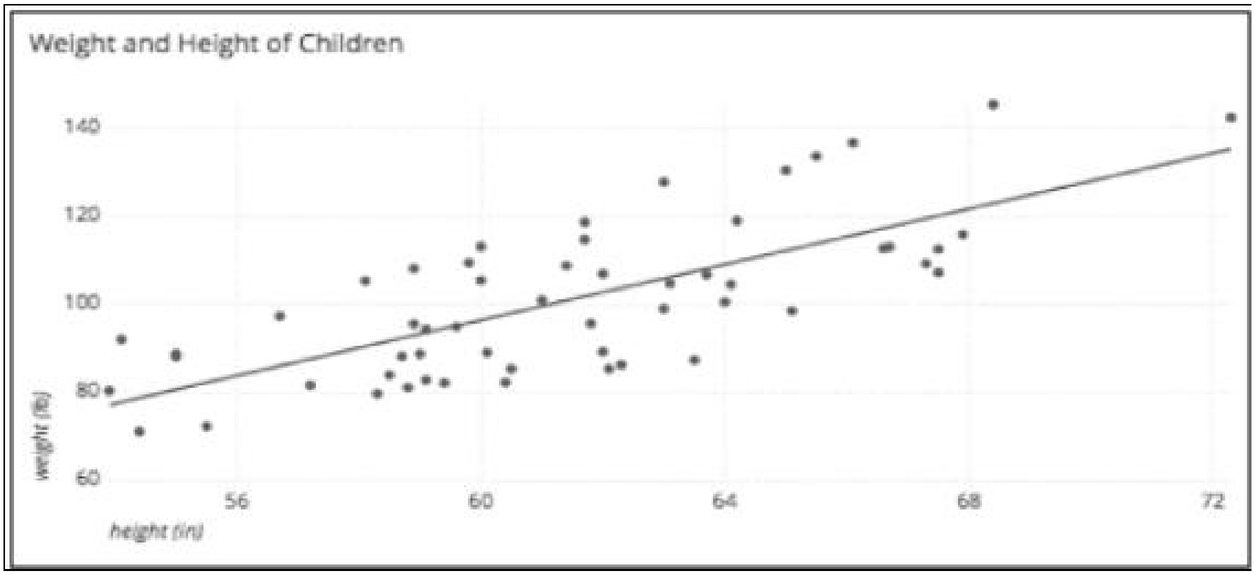


“Scatter plots" are highly useful when you are interested in representing the relationship that exists between two distinct variables. “Scatter plots" are often referred to as "plots of correlation," given the fact that they demonstrate how two distinct variables are related to each other. In the "scatter plot" above, the chart depicts much more than a simple log of the height and weight of children. It also offers a visual of the relationship between the two measures, denoting that as the height increases, the child's weight is also increasing. Now, you can easily conclude that this height and weight relationship isn't ideal, as some taller kids can weigh less than some shorter kids, but the overall trend is fairly satisfactory, and it can be observed that there is a direct relationship between the height and weight of children.

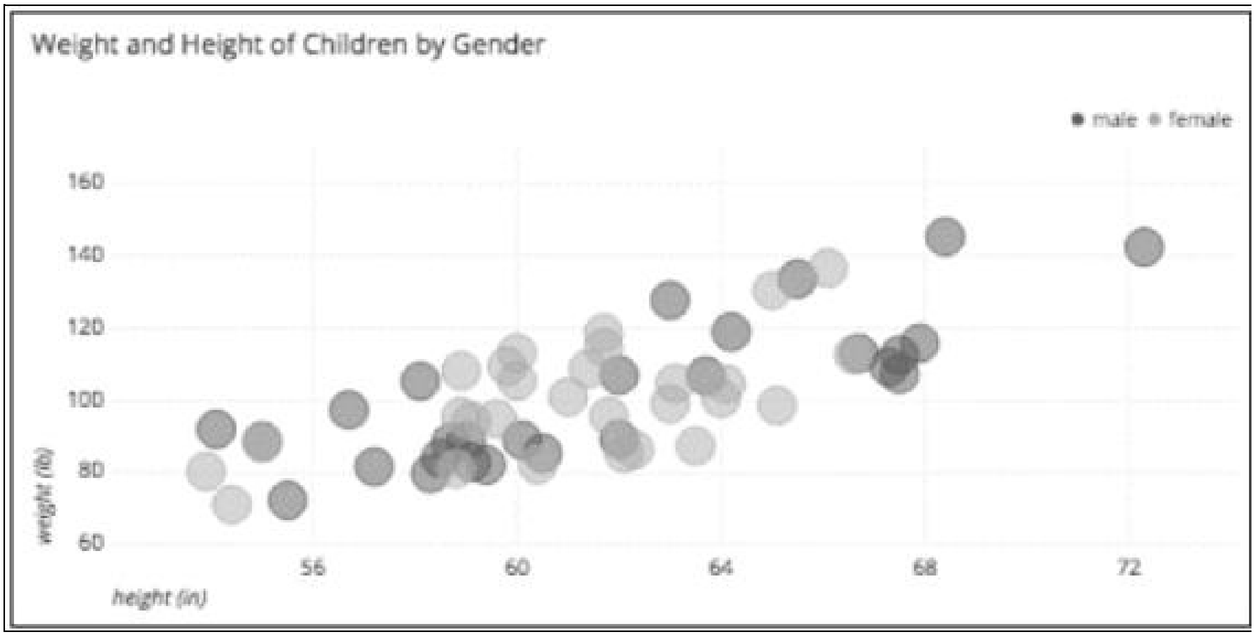
It is important to remember that not all relationships can be linear. For instance, the chart in the picture below indicates an "average of daily high temperature” measured over 7 years, demonstrating a familiar parabolic relationship between these variables as the daily high temperature tends to peak during summer.



Scatter plots" often contain a trend-line to clarify the relationship between the variable, as shown in the picture below.



Moreover, the shape, size, and color of the "dot" can be considered and utilized as additional data variables. For instance, the plot below represents the data on the height and weight of the children, but by adding the color of the "dot" to depict the gender of the child, we have acquired a third variable for our analysis.



```
<li><a href="index.html">Home</a></li>
<li><a href="home-events.html">Home Events</a></li>
<li class="has-children"> <a href="#" class="current">Header Options</a>
    <ul>
        <li><a href="tall-button-header.html">Tall Button Header</a></li>
        <li><a href="image-logo.html">Image Logo</a></li>
        <li class="active"><a href="tall-logo.html">Tall Logo Image</a>
    </ul>
</li>
<li class="has-children"> <a href="#">Carousels</a>
    <ul>
        <li><a href="variable-width-slider.html">Variable Image Wi
        <li><a href="testimonial-slider.html">Testimonial Slider</
        <li><a href="featured-work-slider.html">Featured Work Sli
        <li><a href="equal-column-slider.html">Equal Column Slid
        <li><a href="video-slider.html">Video Slider</a></li>
        <li><a href="mini-bootstrap-carousel.html">Mini Slider
    </ul>
</li>
```

Chapter 4: Applications of Big Data Analysis

Industrial Applications

The applications of Big data and Big Data Analytics are benefiting, both small and big companies across various industrial domains. Some of the widely used industrial applications are:

eCommerce

Over 2.6 billion and counting active social media users include customers and potential customers for every company out there. The race is on to create more effective marketing and social media strategies, powered by machine learning, aimed at providing enhanced customer experience to turn prospective customers into raving fans. The process of sifting through and analyzing a massive amount of data has not only become feasible, but it's easy now. The ability to bridge the gap between execution and big data analysis has been supplemented by artificial intelligence marketing solutions.

Artificial Intelligence (AI) marketing can be defined as a method of you using artificial intelligence consonants like machine learning on available customer data to anticipate customer's needs and expectations while significantly improving the customer's journey. Marketers are able to boost their campaign performance and return on investment read a little to no extra effort in the light of big data insights provided by artificial intelligence marketing solutions. The key elements that make AI marketing as powerful are:

- Big data - A marketing company's ability to aggregate and segment a huge dump of data with minimal manual work is referred to as Big Data. The marketer can then leverage the desired medium to ensure the appropriate message is being delivered to the target audience at the right time.
- Machine learning - Machine learning platforms enable marketers to identify trends or common occurrences and gather effective insights and responses, thereby deciphering the root cause and probability of recurring events.
- Intuitive platform – Super fast and easy to operate applications are integral to AI marketing. Artificial intelligence technology is capable of interpreting emotions and communicating like a human, allowing AI-based platforms to understand open form content like email responses and social media.

Predictive Analysis

All artificial intelligence technology-based solutions are capable of extracting information from data assets to predict future trends. AI technology has made it possible to model trends that could previously be determined only retroactively. These predictive analysis models can be reliably used in decision-making and to analyze customers' purchase behavior. The model can successfully determine when the consumer is more likely to purchase something new or reorder an old purchase. The marketing companies are now able to reverse engineer customer's experiences and actions to create more lucrative marketing strategies. For example, FedEx and Sprint are using predictive analytics to identify customers who are at potential risk of deflecting to the competitor.

Smart searches

Only a decade ago, if you type in "women's flip flops" on Nike.com, the probability of you finding what you were looking for would be next to zero. But today's search engines are not only accurate but also much faster. This upgrade has largely been brought on by innovations like "semantic search" and "natural language processing" that enable search engines to identify links between products and provide relevant search results, recommend similar items, and auto-correct typing errors. The artificial intelligence technology and big data solutions are able to rapidly analyze user search patterns and identify key areas that the marketing companies should focus on.

In 2015, Google introduced the first Artificial Intelligence-based search algorithm called “RankBrain.” Following Google’s lead, other major e-commerce websites, including Amazon has incorporated big data analysis and artificial intelligence into their search engines to offer smart search experience for their customers, who are able to find desired products even when they don’t know exactly what they’re looking for. Even small e-commerce stores have access to Smart search technologies like “Elasticsearch.” The data-as-a-service companies like “Indix" allow companies to learn from other larger data sources to train their product search models.

Recommendation Engines

Recommendation engines have quickly evolved into fan favorites and are loved by the customers just as much as the marketing companies. "Apple Music" already knows your taste in music better than your partner, and Amazon always presents you with a list of products that you might be interested in buying. This kind of discovery aide that is able to sift through millions of available options and hone in on an individual's needs are proving indispensable for large companies with huge physical and digital inventories.

In 1998, Swedish computational linguist, Jussi Karlgren, explored the practice of clustering customer behaviors to predict future behaviors in his report titled

“Digital bookshelves.” The same here, Amazon implemented collaborative filtering to generate recommendations for their customers. The gathering and analysis of consumer data paired with individual profile information and demographics, by the predictive analysis based systems allow the system to continually learn and adapt based on consumer activities such as likes and dislikes on the products in real-time. For example, the company "Sky" has implemented a predictive analysis based model that is capable of recommending content according to the viewer's mode. The smart customer is looking for such an enhanced experience not only from their Music and on-demand entertainment suppliers but also from all other e-commerce websites.

Product Categorization and Pricing

E-commerce businesses and marketing companies have increasingly adopted artificial intelligence in their process of categorization and tagging of the inventory. The Marketing companies are required to deal with awful data just as much, if not more than amazingly organized, clean data. This bag of positive and negative examples serves as training resources for predictive analysis based classification tools. For example, different detailers can have different descriptions for the same product, such as sneakers, basketball shoes, trainers, or Jordan's, but the AI algorithm can identify that these are all the same products and tag them accordingly. Or if the data set is missing the primary keyword like skirts or shirts, the artificial intelligence algorithm can identify and classify the item or product as skirts or shirts based solely on the surrounding context.

We are familiar with the seasonal rate changes of the hotel rooms, but with the advent of artificial intelligence, product prices can be optimized to meet the demand with a whole new level of precision. The machine learning algorithms are being used for dynamic pricing by analyzing customer's data patterns and making near accurate predictions of what they are willing to pay for that particular product as well as their receptiveness to special offers. This empowers businesses to target their consumers with high precision and calculated whether or not a discount is needed to confirm the sale. Dynamic pricing also allows businesses to compare their product pricing with the market leaders and competitors and adjust their prices accordingly to pull in the sale. For example, “Airbnb” has developed its dynamic pricing system, which provides ‘Price Tips' to the property owners to help them determine the best possible listing price for their property. The system takes into account a variety of influencing factors such as geographical location, local events, property pictures, property reviews, listing features, and most importantly, the booking timings and the market demand. The final decision of the property owner to follow or ignore the provided ‘price tips' and the success of the listing are also monitored by the system, which will then process the results and adjust its algorithm accordingly.

Predictive Analysis

All artificial intelligence technology-based solutions are capable of extracting information from data assets to predict future trends. AI technology has made it possible to model trends that could previously be determined only retroactively. These predictive analysis models can be reliably used in decision-making and to analyze customers' purchase behavior. The model can successfully determine when the consumer is more likely to purchase something new or reorder an old purchase. The marketing companies are now able to reverse engineer customer's experiences and actions to create more lucrative marketing strategies. For example, FedEx and Sprint are using predictive analytics to identify customers who are at potential risk of deflecting to the competitor.

Customer Targeting and Segmentation

For the marketing companies to be able to reach their customers with a high level of personalization, they are required to target increasingly granular segments. The artificial intelligence technology can draw on the existing customer data and train Machine learning algorithms against "gold standard" training sets to identify common properties and significant variables. The data segments could be as simple as location, gender, and age, or as complex as the buyer's persona and past behavior. With AI, Dynamics Segmentation is feasible which accounts for the fact that customers' behaviors are ever-changing, and people can take on different personas in different situations.

Sales and Marketing Forecast

One of the most straightforward artificial intelligence applications in marketing is in the development of sales and marketing forecasting models. The high volume of quantifiable data such as clicks, purchases, email responses, and time spent on webpages serve as training resources for the machine learning algorithms. Some of the leading business intelligence and production companies in the market are Sisense, Rapidminer, and Birst. Marketing companies are continuously upgrading their marketing efforts, and with the help of AI and machine learning, they can predict the success of their marketing initiatives or email campaigns. Artificial intelligence technology can analyze past sales data, economic trends as well as industrywide comparisons to predict short and long-term sales performance and forecast sales outcomes. The sales forecasts model aid in the estimation of product demand and to help companies manage their production to optimize sales.

Programmatic Advertisement Targeting

With the introduction of artificial intelligence technology, bidding on and targeting program based advertisement has become significantly more efficient. Programmatic advertising can be defined as “the automated process of buying and selling ad inventory to an exchange which connects advertisers to publishers.” To allow real-time bidding for inventory across social media channels and mobile devices as well as television, artificial intelligence technology is used. This also goes back to predictive analysis and the ability to model data that could previously only be determined retroactively. Artificial intelligence is able to assist the best time of the day to serve a particular ad, the probability of an ad turning into sales, the receptiveness of the user, and the likelihood of engagement with the ad.

Programmatic companies have the ability to gather and analyze visiting customers’ data and behaviors to optimize real-time campaigns and to target the audience more precisely. Programmatic media buying includes the use of "demand-side platforms" (to facilitate the process of buying ad inventory on the open market) and "data management platforms" (to provide the marketing company an ability to reach their target audience). In order to empower the marketing rep to make informed decisions regarding their prospective customers, the data management platforms are designed to collect and analyze the big volume of website "cookie data." For example, search engine marketing (SEM) advertising practiced by channels like Facebook, Twitter, and Google. To efficiently manage huge inventory of the website and application viewers, programmatic ads provide a significant edge over competitors. Google and Facebook serve as the gold standard for efficient and effective advertising and are geared to words providing a user-friendly platform that will allow nontechnical marketing companies to start, run and measure their initiatives and campaigns online.

Visual Search and Image Recognition

Leaps and bounds of the advancements in artificial intelligence-based image recognition and analysis technology have resulted in uncanny visual search functionalities. With the introduction of technology like Google Lens and platforms like Pinterest, people can now find results that are visually similar to one another using visual search functionality. The visual search works in the same way as traditional text-based searches that display results on a similar topic. Major retailers and marketing companies are increasingly using the visual search to offer an enhanced and more engaging customer experience. Visual search can be used to improve merchandising and provide product recommendations based on the style of the product instead of the consumer's past behavior or purchases.

Major investments have been made by Target and Asos in the visual search technology development for their e-commerce website. In 2017, Target announced a partnership with interest that allows integration of Pinterest's visual search application called "Pinterest lens" into Target's mobile application. As a result, shoppers can take a picture of products that they would like to purchase while they are out and about and find similar items on Target's e-commerce site. Similarly, the visual search application launched by Asos called “Asos’ Style Match” allows shoppers to snap a photo or upload an image on the Asos website or application and search their product catalog for similar items. These tools attract shoppers to retailers for items that they might come across in a magazine or while out and about by helping them to shop for the ideal product even if they do not know what the product is.

Image recognition has tremendously helped marketing companies to gain an edge on social media by allowing them to find a variety of uses of their brand logos and products in keeping up with the visual trends. This phenomenon is also called "visual social listening" and allows companies to identify and understand where and how customers are interacting with their brand, logo, and product even when the company is not referred directly by its name.

Healthcare Industry

With the increasing availability of healthcare data, big data analysis has brought on a paradigm shift to healthcare. The primary focus of big data analytics in the healthcare industry is the analysis of relationships between patient outcomes and the treatment or prevention technique used. Big data analysis driven Artificial Intelligence programs have successfully been developed for patient diagnostics, treatment protocol generation, drug development, as well as patient monitoring and care. The powerful AI techniques can sift through a massive amount of clinical data and help unlock clinically relevant information to assist in decision making.

Some medical specialties with increasing big data analysis based AI research and applications are:

- Radiology – The ability of AI to interpret imaging results supplements the clinician's ability to detect changes in an image that can easily be missed by the human eye. An AI algorithm recent created at Stanford University can detect specific sites in the lungs of the pneumonia patients.
- Electronic Health Records – The need for digital health records to optimize the information spread and access requires fast and accurate logging of all health-related data in the systems. A human is prone to errors and may be affected by cognitive overload and burnout. This process has been successfully automated by AI. The use of Predictive models on the electronic health records data allowed the prediction of individualized treatment response with 70-72% accuracy at baseline.
- Imaging – Ongoing AI research is helping doctors in evaluating the outcome of corrective jaw surgery as well as in assessing the cleft palate therapy to predict facial attractiveness.

Entertainment Industry

Big data analysis, in coordination with Artificial intelligence, is increasingly running in the background of entertainment sources from video games to movies and serving us a richer, engaging, and more realistic experience. Entertainment providers such as Netflix and Hulu are using big data analysis to provide users personalized recommendations derived from individual user’s historical activity and behavior. Computer graphics and digital media content producers have been leveraging big data analysis based tools to enhance the pace and efficiency of their production processes. Movie companies are increasingly using machine learning algorithms in the development of film trailers and advertisements as well as pre-and post-production processes. For example, big data analysis and an artificial intelligence-powered tool called "RivetAI" allows producers to automate and excellently read the processes of movie script breakdown, storyboard as well as budgeting, scheduling, and generation of shot-list. Certain time-consuming tasks carried out during the post-production of the movies such as synchronization and assembly of the movie clips can be easily automated using artificial intelligence.

Marketing and Advertising

A machine learning algorithm developed as a result of big data analysis can be easily trained with texts, stills, and video segments as data sources. It can then extract objects and concepts from these sources and recommend efficient marketing and advertising solutions. For example, a tool called "Luban” was developed by Alibaba that can create banners at lightning speed in comparison to a human designer. In 2016, for the Chinese online shopping extravaganza called “Singles Day,” Luban generated a hundred and 17 million banner designs at a speed of 8000 banner designs per second.

The “20th Century Fox” collaborated with IBM to use their AI system "Watson" for the creation of the trailer of their horror movie "Morgan." In order to learn the appropriate "moments" or clips that should appear in a standard horror movie trailer, Watson was trained to classify and analyze input "moments" from audio-visual and other composition elements from over a hundred horror movies. This training resulted in the creation of a six-minute movie trailer by Watson in a mere 24 hours, which would have taken human professional weeks to produce.

With the use of Machine learning, computer vision technology, natural language processing, and predictive analytics, the marketing process can be accelerated exponentially through an AI marketing platform. For example, the artificial intelligence-based marketing platform developed by Albert Intelligence Marketing is able to generate autonomous campaign management strategies, create custom solutions and perform audience targeting. The company reported a 183% improvement in customer transaction rate and over 600% higher conversation efficiency credited to the use of their AI-based platform.

In March 2016, the artificial intelligence-based creative director called “AI-CD ß” was launched by McCann Erickson Japan as the first robotic creative director ever developed. “AI-CD ß" was given training on select elements of various TV shows and the winners from the past 10 years of All Japan Radio and Television CM festival. With the use of data mining capabilities, "AI-CD ß” can extract ideas and themes fulfilling every client’s individual campaign needs.

Personalization of User Experience

The expectations of the on-demand entertainment users for rich and engaging personal user experience is ever-growing. One of the leading on-demand entertainment platforms, Netflix, rolled out artificial intelligence-based workflow management and scheduling application called "Meson," comprised of various "machine learning pipelines" that are capable of creating, training, and validating personalization algorithms to provide personalized recommendations to users. Netflix collaborated with the University of Southern California to develop a new Machine learning algorithms that can compress video for high-quality streaming without degrading image quality called "Dynamic Optimizer." This artificial intelligence technology will address streaming problems in developing nations and mobile device users by optimizing video fluency and definition.

IBM Watson recently collaborated with IRIS.TV offers a business-to-business service to media companies such as CBS, The Hollywood Reporter, and Hearst Digital Media by tracking and improving the introduction of their customers with their web content. IBM Watson is boosting IRIS.TV company's Machine learning algorithms that can ‘learn' from users search history and recommend similar content. Reportedly, a 50% increase in view or retention or a small PDF three months was achieved by the Hollywood reporter with the use of IRIS.TV application.

Search Optimization and Classification

The ability to transform text, audio, and video content into digital copies has led to an explosion of media availability on the Internet, making it difficult for people to find exactly what they're looking for. To optimize the accuracy of search results, advancements are being made in machine learning technology. For example, Google is using artificial intelligence to augment its platform for accurate image searching. People can now simply upload a sample picture to Google Image instead of typing in keywords for their search. The image recognition technology used by Google image will automatically identify and manage features of the uploaded user image and provide search results with similar pictures. Google is also using artificial intelligence technology in advertisement positioning across the platform. For example, a pet food ad will only appear on the pet-related website, but a chicken wings advertisement will not appear on a site targeted to vegetarians.

The company Vintage Cloud has partnered with an artificial intelligence-based startup called "ClarifAI” to develop a film digitalization platform. With the use of computer vision API provided by ClarifAI, Vintage Cloud succeeded in burgeoning the speed of movie content classification and categorization.

A visual assets management platform integrated with machine learning algorithms has been developed by a company called “Zorroa." This platform enables users to search for specific content within large databases called an "Analysis Pipeline." The database contains processors that can tag each visual asset uniquely and Machine learning algorithms that have been ‘trained' to identify specific components of the visual data. This visual content is then organized and cataloged to deliver high-quality search results.

Conclusion

Thank you for making it through to the end of *Python for Data Analysis: A Basic Guide for Beginners to Learn the Language of Python Programming Codes Applied to Data Analysis with Libraries Software Pandas, Numpy, and IPython.* Let’s hope it was informative and able to provide you with all of the tools you need to achieve your goals, whatever they may be.

The next step is to make the best use of your new-found wisdom on the cutting-edge technology of "Big Data Analytics" and its applications in the world of healthcare and eCommerce. The smart and savvy customers today can be easily swayed by modern companies with a whimsical edge offering consumers a unique, rich, and engaging experience. It is getting increasingly challenging for traditional businesses to retain their customers without adopting the big data analytics technology explained in this book.

You are now ready to make your own predictive analysis model by leveraging all the free and open-source data libraries described in this book. To make the best use of this book, I recommend that you download these free resources and perform hands-on exercises to solidify your understanding of the concepts explained. The skillset of data analysis is always in demand with a lot of high pay job opportunities. Here’s hoping this book has taken you a step closer to your dream job!

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