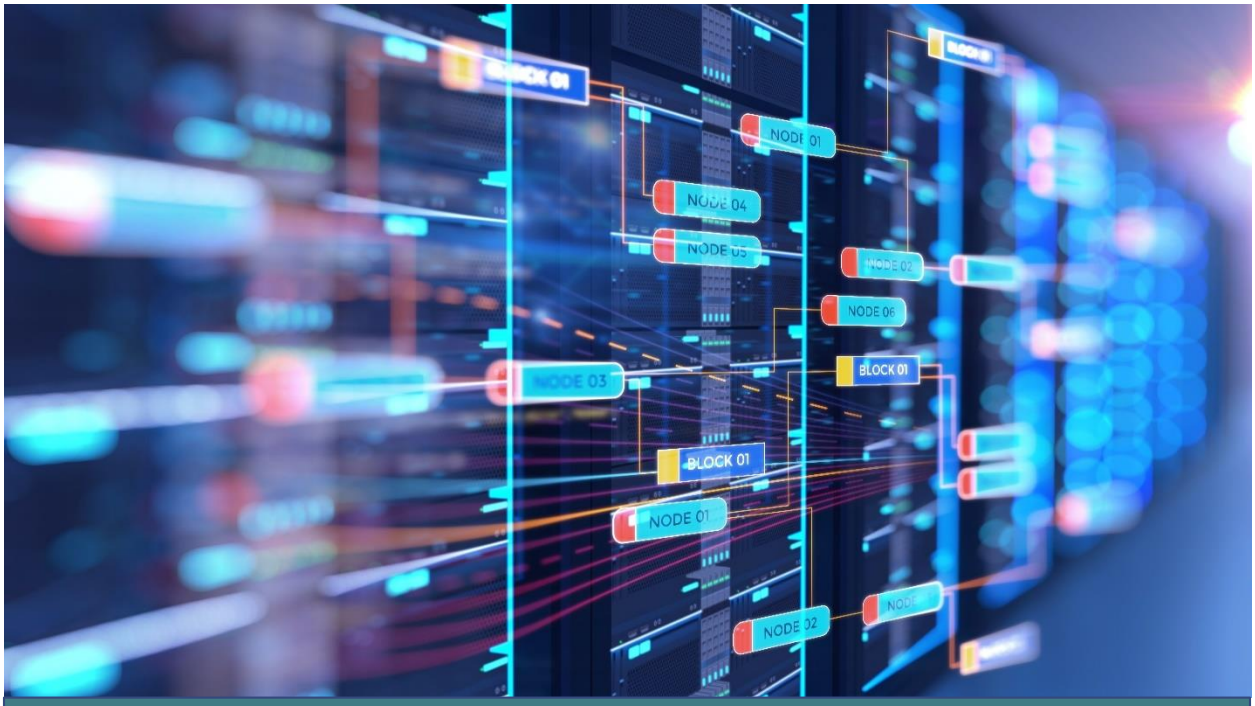


DATA SCIENCE

GRADE XII

Version 1.0

Microsoft

DATA SCIENCE

GRADE XII

Student Handbook

ACKNOWLEDGEMENTS

Patrons

- Sh. Ramesh Pokhriyal 'Nishank', Minister of Human Resource Development, Government of India
- Sh. Dhotre Sanjay Shamrao, Minister of State for Human Resource Development, Government of India
- Ms. Anita Karwal, IAS, Secretary, Department of School Education and Literacy, Ministry Human Resource Development, Government of India Advisory

Editorial and Creative Inputs

- Mr. Manuj Ahuja, IAS, Chairperson, Central Board of Secondary Education

Guidance and Support

- Dr. Biswajit Saha, Director (Skill Education & Training), Central Board of Secondary Education
- Dr. Joseph Emmanuel, Director (Academics), Central Board of Secondary Education
- Sh. Navtez Bal, Executive Director, Public Sector, Microsoft Corporation India Pvt. Ltd.
- Sh. Omjiwan Gupta, Director Education, Microsoft Corporation India Pvt. Ltd
- Dr. Vinnie Jauhari, Director Education Advocacy, Microsoft Corporation India Pvt. Ltd.
- Ms. Navdeep Kaur Kular, Education Program Manager, Allegis Services India

Value adder, Curator and Co-Ordinator

- Sh. Ravinder Pal Singh, Joint Secretary, Department of Skill Education, Central Board of Secondary Education

ABOUT THE HANDBOOK

In today's world, we have a surplus of data, and the demand for learning data science has never been greater. The students need to be provided a solid foundation on data science and technology for them to be industry ready.

The objective of this curriculum is to lay the foundation for Data Science, understanding how data is collected, analyzed and, how it can be used in solving problems and making decisions. It will also cover ethical issues with data including data governance and builds foundation for AI based applications of data science.

Therefore, CBSE is introducing 'Data Science' as a skill module of 12 hours duration in class VIII and as a skill subject in classes IX-XII.

CBSE acknowledges the initiative by Microsoft India in developing this data science handbook for class XII students. This handbook introduces Classification and Regression algorithms; Unsupervised learning with practical examples. The course covers the theoretical concepts of data science followed by practical examples to develop critical thinking capabilities among students.

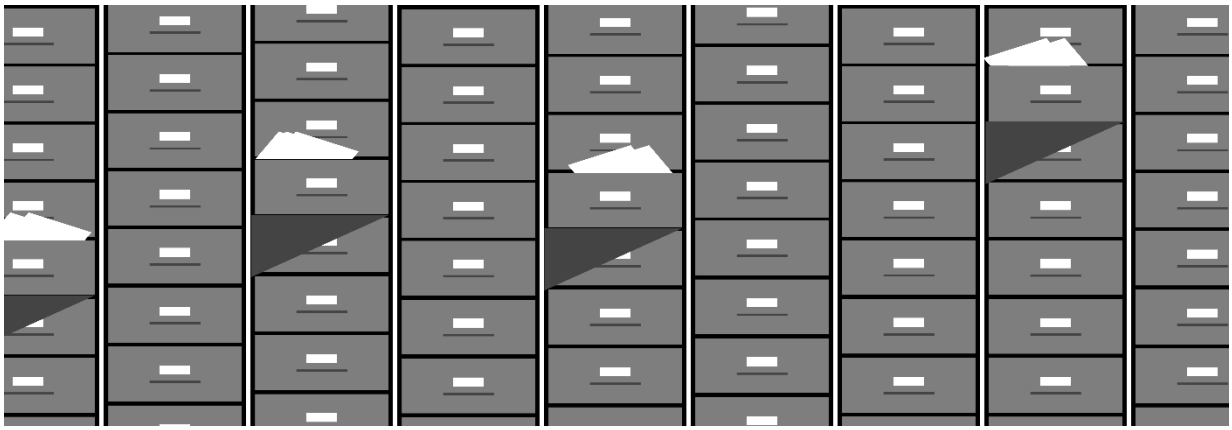
The purpose of the book is to enable the future workforce to acquire data science skills early in their educational phase and build a solid foundation to be industry ready.

Contents

CHAPTER

1

Data Governance



1. What is Data Governance?

Data governance can be thought of as a collection of people, technologies, processes, and policies that protect and help to manage the efficient use of data. Through data governance, we can ensure that the quality and security of the data used is maintained. Data governance also defines who can act, upon which data and using what methods.

Studying this chapter should enable you to understand:

- What is Data Governance?
- What are the ethical guidelines for governing data?
- What is Data Privacy?

Hence, data governance is a data management concept that ensures that high data quality exists throughout the complete lifecycle of the data along with effective controls on how and with whom data is shared.

Data governance focuses on areas such as include data integrity, data availability, usability and data consistency.

Data Governance covers the following aspects.

- Data Quality
- Data Security
- Data Architecture
- Data Integration and Interoperability
- Data Storage

2. Ethical Guidelines

Ethics can be said to be the moral principles that govern the behavior or actions of an individual or a group. These principles help us decide what is good or bad.

Software products and data are not always used for purposes that are good for society. This is why we need to adhere to a set of guidelines that can guide us on what is right and what is wrong.

To begin with, we must make sure that qualities such as integrity, honesty, objectivity, nondiscrimination are always part of the high-level principles which should be incorporated in all our processes.

Besides that, while dealing with data, we should also seek to include the following points.

- Keep the data secure.
- Create machine learning models that are impartial and robust
- Be as open and accountable as possible
- Use technologies and data architecture that has the minimum intrusion necessary.

Data privacy covers aspects such as

- How personal data is collected and stored by organizations.
- Whether and how personal data is shared with third parties.
- Government policies and regulatory restrictions regarding the storage and sharing of personal information.

3. Data Privacy

Data privacy is the right of any individual to have control over how his or her personal information is collected and used.

Data privacy is not just about secure data storage. There could be cases where personal identifiable information is collected and stored securely in an encrypted format, however, there is no agreement from the users regarding the collection of the data itself. In such cases, there is a clear violation of data privacy rules.

One major aspect of data privacy is that the individual is considered to be the sole owner of his data. In other words, he can request any organization to remove all the data they have collected

about him at any point in time. Data privacy rules are still evolving with time as more and more awareness about data privacy continues to spread.

Some of the important legislations for data privacy are discussed below.

GDPR - General Data Protection Regulation

The European Union made General Data Protection Regulation effective on May 25th, 2018 to protect European Union consumer data. All of the reformed laws are made to help consumers gain a high level of control over their data. It also offers more transparency to the end-user about the data collection and use process.

GDPR is based on the following important aspects

- Obtaining Consent
- Timely breach notification
- Right to access data
- Right to be forgotten
- Privacy by design

HIPAA - Health Insurance Portability and Accountability Act

The Health Insurance Portability and Accountability Act (HIPAA) was passed in the United States to protect healthcare information from fraud and theft. It also helps to manage Personally Identifiable Information stored by healthcare and insurance companies.

HIPAA returns control of data to the individuals by giving them the option to see their data at any time, ask for corrections and report any violations of privacy that they might suspect.

Some of the personal identifiers that HIPAA protects are as follows

- Names of parts of names
- Phone numbers, email addresses
- Geographical identifiers
- Fingerprints and retinal prints
- Social security numbers
- Medical records.

CCPA – California Consumer Privacy Act

California passed the CCPA on June 28, 2018, and it went into effect on January 1, 2020. The CCPA is landmark legislation designed to protect consumer data.

The CCPA provides residents living in the state of California with the right to request businesses:

- To disclose to them what personal information the businesses have about them and what they intend to do with it
- To request businesses to delete their personal information
- To request businesses not to sell their personal information

COPPA - Children's Online Privacy Protection Act

The Children's Online Privacy and Protection Act, which is commonly known as COPPA, is a law that deals with how websites and other online companies collect data from children who are less than the age of 13.

It was passed in the US in 1998 and came into effect on April 21, 2000. It details what must be included in a privacy policy for children. It also addresses when and how to seek

consent from parents or guardians for certain services and what responsibilities a company has to protect children's privacy and safety online.

PDP – Personal Data Protection Bill

The Personal Data Protection Bill 2019 was tabled in the Indian Parliament by the Ministry of Electronics and Information Technology on 11 December 2019. As of March 2020, the Bill is being analyzed by a Joint Parliamentary Committee (JPC) in consultation with experts and stakeholders. The Bill covers mechanisms for the protection of personal data and proposes the setting up of a Data Protection Authority of India for it.

Recap

- Data governance defines who can take action, upon which data and using what methods.
- Ethics guidelines for data stems from qualities like integrity, honesty, objectivity and nondiscrimination.
- Data privacy is the right of any individual to have control over how his or her personal information is collected and used.

Exercises

Objective Type Questions

Please choose the correct option in the questions below.

1. Which of the following statements is true?
   a. Data governance helps in effective data management.
   b. Ethical guidelines must be well-defined in any organization dealing with lots of data
   c. Data privacy is only about secure data storage
   d. Transparency is an important ethical guideline.

2. What are some important ethical guidelines for data?
   a. Keeping data secure.
   b. Making models impartial.
   c. Being open and accountable.
   d. All of the above.
3. Some organizations store more personal information than required. This is in accordance with Data Ethics.
   a. True
   b. False
4. Which data legislation was introduced in Europe?
   a. GDPR
   b. HIPAA
   c. COPPA
5. Which are some of the areas that data governance focuses on?
   a. data integrity,
   b. data availability
   c. data consistency
   d. All of the above

Standard Questions

Please answer the questions below in no less than 100 words.

1. What are some of the aspects covered by data governance?
2. Write a short note on the California Consumer Privacy Act.
3. Write a short note on the General Data Protection Regulation.

Higher Order Thinking Skills(HOTS)

Please answer the questions below in no less than 200 words.

1. In 2019, Canva, which is a famous website used for design, suffered a data breach that impacted more than 100 million users. The breach caused data such as email addresses and passwords to be leaked. Considering this situation, discuss how the website can prevent further leaks based on ethical guidelines.

2. Write a short note on how children are at higher risk of being manipulated on the internet.

Applied Project

Discuss how data governance and data best practices are followed at your school.

CHAPTER

2

Exploratory Data Analysis

Studying this chapter should enable you to understand:

- What is Exploratory Data Analysis?
- What is Univariate Analysis?
- What is Multivariate Analysis?
- What are the techniques to clean data?

1. Introduction

Exploratory Data Analysis is the process of carrying out an initial analysis of the available data to find out more about the data. We usually try to find patterns, try to spot anomalies, and test any hypotheses or assumptions that we may have about the data. The process of Exploratory Data Analysis is done with the help of summary statistics and graphical representations.

Thus, exploratory data analysis can be said to be an approach for analyzing data sets to summarize their key characteristics, often using visual methods. Often, in real life, exploratory data analysis (EDA) techniques that are used are graphical and only a few statistical techniques are used. The main reason for this is that EDA is way to explore data quickly and find patterns and this can be done best by using graphs.

There are a number of tools and methods to perform exploratory data analysis. Some of them have been discussed below.

- Univariate analysis of each feature variable in the raw dataset by preparing visualizations along with summary statistics.
- Bivariate analysis of feature variables by preparing visualizations

and summary statistics that allow us to determine the relationship between two variables in the dataset.

- Multivariate analysis of multiple feature variables for mapping and understanding interactions between different fields in the data.

- Graphical analysis by plotting the raw data (histograms, probability plots and lag plots) and plotting simple statistics (mean plots, standard deviation plots, and box plots)

- Using unsupervised learning techniques like clustering to identify the number of clusters in the data set. Clustering finds application in image compression and pattern recognition.

2. Univariate Analysis

Univariate analysis can be considered as the easiest form of data analysis where we only analyze only one variable from the entire dataset. Since we deal with only one variable, we do not have to worry about causes or relationships. The main purpose of the univariate analysis is to describe the data and find patterns that exist within it.

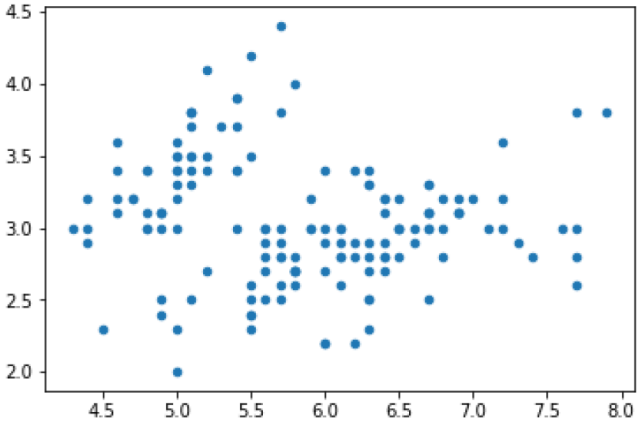
For univariate analysis, we pick up a variable from the dataset and try to analyze it in depth. One example of a variable in the univariate analysis might be "revenue". Another might be "height". For univariate analysis, we would not look at these two variables at the same time, nor would we look at the relationship between them.

Univariate analysis techniques involve both statistical and graphical methods. Some statistical methods for univariate data include looking at mean, mode, median, range, variance, maximum, minimum, quartiles, and standard deviation.

Some graphical methods involve preparing frequency distribution tables, bar charts, histograms, frequency polygons, and pie charts.

Now let's look at some of the graphs used for univariate analysis.

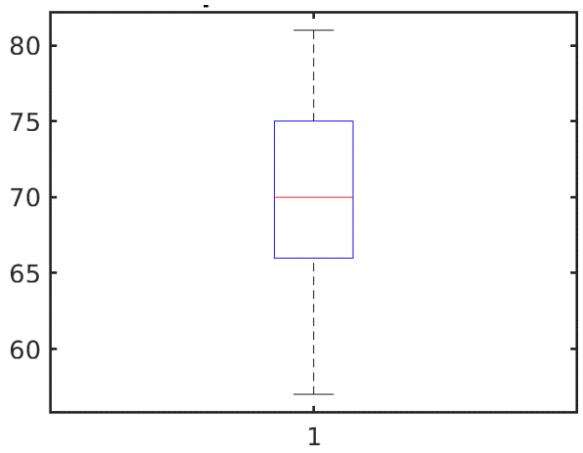
The diagram below shows a scatter plot for a single variable.



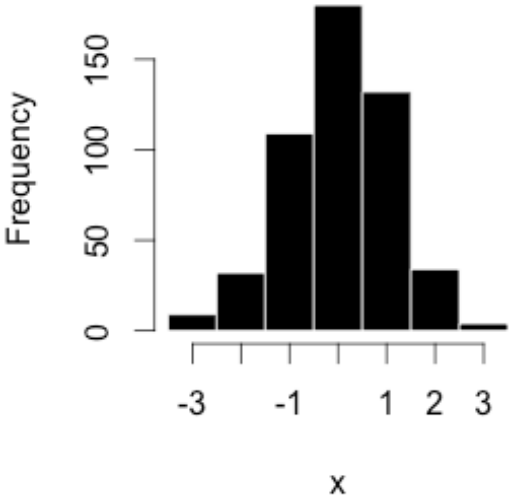
Activity 2.1

Take a look at the famous 'iris' dataset and create a scatter plot of the petal legths on a graph paper.

The diagram below shows the box plot for a variable. The box plot helps us to see the quantile ranges of the variable and whether any outliers are present in the data.



The diagram below shows a histogram for a variable showing the frequency distribution versus the range.

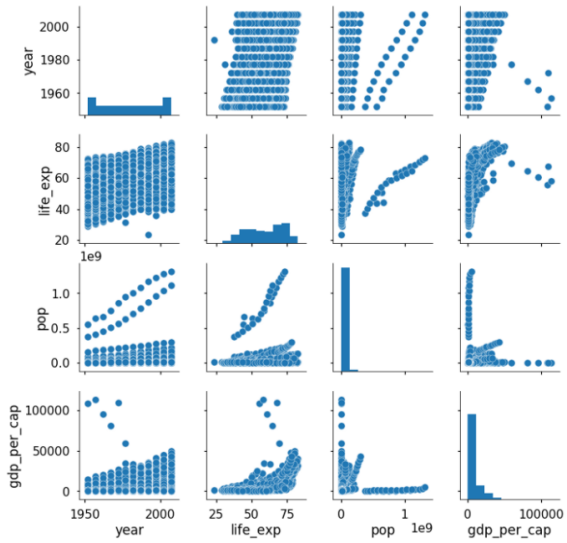


3. Multivariate Analysis

One of the ways to do multivariate analysis is Bivariate analysis. It refers to the analysis of two or more variables in the dataset. It is usually carried out between the target variable and another feature of the dataset. The main objective is to find out if there is a relationship between two different variables.

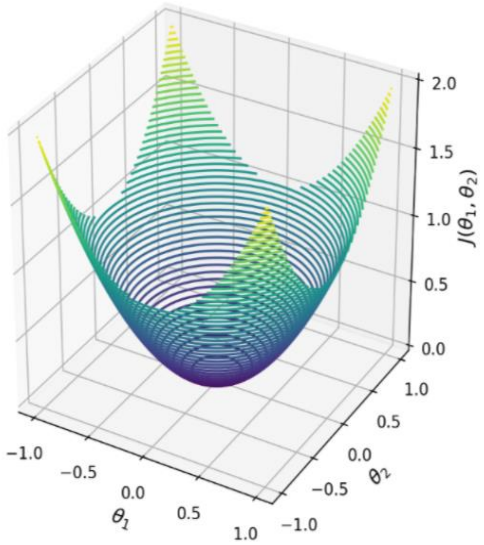
Bivariate analysis is usually done by using graphical methods like scatter plots, line charts, and pair plots. These simple charts can give us a picture of the relationship between the variables. Bivariate analysis is also a good way to measure the correlations between the two variables. For example – in a market survey we may be looking to analyze the relationship between price and sales of a product to see if there is any relationship.

Let us take a look at a pair plot used for bivariate analysis.



Multivariate analysis is a more complex form of statistical analysis technique and is used to analyze more than two variables in the data set. There are several ways to do a multivariate analysis, but it depends on your goals. Some of these methods include

Canonical Correlation Analysis, Cluster Analysis, contour plots, and Principal Component Analysis.



Activity 2.2

Take a look at the 'iris' dataset and try to plot the values of the petal length vs the sepal length. Do you find a positive relationship between the two?

4. Data Cleaning

Data cleaning is a very essential and often overlooked step in the pre-processing of data. It refers to the process of identifying incorrect, incomplete, and inaccurate data. We then clean the dataset by either removing the incorrect data or replacing it with better data.

Data cleaning is a fundamental aspect of data science. If you have a dataset that has been cleaned well, even simple algorithms can learn and give impressive insights from the data. Different types of data might need different approaches to cleaning. However, there is a systematic approach that works well on all kinds of data.

Some of the steps of data cleaning are as mentioned below.

1. **Remove duplicate observations** - Duplicate observations most frequently arise during data collection especially when we combine datasets from multiple places or scrape data from online sources. It is important to remove duplicates otherwise they can adversely affect the models we build.

2. **Remove irrelevant observations** – Quite often we are presented with datasets that have lots of extra, irrelevant data that does not add any value to the problem we are trying to solve. In such cases, we should remove the columns or rows of data that are irrelevant so that the model has to only learn from good relevant data.

3. **Remove unwanted outliers** - Outliers can cause problems with certain types of models. For example, linear regression models are less robust to outliers than decision tree models. In general, you should always look out for outliers in your data and remove them if they are unwanted so that it helps your model's performance. However, some outliers might be valid data so those values should be preserved.

4. **Fix data type issues** – Data types are often overlooked aspects of data. Many times, numerical or DateTime data might be saved as text. If this is not corrected in the data cleaning step, it

will cause problems when we use it to build a model. Therefore, you should always correct data type issues.

5. **Handle missing data** – Missing data is also a common issue with datasets. Many machine learning algorithms do not work well with missing data and so this must be handled well during the cleaning stage. The two common techniques to handle missing data is to either remove that row of data or to insert a value that is quite close to the mean or mode of the variable that is missing. For example, the height of students is univariate data.

Recap

- Exploratory Data Analysis is the process of carrying out an initial analysis of the available data.
- EDA involves graphical methods like frequency distribution tables, bar charts, histograms, frequency polygons, and pie charts.
- Univariate analysis is the simplest form of data analysis where we only analyze only one variable .
- Bivariate analysis refers to the analysis of two or more variables in the dataset.
- Bivariate analysis refers to the analysis of three or more variables.
- Data cleaning refers to the process of removing the incorrect data or replacing it with better data.

Exercises

Objective Type Questions

Please choose the correct option in the questions below.

1. You need to check the relationship between the two variables. Which graph would you use?
    a. Histogram
    b. Pair plot
    c. Box plot
    d. None of the above
2. You need to check if a variable has outliers. Which graph would you use?

a. Histogram
b. Pair plot
c. Box plot
d. None of the above

3. You need to perform a multivariate analysis. Which graph will you use?
    a. Contour plot
    b. Scatter plot
    c. Box plot
    d. None of the above
4. You need to perform a univariate analysis. Which graph will you use?
    a. Scatter plot
    b. Histogram
    c. Contour plot
    d. Both a and b
5. What is a data cleaning step?
    a. Removing duplicates
    b. Removing outliers
    c. All of the above

Standard Questions

Please answer the questions below in no less than 100 words.

1. What are some of the differences between univariate and multivariate analysis? Give some examples.
2. What are the ways to handle missing data?
3. What are some of the methods for univariate analysis?
4. What are the steps for cleaning raw data?

Higher Order Thinking Skills(HOTS)

Please answer the questions below in no less than 200 words.

1. What problems can outliers cause?
2. Why should irrelevant observations be removed from the data?
3. How can we use unsupervised learning for EDA?

Applied Project

Using the iris dataset provided in R Studio, perform a univariate analysis by creating scatter plots of sepal length, sepal width, petal length, and petal width.

CHAPTER

3 Classification Algorithms I

Studying this chapter should enable you to understand:

- What is a Decision Tree?
- How are Decision Trees used in Data Science?
- How to create a Decision Tree?

1. Introduction

In the last chapter, we learned about how to conduct an exploratory data analysis using various graphical techniques and how to clean the data that has been collected. In this chapter we will see how we can classify the data using an important algorithm called Decision Trees.

2. Introduction to Decision Trees

A Decision tree is a diagrammatic representation of the decision-making process and has a tree-like structure. Each internal node in the decision tree denotes a question on choosing a particular class. Every branch represents the outcome of the test, and each leaf node holds a class label.

You often use Decision Trees in your daily life without noticing them.

For example, when you go to a supermarket to buy milk for your family, the question which comes to your mind is – How much bread should I buy today?

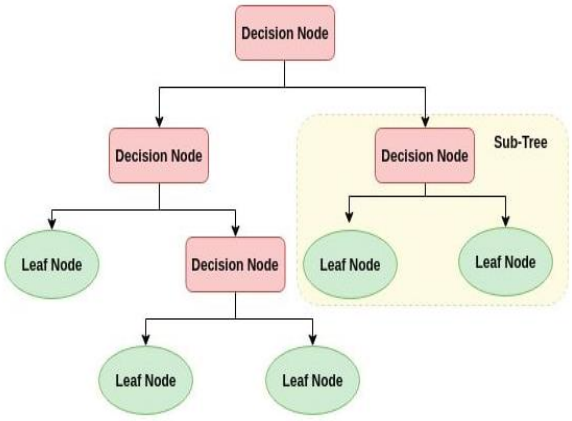
To answer the question, you subconsciously make calculations and you purchase the required quantity of bread.

Is it a weekday? On weekdays we require 1 packet of bread.

Is it a weekend? On weekends we require 2 packets of bread

Are we expecting any guests today? We need to buy extra bread for each guest.

The diagram below shows a sample decision tree. Each leaf node contains a class label and we split the entire population based on the test or criteria at each decision node.



Thus, in the end, we can classify the population based on the criteria we choose.

Decision Trees are considered to be one of the most efficient classification techniques. An even better way of using decision trees is to use the Random Forest algorithm which makes predictions based on the outcomes of several decision trees.

3. Applications of

Activity 3.1

Think about an everyday activity where you need to make a decision by thinking about several possibilities. Can a Decision Tree make help you make an effective decision?

Decision Trees

Decision Trees and other tree-based learning algorithms are considered to be one of the best and most used supervised learning methods. They are important as they are easy to visualize, understand and have a high ease of interpretation.

Sometimes the trend in the data is not linear, so we cannot apply linear classification techniques for these problems. The linear approaches we've seen so far will not produce accurate results.

For such cases, we need to build our models differently. Decision trees are a good tool for classifying observations when the trend is non-linear.

Decision Trees are versatile as they can be used to any kind of problem at hand - classification or regression. Also, unlike linear models that we have studied earlier, decision trees map both linear and non-linear relationships quite well.

Decision tree outputs are very easy to understand even for people from a non-analytical background. They do not

require any statistical knowledge to read and interpret them. Their graphical representation is very intuitive, and users can easily relate to their hypothesis.

Another major advantage of decision tree is that they can handle both numerical and categorical variables. Therefore, they require fewer data cleaning steps compared to some other modeling techniques. They are also not influenced much by outliers and missing values to a fair degree.

Decision trees are used to solve both classification and regression problems. However, there are certain differences between them. Let us take a brief look at the differences between them.

1. Regression trees are used when the dependent variable is continuous. Classification trees are used when the dependent variable is categorical.

2. In case of a regression tree, the value of the terminal nodes after training is the mean of the observations. Thus, predictions on unseen data are made using the mean.

3. In case of a classification tree, the value or class of the terminal nodes after training is the mode of the observations. Thus, predictions on unseen data are made using the mode.

The same data when plotted in a column chart will look like the below.

4. Creating a Decision

Activity 3.2

Try to find a real-world scenario where decision trees used for classification.

Tree

Do you know a real-world decision tree that helped save lives?

In late 1970, Lee Goldman, a U.S. Navy cardiologist, developed a decision tree to determine if a person was likely to have a heart attack. Lee spent years developing and testing a single model that would allow submarine doctors to quickly evaluate possible heart attack symptoms and determine if the submarine had to resurface and evacuate the chest pain sufferer.

This visual and simplified approach to decision making was one of the first uses of decision trees in real-world scenarios.

To create a decision tree, you can follow the steps below.

1. Think about your main objective for which you are creating the decision tree. The main decision that you are trying to make should be placed at the very top of

the decision tree. Therefore, the main objective should be the "root" of the entire diagram.

2. Next, you need to draw the branches and leaf nodes. For every possible decision, stemming from the root make a branch. One root or node can have two or more branches. At the end of the branches, attach the leaf nodes. The leaf nodes should represent the results of each decision. If another decision has to be made, draw a square leaf node. If the outcome is not quite certain, you should draw a circular node.

3. Finally, you need to calculate the probability of success of each decision being made. While creating the decision tree, it is essential to do some research, so that you can predict the probability of each decision. To do this research, you may examine old data or assess previous projects. Once you calculate the expected value of each decision in a tree, put the values on the branches.

The decision tree can be made in a linear form of decision rules where the outcome is the contents of the leaf node.

Activity 3.3

Create your own decision tree for a daily activity based on the steps above.

Recap

- A Decision tree is a flowchart like tree structure, where each internal node denotes a test on an attribute.
- An even better way of using decision trees is to use the Random Forest algorithm which makes predictions based on the outcomes of several decision trees.
- Decision tree outputs are very easy to understand even for people from a non-analytical background.
- A major advantage of decision trees is that they can handle both numerical and categorical variables.
- A visual and simplified approach to decision making was the one of the first uses of decision trees in real world scenarios.

Exercises

Objective Type Questions

Please choose the correct option in the questions below.

1. Which of the following are parts of a Decision Tree?
    a. Decision Node
    b. Leaf Node
    c. Branch
    d. All of the above
2. Which of the following statement is false?
    a. Decision Trees can contain only one branch.
    b. Decision Trees can be used for classification and regression.
    c. Random Forests algorithm uses Decision Trees.
    d. None of the above
3. Which of the following is a use case for Decision Trees?
    a. Classification
    b. Regression
    c. Both of the above
4. A decision tree can be further divided into further sub-trees.
    a. True
    b. False
5. Decision Trees are easier to understand and interpret.
    a. True
    b. False

Standard Questions

Please answer the questions below in no less than 100 words.

1. Write a short note on the application of classification algorithms.
2. In your own words, write down the steps to create a decision tree.
3. Write two advantages of using a decision tree.

Higher Order Thinking Skills(HOTS)

Please answer the questions below in no less than 200 words.

1. Write two disadvantages of using a decision tree.
2. Write a short note on the Random Forest algorithm.

Microsoft

Applied Project

In this exercise, we will use R Studio to generate a Decision Tree model to classify the famous iris dataset.

The iris dataset has 150 rows. Each row has the data of the iris plant under five attributes - sepal length, sepal width, petal length, petal width and species. There are three different kinds of iris in the dataset and each type has 50 rows of data. The three types are – setosa, versicolor and virginica. The iris dataset is already present in R Studio and does not need to be loaded.

The objective of the exercise is to build a Decision Tree model which can correctly classify a new iris flower into one of the three groups - setosa, versicolor, and virginica.

Launch R Studio and follow the steps below.

1. First, let us take a look at the data to see the attributes, rows, and columns. To do so, paste the code below in R Studio and click on Run.

```
library(rpart)
library(rpart.plot)
v <- iris$Species
table(v)
summary(iris)
head(iris)
```

You should get the output as shown below.

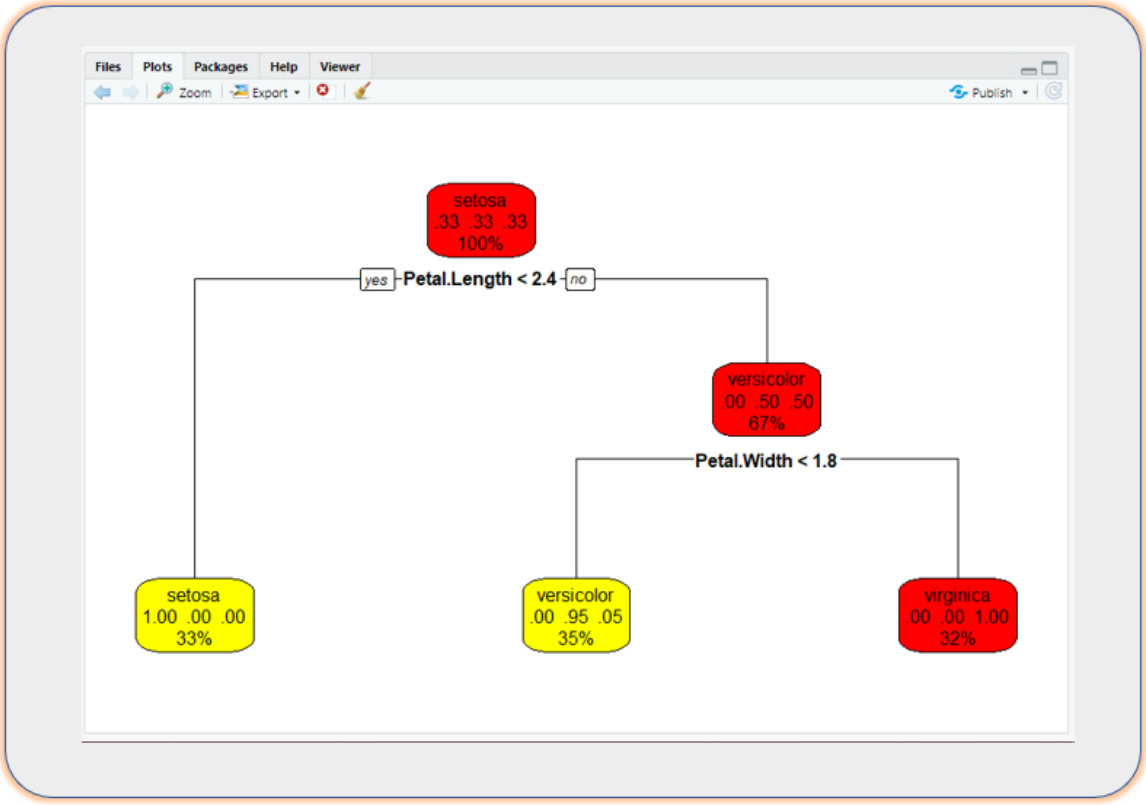
```
> table(v)
v
    setosa versicolor  virginica
        50         50         50
>
> summary(iris) #view statistical summary of dataset
  Sepal.Length    Sepal.Width     Petal.Length    Petal.Width          Species       train
 Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100   setosa    :50   Min.   :0.00
 1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.300   versicolor:50   1st Qu.:0.00
 Median :5.800   Median :3.000   Median :4.350   Median :1.300   virginica :50   Median :1.00
 Mean   :5.843   Mean   :3.057   Mean   :3.758   Mean   :1.199                   Mean   :0.74
 3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd Qu.:1.800                   3rd Qu.:1.00
 Max.   :7.900   Max.   :4.400   Max.   :6.900   Max.   :2.500                   Max.   :1.00
> head(iris)
  Sepal.Length Sepal.Width Petal.Length Petal.Width Species train
1          5.1         3.5          1.4         0.2  setosa     1
2          4.9         3.0          1.4         0.2  setosa     1
3          4.7         3.2          1.3         0.2  setosa     0
4          4.6         3.1          1.5         0.2  setosa     1
5          5.0         3.6          1.4         0.2  setosa     1
6          5.4         3.9          1.7         0.4  setosa     1
```

The output shows that we have 50 rows of data for the three varieties of the iris flower – setosa, versicolor, and virginica. It also shows that we have four attributes for the flowers – sepal length, sepal width, petal length, and petal width.

2. Now let us create a decision tree and plot it using rpart.plot. Paste and run the following code in RStudio.

```
set.seed(522)
iris[, 'train'] <- ifelse(runif(nrow(iris)) < 0.75, 1, 0)

trainSet <- iris[iris$train == 1,]
testSet <- iris[iris$train == 0, ]

trainColNum <- grep('train', names(trainSet))

trainSet <- trainSet[, -trainColNum]
testSet <- testSet[, -trainColNum]

treeFit <- rpart(Species~.,data=trainSet,method = 'class')

rpart.plot(treeFit, box.col=c("red", "yellow"))
```

You should get an output on the plot window as shown below.


Files
Plots
Packages
Help
Viewer
Zoom
Export
Publish
setosa
.33 .33 .33
100%
yes
Petal.Length < 2.4
no
versicolor
.00 .50 .50
67%
Petal.Width < 1.8
setosa
1.00 .00 .00
33%
versicolor
.00 .95 .05
35%
virginica
.00 .00 1.00
32%

CHAPTER

4

Classification Algorithms II

Studying this chapter should enable you to understand:

- What is the K- Nearest Neighbors algorithm?
- What are the pros and cons of k-Nearest Neighbors?
- What is cross validation and why is it useful?

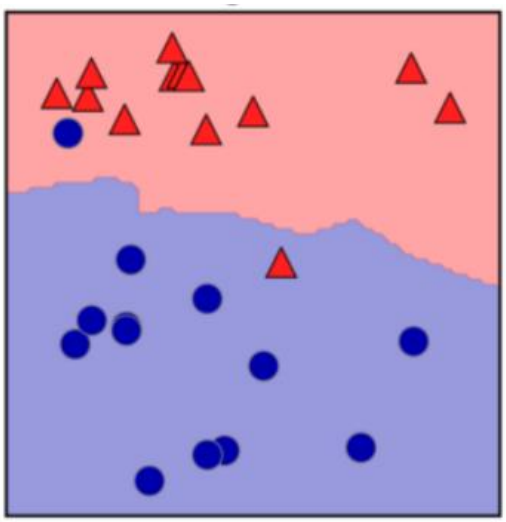
1. Introduction

In the last chapter, we learnt about how to use decision trees for classification problems. In this chapter, we will see how we can classify the data using another important algorithm called K-Nearest Neighbors. We will also take a look at cross-validation and how that helps us.

2. Introduction to K-Nearest Neighbors

The k-nearest neighbors (K-NN) algorithm is one of the most basic and easy-to-implement supervised machine learning algorithms. It can be used to solve both classification and regression problems. However, it is mostly used for classification problems especially in pattern recognition and data mining.

The K-NN algorithm works on the principle that similar things exist close to each other. In other words, data for a particular category of objects will usually be in the same feature space. We can get a more intuitive understanding of this concept by looking at the diagram below.



We can see from the diagram that there are two classes – circles and triangles. Most of the circular points lie at the bottom of the graph while the triangular points lie at the top. The boundary between the two classes is called the decision surface.

Thus, if we input a new unknown data point to the K-NN algorithm, it will classify it as a triangle if it lies in the upper part of the graph or as a circle if it lies on the lower part of the graph.

Internally, the K-NN algorithm finds the distance between the query point and all the points in the data. It then selects 'K' closest values to the query point and checks the class labels of these 'K' nearest points.

It then performs a majority vote to decide the class which is most frequent and outputs that as the predicted class label.

Now let us understand what 'K' in the K-NN algorithm is. K is the number of points that the K-NN algorithm checks before doing a majority vote.

In simple terms, if the value of K is 1, KNN will only check the value of the closest data point to the query point. If the value of K is 2, it will check the two closest points and do a majority vote, and so on.

The value of K is not fixed, rather we need to decide on the optimal value of K for each dataset. This can be done by setting different values of K and then checking the accuracy of the algorithm.

We can then set the value of K to the value which gives us the best accuracy. Usually, as a thumb rule, we select odd values of K as this helps the algorithm take a majority vote. In general, a big value of k reduces the effect of the outliers but also causes the decision surface to generalize.

There are two important characteristics of K-NN.

• **Lazy Learning** – K-NN follows the principle of lazy learning. It does not have a specific training phase where it learns about the data. It uses all the training data while performing a classification operation.

• **Non-parametric Learning** – K-NN is a non-parametric algorithm as it does not assume anything about the distribution of the data. So KNN does not have to find any parameter for the distribution of data. The only hyperparameter that KNN

has is K, and that is provided by the user to the model.

Activity 4.1

Discuss how the K-Nearest Neighbours algorithm is different from decision trees.

3. Pros and Cons of using K-NN

K-NN has many advantages compared to other algorithms. Let's look at some of them below.

Simple and Intuitive – The K-NN algorithm is quite easy to understand and implement. Interpretability of the K-NN algorithm is also very high. To classify a new data point K-NN reads through whole dataset to find out K nearest neighbors and then takes a majority vote.

No Training Step – The K-NN algorithm does not explicitly have a training step. It simply reads all the data points during the prediction stage and makes a prediction.

Good for Multi-class problems – Many classification algorithms are easy to implement for binary classification problems but need extra effort to work for multi class problems. K-NN on the other hand adjusts well to multi-class problems without any additional steps.

Classification and Regression: One of the major pros of K-NN is that it can be used for solving classification and regression problems. Very few algorithms can be used for both problems.

No assumptions - K-NN does not assume anything about the data distribution because it is a non-parametric algorithm. Some models which are parametric like linear regression do make assumptions about the data before it can be implemented.

Even through K-NN has some advantages, it also has several disadvantages compared to other algorithms.

Slow and memory inefficient – One of the major drawbacks of K-NN is that it very slow and as the dataset grows the speed of the algorithm declines very fast. This is because K-NN needs to evaluate all the points to make a prediction. It is also quite a memory inefficient as the entire dataset needs to be stored in memory during the prediction stage.

Problems with Imbalanced data - K-NN does not have good accuracy when trained on imbalanced datasets. Let's say we have two classes, A and B, and the majority of the training data is labeled as A. In this case, the model will be biased towards class A. This will result in the incorrect classification of some objects from class B.

Sensitive to Outliers: The K-NN algorithm is very sensitive to outliers because it chooses the neighbors based

on distance criteria. If there are outliers in the data, especially near the decision boundary then the accuracy can become very low.

Curse of Dimensionality - KNN works well with a small number of input variables but as the numbers of variables grow K-NN algorithm struggles to predict the output of a new data point.

Distance Measurements - K-NN cannot be used for data that cannot be compared using a common distance technique like Euclidean or Manhattan distance. We need to have a way to compare distances between data points to use K-NN.

Activity 4.2

Discuss scenarios where K-NN would be useful and those in which it would not be useful.

4. Cross Validation

Cross Validation refers to a technique in which we reserve a particular portion of a dataset on which we do not train the model. After the training is over, we test the resulting model on this portion of the data before finalizing it.

The steps involved in cross validation are as follows -

1. Reserve a small portion of data set called validation data.

2. Train the model using the remaining dataset

3. Test the model on the validation data set and check its accuracy.

Cross validation helps us in gauging the effectiveness of our model's performance. If the model delivers high accuracy on validation data, we can go ahead and use the model for solving problems on real world data.

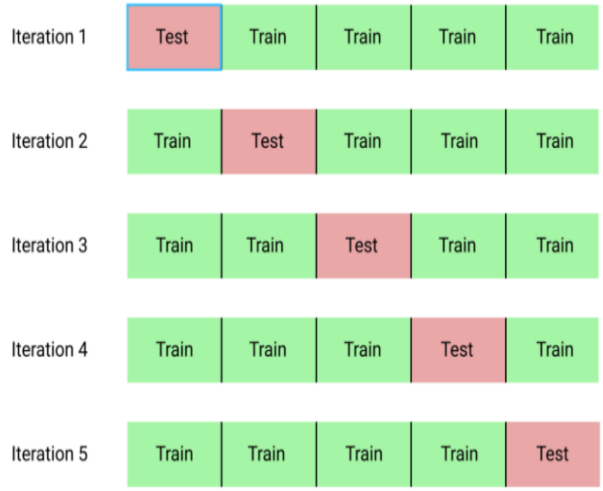
One popular way of doing cross validation is to use the k-fold cross-validation technique. In this technique, we split the data into k different but similar folds or sets.

We then perform k iterations, and, in each iteration, we choose one-fold as the validation set or test set and the rest as training sets. This helps us train the model better and avoid any bias as we are using the entire data for training and testing.

To determine the number of iterations `k` for cross-validation, you should consider a value such that each of the data samples is large enough to be statistically representative of the broader dataset.

If you are unsure as to which value of k should be chosen, then you can take k=10 as a thumb rule since it is common in the field of applied machine learning.

In the diagram given below, we perform a 5-fold validation.



Cross-validation might also be used to determine the value of K while using K-NN. In most cases, the accuracy is highest for K=7 to K=13 and falls as the value of K increases.

Recap

- The k-nearest neighbors (K-NN) algorithm is a one of the most basic and easy-to-implement supervised machine learning algorithm
- The K-NN algorithm works on the principal that similar things exist in close proximity to each other.
- Internally, the K-NN algorithm finds the distance between the query point and all the points in the data. It then selects 'K' closest and then performs a majority vote to decide the class which is most frequent.
- There are two important characteristics of K-NN - Lazy Learning and Non-parametric Learning.
- Cross Validation is a technique in which we reserve a portion of a dataset on which we do not train the model. After the training is over, we test the resulting model on this portion of the data.

Exercises

Objective Type Questions

Please choose the correct option in the questions below.

1) What are K-Nearest Neighbors used for?
   a. Regression
   b. Classification
   c. Both of the above
   d. None of the above
2) Which of the following statement is false?
   a. K-Nearest Neighbors uses proximity for prediction.
   b. K-Nearest Neighbors takes a majority vote on K data points.
   c. K-Nearest Neighbors can't be used for regression.
   d. None of the above
3. What are some of the advantages of K-NN?
   a. Easy to interpret
   b. No extra training step
   c. Lots of memory needed
   d. Both a and b
4. K-NN works well with imbalanced data sets.
   a. True
   b. False
5. Cross-validation helps us do the following.
   a. Removes bias
   b. Helps to test tune parameters
   c. Both a and b

Standard Questions

Please answer the questions below in no less than 100 words.

1. Write a short note on the advantages of using K-Nearest Neighbors.
2. How does K- Nearest Neighbors work internally?
3. What is cross-validation?
4. Write a short note on the disadvantages of using K-Nearest Neighbors.

Higher Order Thinking Skills(HOTS)

Please answer the questions below in no less than 200 words.

1. Describe how we can use K- Nearest Neighbors for multi-class classification.

2. How does cross-validation help us remove bias in a model?

Applied Project

In this exercise we will use R Studio to generate a K-NN model to analyze the famous iris dataset. The dataset contains150 records of the iris plant under five attributes - sepal length, sepal width, petal length, petal width. There are three different kinds of iris in the dataset and each type has 50 rows of data. The three types are – setosa, versicolor and virginica. The iris dataset is already present in R Studio and does not need to be loaded.

The objective of the exercise is to build a k-NN model which can correctly classify the iris flowers into the three groups - setosa, versicolor and virginica

Launch R Studio and follow the steps below.

1. First, let us look at the data to see the attributes, rows and columns. To do this, paste the code below in R Studio and click on Run.

```
require("class")
require("datasets")
data("iris")
str(iris)
```

You should get an output on the console as shown below.

```
> require("class") # load pre-installed package
Loading required package: class
> str(iris) #view structure of dataset
'data.frame':   150 obs. of  5 variables:
 $ Sepal.Length: num  5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...
 $ Sepal.Width : num  3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...
 $ Petal.Length: num  1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...
 $ Petal.Width : num  0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...
 $ Species     : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1 1 ...
```

2. Next, let us check the summary and top 5 rows of the iris dataset. Add the code below to the earlier code.

```
summary(iris)
head(iris)
```

```
> summary(iris) #view statistical summary of dataset
  Sepal.Length    Sepal.Width     Petal.Length    Petal.Width          Species
 Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100   setosa    :50
 1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.300   versicolor:50
 Median :5.800   Median :3.000   Median :4.350   Median :1.300   virginica :50
 Mean   :5.843   Mean   :3.057   Mean   :3.758   Mean   :1.199
 3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd Qu.:1.800
 Max.   :7.900   Max.   :4.400   Max.   :6.900   Max.   :2.500
> head(iris) #view top  rows of dataset
  Sepal.Length Sepal.Width Petal.Length Petal.Width Species
1          5.1         3.5          1.4         0.2  setosa
2          4.9         3.0          1.4         0.2  setosa
3          4.7         3.2          1.3         0.2  setosa
4          4.6         3.1          1.5         0.2  setosa
5          5.0         3.6          1.4         0.2  setosa
6          5.4         3.9          1.7         0.4  setosa
```

3. We now need to preprocess the dataset. Since we are performing classification, we need two sets of data - Training and Testing data (generally in 80:20 ratio).

We will divide the iris dataset into two subsets. Since the iris dataset is sorted by "Species" by default, we will first jumble the data rows and then take subset. Please use the code below in R Studio.

```
set.seed(99) # required to reproduce the results
rnum<- sample(rep(1:150)) # randomly generate numbers from 1 to 150
rnum
iris<- iris[rnum,] #randomize "iris" dataset
head(iris) #check if the dataset is random
```

You should get an output as shown below.

```
> head(iris) #check if the dataset is random
    Sepal.Length Sepal.Width Petal.Length Petal.Width    Species
88           6.3         2.3          4.4         1.3 versicolor
17           5.4         3.9          1.3         0.4     setosa
102          5.8         2.7          5.1         1.9  virginica
146          6.7         3.0          5.2         2.3  virginica
79           6.0         2.9          4.5         1.5 versicolor
141          6.7         3.1          5.6         2.4  virginica
```

This shows that the dataset has now been randomized. This will remove any bias in the model when we select a portion of the data for test and a portion for training.

4. We now need to normalize the data between 0 and 1 so that we can compare across the various features present in the dataset and we can also remove the need for units for the various lengths. Please use the code below in R Studio.

```
# Normalize the dataset between values 0 and 1
normalize <- function(x){
  return ((x-min(x))/(max(x)-min(x)))
}
iris.new<- as.data.frame(lapply(iris[,c(1,2,3,4)],normalize))
head(iris.new)
```

You should get an output as shown below to show that the data has been normalized.

```
> head(iris.new)
  Sepal.Length Sepal.Width Petal.Length Petal.Width
1    0.5555556   0.1250000    0.57627119   0.5000000
2    0.3055556   0.7916667    0.05084746   0.1250000
3    0.4166667   0.2916667    0.69491525   0.7500000
4    0.6666667   0.4166667    0.71186441   0.9166667
5    0.4722222   0.3750000    0.59322034   0.5833333
6    0.6666667   0.4583333    0.77966102   0.9583333
```

5. Next we need to make a subset of data for training and a subset for testing.

```
1. # subset the dataset
2. iris.train<- iris.new[1:130,]
3. iris.train.target<- iris[1:130,5]
4. iris.test<- iris.new[131:150,]
5. iris.test.target<- iris[131:150,5]
6. summary(iris.new)
```

You should get an output as shown below.

```
> summary(iris.new)
  Sepal.Length     Sepal.Width      Petal.Length     Petal.Width
 Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   Min.   :0.00000
 1st Qu.:0.2222   1st Qu.:0.3333   1st Qu.:0.1017   1st Qu.:0.08333
 Median :0.4167   Median :0.4167   Median :0.5678   Median :0.50000
 Mean   :0.4287   Mean   :0.4406   Mean   :0.4675   Mean   :0.45806
 3rd Qu.:0.5833   3rd Qu.:0.5417   3rd Qu.:0.6949   3rd Qu.:0.70833
 Max.   :1.0000   Max.   :1.0000   Max.   :1.0000   Max.   :1.00000
```

Next, we will create the K-NN algorithm. We will use the k value as 16. Now we can check the accuracy of the model by running it on test data.

```
model1<- knn(train=iris.train, test=iris.test, cl=iris.train.target, k=16)
table(iris.test.target, model1)
```

You should get the output below.

```
> table(iris.test.target, model1)
                 model1
iris.test.target setosa versicolor virginica
      setosa          5          0         0
      versicolor      0          7         1
      virginica       0          2         5
```

This result shows that al 5 setosa were correctly classified as setosa. Seven versicolor were correctly identified, one was labeled wrongly as virginica. Five virginica were correctly classified and two were wrongly labeled as versicolor.

Thus, this shows that our model is fairly accurate. You can experiment by changing the value of K and see if that improves the accuracy of the model.

CHAPTER

5

Regression Algorithms I

Studying this chapter should enable you to understand:

- What is Linear Regression?
- What is Mean Absolute Error?
- What is Mean Square Deviation?

1. Introduction

In the last two chapters, we learned about how to use decision trees and K-Nearest Neighbors algorithm for classification problems. In this chapter, we will look at another important problem called regression. We will also take a look at linear regression and two metrics that that help us make a good linear regression model.

2. Introduction to Linear Regression

Linear regression a way to explain the relationship between a variable y given the values of some other variable x. The target variable, y, is generally called the "dependent variable". The other variable x is called the "independent variable".

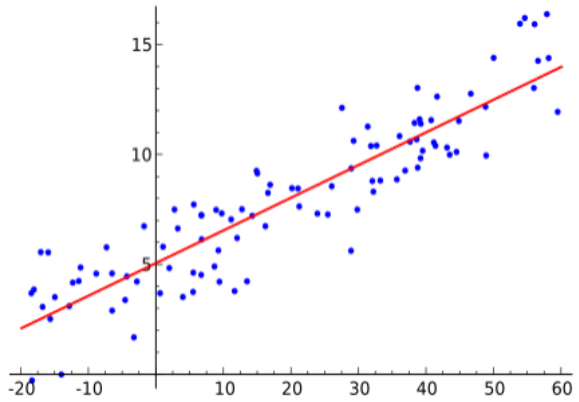
Regression, in general, is the problem of predicting the value of the dependent variable. Linear regression is known as "linear" because the relation of the dependent to the independent variables is a linear function of some parameters. Regression methods that do not have a linear function are called nonlinear regression models.

The term independent variable means that its value can be chosen at will, and the dependent variable will adjust based

on the value of the independent variable. Linear regression helps us predict how much that adjustment is going to be. Depending on the kind of relationship between the dependant and independent variable, the adjustment can be positive or negative or zero.

There are several real-life applications of linear regression. We can classify most applications fall into one of the following two broad categories. We usually use linear regression when we want to know:

- The nature and strength relationship between two variables (e.g. is there a positive relationship between rainfall and crop growth).
- The predicted value of the dependent variable for a given value of the independent variable (e.g. the amount of crop growth for a certain level of rainfall).



The diagram above shows a scatter plot of two variables x and y. The red line is known as the “line of best fit”. Linear regression helps us to find the line of best fit.

Once the line of best fit has been determined, we can easily say that the variables x and y have a positive relationship, that is y increases as x increases. We can also find the value of y at any value of x from the equation of the line of best fit.

The equation for a simple linear regression is of the form

$$Y = m*X + b$$

where

Y is Dependent Variable,

X is an Independent Variable,

b is intercept and

m is slope.

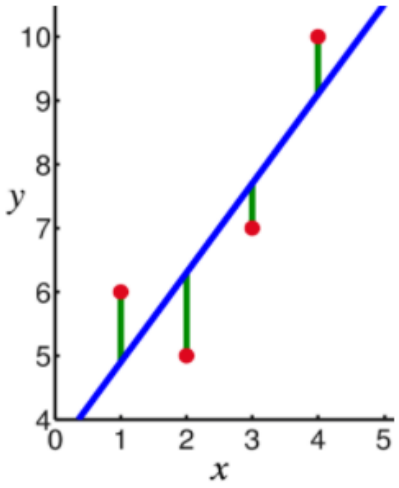
Activity 5.1

Create your own decision tree for a daily activity based on the steps above.

3. Mean Absolute Error

The basic objective of linear regression is to try to reduce the vertical distance between the line and the data points to make it minimum. This process is called "fitting the line to the data." To do so we can use the Mean Absolute Error (MAE).

While figuring out the line of best fit, we want to minimize the deviation of the line from the actual data points. We do this by finding the mean absolute error and minimizing it.



In the diagram above, the red dots are observed values, the blue line is the line of best fit and the green lines represent the errors or residuals.

Mean Absolute Error measures the average magnitude of the errors in a set of predictions, without considering their direction.

4. Root Mean Square Deviation

The Root Mean Square Deviation is used to determine how close the observed points are to the model's predicted values. Mathematically, the Root Mean Square Deviation is the square root of the variance of the residuals.

In real-life scenarios, it is best if the RSME value is small. A small RSME value means that the model is a better bit to the data and thus more accurate. A large RSME value shows that the model is not a good fit and might need to be retrained.

The actual value of the RSME depends on the data and degree of accuracy required. For example, a RSME of 1 cm might not be significant for designing a building but will be very significant for designing a precision tool.

Recap

- Linear regression a way to explain the relationship between a variable y given the values of some other variable x..
- Regression, in general, is the problem of predicting the value of the dependent variable.
- The equation for a simple linear regression is of the form y=mx+b.
- Mean Absolute Error measures the average magnitude of the errors in a set of predictions, without considering their direction.
- The Root Mean Square Deviation is used to determine how close the observed points are to the model's predicted values.
- In real life scenarios, it is best if the RSME value is small.

Exercises

Objective Type Questions

Please choose the correct option in the questions below.

1. Regression can be performed for which kind of variables:
   a. Continuous
   b. Discrete
   c. Categorical
   d. All of the above

2. Which of the following statement is false?

   a. Regression can be done for only two variables.
   b. Regression is a supervised learning technique.
   c. Regression can be done for categorical variables.
   d. None of the above

3. Which of the following are used for regression?
   a. K-NN
   b. Linear Regression
   c. Both of the above

4. The value of RSME is usually greater than the value of MAE

   a. True
   b. False

5. MEA considers the direction of the residuals

   a. True
   b. False

Standard Questions

Please answer the questions below in no less than 100 words.

1. In your own words, write down the steps to create a line of best fit.
2. What is the application of root mean square deviation?
3. Write a short note on mean absolute error.

Higher Order Thinking Skills(HOTS)

Please answer the questions below in no less than 200 words.

1. Which is a better metric - Mean Absolute Error or Root Mean Square Deviation? Discuss.
2. What are some of the drawbacks of using linear regression?

Applied Project

Cloth manufacturers need to create patterns for clothing so that the clothes are likely to fit their buyers. For the clothes to fit people well, designers must understand the relationship that exists between the different parts of the human body. For example, a shirt designer must take into consideration the length of a person's arm with the length of their upper body.

The question we need to answer is - Is forearm length relate to height? If so, can we use for forearm length to predict height?

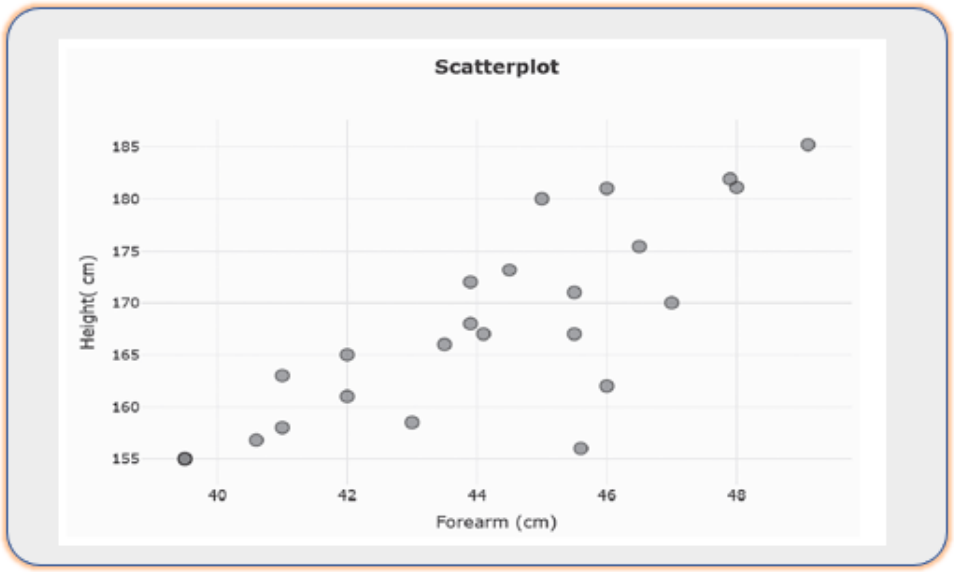
Consider the following data as the basis for this assignment.

Forearm (cm)	Height (cm)	Forearm (cm)	Height (cm)
45	180	41	163
44.5	173.2	39.5	155
39.5	155	43.5	166
43.9	168	41	158
47	170	42	165
49.1	185.2	45.5	167
48	181.1	46	162
47.9	181.9	42	161
40.6	156.8	46	181
45.5	171	45.6	156
46.5	175.5	43.9	172
43	158.5	44.1	167

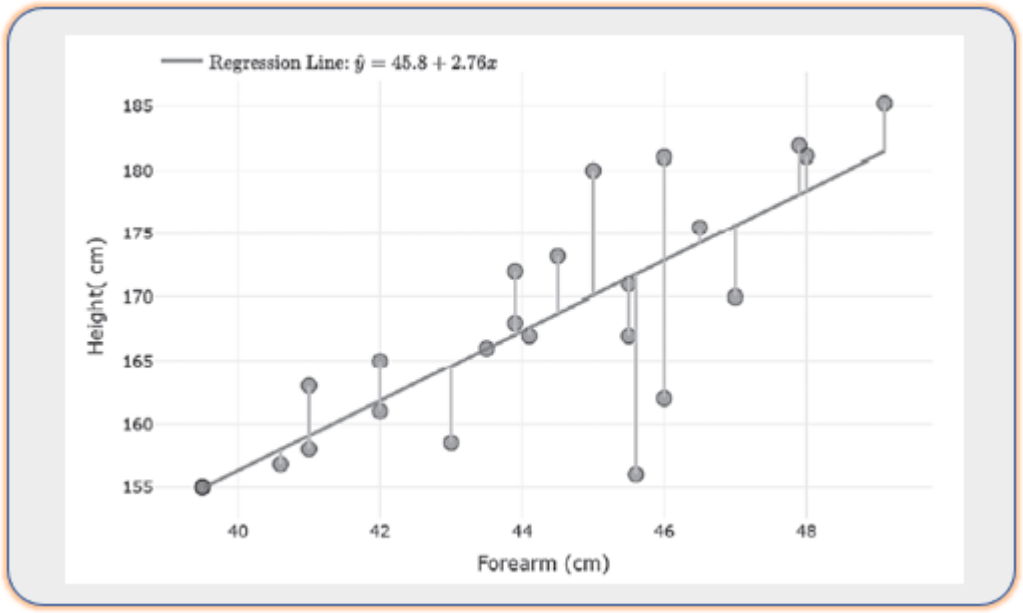
Using the data above, we will try to establish a relationship between a person's forearm and height.

We can first create a scatter plot to see if the two variables forearm and height have any correlation between them. By creating a scatter plot as shown below, we can see that

there is a strong positive relationship between the two variables. As the height is increasing, the forearm length also continues to increase.



We can then draw the line of best fit and using linear regression and calculate the equation of the line. The equation of the line of best fit can then be used to predict the forearm length for any given height and vice-versa.



Thus, we can see how linear regression is used in real-life scenarios.

CHAPTER

6

Regression Algorithms II

Studying this chapter should enable you to understand:

- What is Multiple Linear Regression?
- What is Non-Linear Regression?

1. Introduction

In the last chapter, we learned about how to use linear regression for regression problems. In this chapter, we will see how we can solve regression problems using another important technique called Non-Linear Regression. We will also take a look multiple linear regression and how that helps us.

2. Multiple Linear Regression

You already know what linear regression is. Multiple Linear Regression uses multiple independent variables to predict the outcome of a dependent variable. For example, effects of age, weight and height on cholesterol levels of an individual. Here, age, weight and height are independent variables and cholesterol level is dependent variable because it is dependent on the factors age, height and weight.

A simple regression equation has an intercept on the right-hand side and an explanatory variable with a coefficient. A multiple regression has multiple variables on the right-hand side, each with its slope coefficient.

The basic model of multiple linear regression is

$Yi = \beta 0 + \beta 1Xi1 + \beta 2Xi2 + \beta 3Xi3 + \ldots + \beta pXip + \epsilon i$

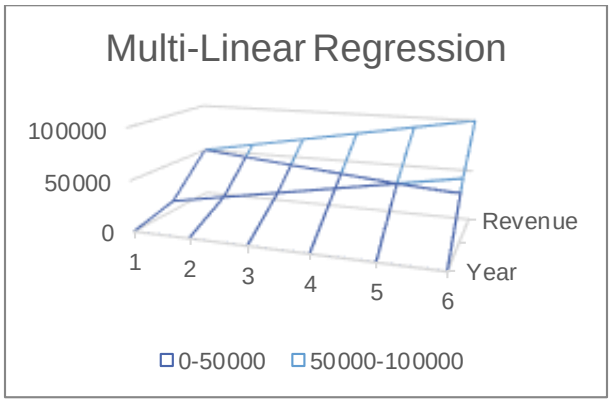
where i = 1, 2, 3, …, n for each observation.

In the above formula, we consider n number of observations of one dependent variable and p number of independent variables.

Therefore, Yi is the ith observation of the jth independent variable where j = 1, 2, 3, …, p. The values βj represent the features to be estimated and ϵi is the ith independent identically distributed normal error. In more general multivariate linear regression, the above observations can be defined into one equation.

$Yij = \beta 0j + \beta 1jXi1 + \beta 2jXi2 + \beta 3jXi3 + \ldots + \beta pjXip + \epsilon i$

for all observations indexed as i = 1, 2, …, n and for all dependent variables indexed as j = 1, 2, …, m.



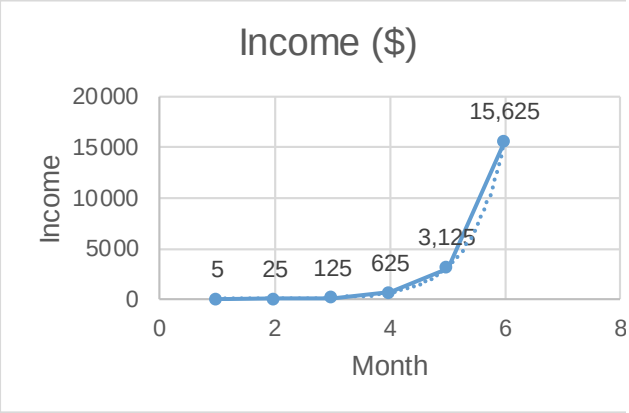
3. Non-linear Regression

Non-linear regression is more flexible than linear regression. The formula for non-linear regression is y ~ f (x, β) where x is a vector of independent variables and y is the dependent variable. These functions are called non-linear functions.

Examples of non-linear functions include exponential functions, logarithmic functions, trigonometric functions, power functions, etc.

The graph of linear regression follows the equation of line and the graph of non-linear regression follows equation of a curve. In the formula, f (x, β) varies depending on the type of curve.

For example, Andy started a business 5 months back. He observed that his income increases exponentially every month. We will try to predict what would be his income next month using non-linear regression.



From the graph, we can see that here f (x, β) = 5x. So, the equation for this non-linear graph in this case will be y = 5x. Therefore, we can predict that the income in the 6th month would be 56 = 15,625.

Recap

- Multiple Linear Regression uses multiple independent variables to predict the outcome of a dependent variable.
- Non-linear regression is more flexible than linear regression.
- Basically, the graph of linear regression follows equation of line and the graph of non-linear regression follows equation of a curve.

Exercises

Objective Type Questions

Please choose the correct option in the questions below.

1. Non-Linear Regression can be performed for which kind of variables:
   a. Continuous
   b. Discrete
   c. Categorical
   d. All of the above

2. Which of the following statement is false?

   a. The equation for Non-Linear Regression follows the equation of a line.
   b. The equation for Non-Linear Regression follows the equation of a curve.
   c. None of the above

3. Which of the following are used for regression?

   a. K-NN
   b. Linear Regression
   c. Both of the above

Standard Questions

Please answer the questions below in no less than 100 words.

1. What is the difference between multiple linear regression and non-linear regression?
2. What are the steps for performing non-linear regression?
3. Write a short note on non-linear regression.

Higher Order Thinking Skills(HOTS)

Please answer the questions below in no less than 200 words.

1. In your own words, write down a real-world scenario where multiple linear regression is used.
2. What are the advantages of using non-linear regression?

Applied Project

Write a note on the comparison of multiple linear regression and logistic regression.

CHAPTER

7

Unsupervised Learning

Studying this chapter should enable you to understand:

- What is unsupervised learning?
- What are the real-world applications of unsupervised learning?
- What is clustering and k-means clustering?
- What is k-means clustering?

1. Introduction

In the last few chapters, we learned about how to use to solve classification and regression problems using supervised learning techniques. In this chapter, we shall learn about another important branch of machine learning called unsupervised learning. We will also take a look at some real-world scenarios on how that helps us.

2. Introduction to Unsupervised Learning

Supervised learning, as we have seen earlier, is a type of machine learning process in which algorithms are trained using data that is well labeled. This means that we need to have some data is already tagged with the correct labels which we can use for training.

After that, we use the trained algorithm to make predictions on new sets of data either for classification or regression problems.

Unsupervised learning is the process in which the algorithms are not trained using data that is classified or labeled. In unsupervised learning, algorithms act on data without human intervention.

These algorithms discover hidden patterns or data groupings with their ability to discover similarities and differences in information. This makes unsupervised learning algorithms ideal for solving real-life problems like exploratory data analysis, customer segmentation, and image recognition.

Activity 6.1

If unsupervised algorithms can discover hidden pattern automatically why do we need supervised algorithms? Discuss.

3. Real-world applications of Unsupervised Learning

Machine learning is now being used by several organizations to improve the user experience for their products. With the help of unsupervised learning, we can easily get an exploratory view of the raw data without having to perform a lot of analysis on it. This helps businesses to identify patterns in large volumes of data more quickly as compared to using manual observation techniques. There are several real-world applications of unsupervised learning and more are being discovered each day. Some of the common applications of unsupervised learning have been discussed below.

1. **Recommendation Engines**: Many websites selling products use recommendation engines to predict what products a customer is likely to purchase. This is done by using past purchase behavior data and unsupervised learning techniques which can help to discover trends in the data. These predictions are then used to make add-on recommendations relevant to a particular customer during the checkout process.

2. **Medical imaging**: Unsupervised machine learning provides essential features to medical imaging devices, such as image detection, classification, and segmentation, used in radiology and pathology to diagnose patients quickly and accurately.

3. **Anomaly detection**: Anomaly detection is also an important application for unsupervised learning. Anomalies can be useful for detecting fraud in financial systems or other security applications. Unsupervised learning models can go through large amounts of raw data and find unusual data points within a dataset. These unusual data points can then be analyzed manually to see if there has indeed been a fraud or security breach.

4. **Customer personas**: Customer personas are used to understand common purchasing habits and purchasing times for customers of a product. With the help of unsupervised

learning, organizations can build better buyer persona profiles. This, in turn, enables them to align their sales and ads to such customer segments more appropriately.

5. **News Sections**: Some online news websites use unsupervised learning techniques to categorize articles from various online news outlets to put them under various sections like sports, entertainment, international news, etc.

Activity 6.2

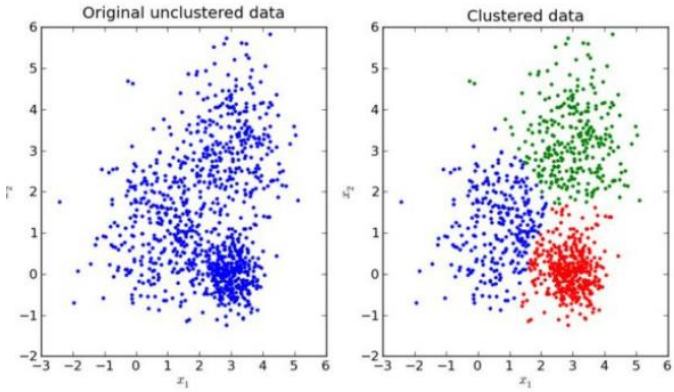
Check out an e-commerce store and try to find if recommendation engines are being used.

4. Introduction to Clustering

Clustering refers to the process of dividing entire raw data into several groups such that the data points in one group are similar to other data points in the same group but different from those in other groups. Clustering techniques apply when there is no class to be predicted but rather when the instances are to be divided into natural groups.

As shown in the diagram below, the input for a clustering algorithm is the original raw data and the output is a well-clustered data set with three distinct clusters.

We can understand the clustering technique with the help of real-world examples. When we visit any supermarket, we can see that different



item are grouped in different areas. For example, vegetables are in one section while fruits are grouped into a separate section. The clustering technique also works in the same way.

There are many ways to perform clustering. Some of the main clustering methods are -

- Partitioning Clustering
- Density-Based Clustering
- Distribution Model-Based Clustering
- Hierarchical Clustering

5. K - Means Clustering

The k-means clustering method is a technique that is used to spot clusters of data classes in a dataset. It is an unsupervised machine learning technique. Among the different types of clustering methods available, the k-

means clustering is one of the easiest and easy to understand clustering algorithms.

Usually, the k-means require only a few steps. Firstly, the algorithm randomly selects centroids, equal to the number of clusters that the user chooses. The approach the k-means algorithm follows is to solve the problem which is called Expectation-Maximization.

The k-means clustering algorithm works is as follows:

1 First, specify the number of clusters K depending on the input.
2 Initialize the centroids by shuffling the data points and then selecting K data points for the centroids randomly.
3 Do iterations till the point that there is no change to the centroids such that the assignment of data points to clusters isn't changing.
4 Compute the sum of the squared distance between data points and all centroids.
5 Classify or mark each data point to the cluster it is closest to.

The k-means algorithm is very popular and used in a variety of real-world applications such as market segmentation, document clustering and image segmentation.

Recap

- Supervised learning is a type of machine learning process in which algorithms are trained using data which is well labeled.
- Unsupervised learning is a type of machine learning process in which the algorithms are not trained using date that is classified or labeled.
- Machine learning is now being used by several organizations in order to improve the user experience for their products.
- Clustering techniques apply when there is no class to be predicted but rather when the instances are to be divided into natural groups.
- K-means clustering is one of the easiest and easy to understand clustering algorithms.

Exercises

Objective Type Questions

Please choose the correct option in the questions below.

1. Which of the following are applications of unsupervised learning:

    a. Medical Imaging
    b. Anomaly detection
    c. News
    d. All of the above

2. Which of the following statement is false?
    a. Unsupervised learning is done on unlabeled data sets.
    b. Unsupervised learning needs human intervention.
    c. Unsupervised learning is a continuous process.
    d. None of the above

3. What are some of the clustering techniques?
    a. Partition based clustering
    b. Density based clustering
    c. Both a and b
4. Unsupervised learning can be used for classification.
    a. True
    b. False
5. Unsupervised learning can help us find out irrelevant data.
    a. True
    b. False

Standard Questions

Please answer the questions below in no less than 100 words.

1. Write a short note on clustering.
2. How does a K-Means Clustering algorithm work?
3. Describe three applications of unsupervised learning.
4. What are recommendation engines and why are they used?

Higher Order Thinking Questions (HOTS)

Please answer the questions below in no less than 200 words.

1. What are some of the well-known clustering methods?
2. How does unsupervised learning help with anomaly detection?

Applied Project

Discuss how unsupervised learning can be used to make self-driving cars.

CHAPTER

8

Final Project I

Studying this chapter should enable you to understand:

- How to use Visual Studio code
- How to use Python for exploratory data analysis
- How to combine data from different sources
- How to make a predictor function with Python

1. Introduction

In the previous chapters, we learned about how to solve classification and regression problems. In this project, we will see how we can make predictions using historical data. We will also take a look at some important Python functions and how they help us.

2. Introduction to the Project

Meteors are celestial bodies that are visible from Earth every night. The best meteor showers are the ones that originate from one of the comets that orbit around our sun.

To see meteor showers, we must know the path of the comets and also consider the side of the Earth where the meteors will enter our atmosphere. We should also check if the night sky will be dark enough so that we can view the meteor trails.

In this project, we will predict whether or we can see meteor showers from any city by using historical data.

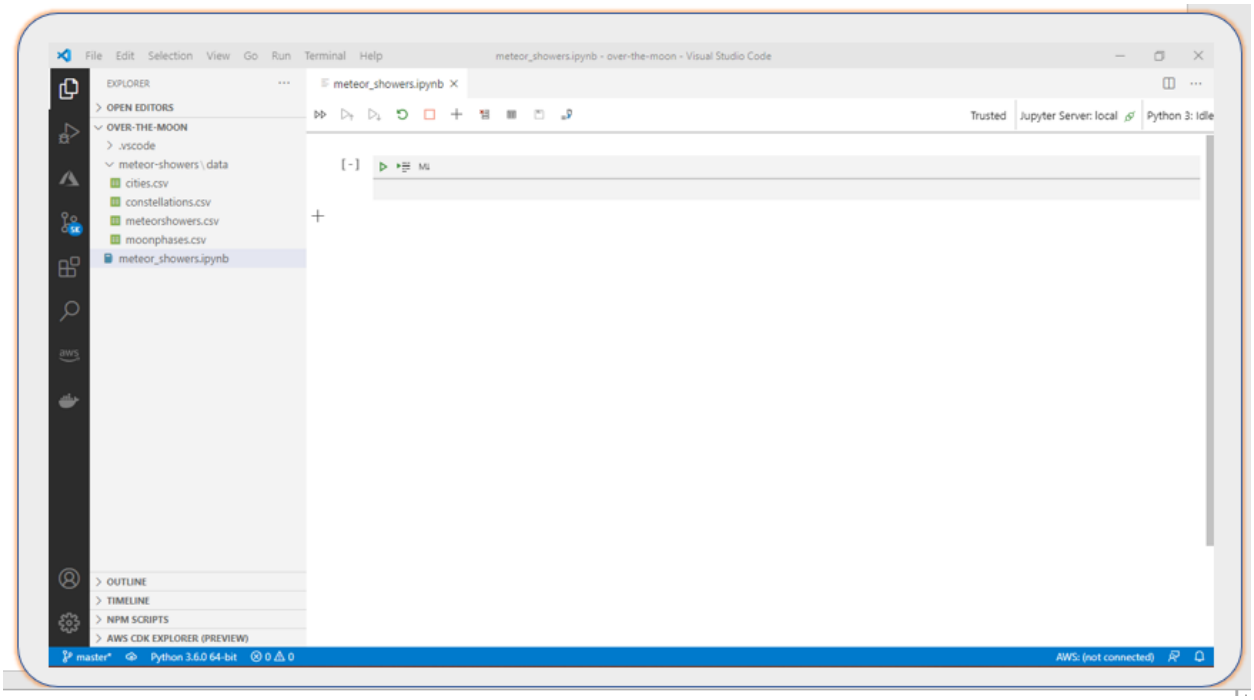
3. Setup Visual Studio Code and Python

To complete this exercise, you will need to install Python and Visual Studio Code. You will also need to install Miniconda.

After you install everything, follow these steps to prepare your environment:

1. Create a folder on your computer called over-the-moon.
2. Open the folder in Visual Studio Code.
3. Create a folder inside the over-the-moon folder. Name it meteor-showers.
4. Create a folder inside the meteor-showers folder. Name its data.
5. Create a file called meteor-showers.ipynb.
6. Open the meteor-showers.ipynb file in Visual Studio Code.
7. Make sure you're using the conda environment that you set up earlier.

After the setup, your VS Code should look like the screenshot below.



4. Gather data for the meteor showers

Before we start looking at the data, let us quickly learn about the comets and meteors that we will be tracking in this project. We should also have an idea about the phases of the moon so that we know the ideal time to look for the meteors.

We know that there are several comets in the solar system. However, for this project, we will focus on a few famous comets that are often observed by astronomers.

1. Comet Thatcher

Comet Thatcher was first discovered in 1861. It takes this comet 415.5 years to go around the sun. The Lyrids meteor shower which can be seen in April is created from the debris of this comet.

2. Comet Halley

Comet Halley was first discovered in 1531. It takes this comet 76 years to go around the sun. The Eta Aquarids meteor shower which can be seen in May is created from the debris of this comet. The meteor shower from Orionids which can be sighted in October is also from the debris of this comet.

3. Comet Swift-Tuttle

Comet Swift-Tuttle was first discovered in 1862. It takes this comet 133 years to go around the sun. The Perseids meteor shower which can be seen in August is created from the debris of this comet. The Perseids meteor shower appears to come from the direction of the constellation Perseus.

4. Comet Tempel-Tuttle

Comet Tempel-Tuttle was discovered separately in 1865 and 1866. It takes this comet 33 years to go around the sun. The Leonids meteor shower which can be seen in November is created from the debris of this comet. Sometimes, the Leonids meteor shower can also become a meteor storm.

Phases of the moon

When the moon goes around the Earth, different amounts of the light of the sun are reflected from the Moon to Earth. This change in the number of light cases the size of the moon to change every day during the month. The phases of the Moon have been named depending on the amount of light reflecting from the moon.

The different phases of the Moon are as follows:

- New Moon
- Waxing crescent
- First-quarter
- Waxing gibbous
- Full Moon

Microsoft

- Waning gibbous
- Third-quarter
- Waning crescent

Data files

The data for this project is available on GitHub on this link - https://github.com/sguthals/learnwithdrg/tree/main/OverTheMoon/meteor-showers

You can try to collect additional data that you think will help you improve the predictions of meteor showers.

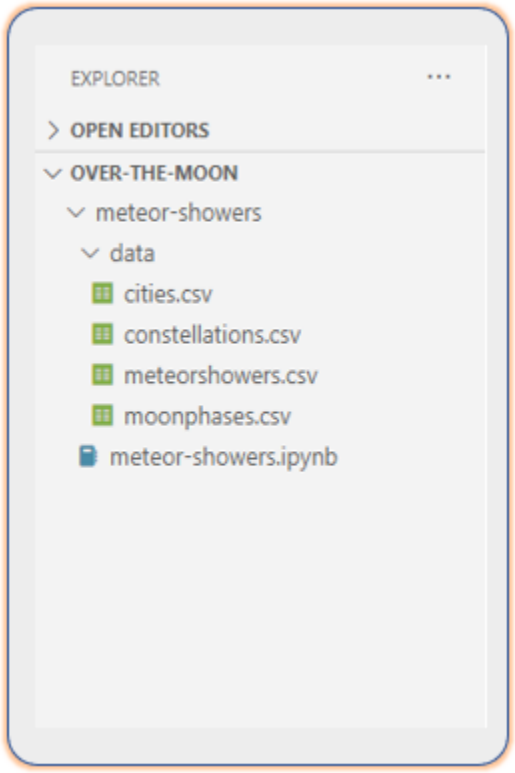
There are 4 data files for this project as described below.

moonphases.csv - This file contains the Moon phases for every day of 2020. The missing data will be added in the next unit.

meteorshowers.csv - This file contains data for each of the five meteor showers that we described earlier. Data includes their preferred viewing month, the months when they're visible, and the preferred hemisphere for viewing.

constellations.csv - This file contains data for the four constellations that are radiant for the five meteor showers. Data includes the latitudes for which they're visible and the month for the best viewing.

cities.csv - This file contains a list of country capitals and their associated latitudes.

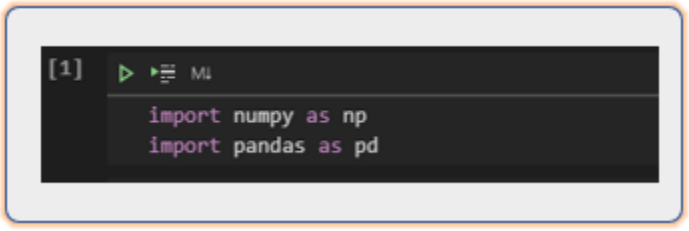


5. Cleanse meteor data

Let us now use Python on the Jupyter notebook that we created earlier to explore the data and make any corrections if needed.

To follow along, input each code snippet in a cell in your Jupyter notebook and click on the green button to run the code as shown below.

```
import numpy as np
import pandas as pd
```



```
meteor_showers = pd.read_csv('data/meteorshowers.csv')
moon_phases = pd.read_csv('data/moonphases.csv')
constellations = pd.read_csv('data/constellations.csv')
cities = pd.read_csv('data/cities.csv')
```

```
change_meteor_shower = {'name':'Chang\'e','radiant':'Draco','bestmonth':'o
ctober','startmonth':'october','startday':1,'endmonth':'october','endday':
31,'hemisphere':'northern','preferredhemisphere':'northern'}

meteor_showers = meteor_showers.append(change_meteor_shower, ignore_index=
True)
```

```
draco_constellation = {'constellation':'Draco','bestmonth':'july','latitud
estart':90,'latitudeend':-15,'besttime':2100,'hemisphere':'northern'}

constellations = constellations.append(draco_constellation, ignore_index=T
rue)
```

```
meteor_showers.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 6 entries, 0 to 5
Data columns (total 9 columns):
name                    6 non-null object
radiant                 6 non-null object
bestmonth               6 non-null object
startmonth              6 non-null object
startday                6 non-null int64
endmonth                6 non-null object
endday                  6 non-null int64
hemisphere              6 non-null object
preferredhemisphere     6 non-null object
dtypes: int64(2), object(7)
memory usage: 504.0+ bytes
```

```
moon_phases.head()
```

	month	day	moonphase	specialevent
0	january	1	NaN	NaN
1	january	2	first quarter	NaN
2	january	3	NaN	NaN
3	january	4	NaN	NaN
4	january	5	NaN	NaN

```
constellations.head()
```

	constellation	bestmonth	latitudestart	latitudeend	besttime	hemisphere
0	Lyra	august	90	-40	21:00	northern
1	Aquarius	october	65	-90	21:00	southern
2	Orion	january	85	-75	21:00	northern
3	Perseus	december	90	-35	21:00	northern
4	Leo	april	90	65	21:00	northern

```
cities.head()
```

	city	latitude	country
0	Abu Dhabi	24.47	United Arab Emirates
1	Abuja	9.07	Nigeria
2	Accra	5.55	Ghana
3	Adamstown	-25.07	Pitcairn Islands
4	Addis Ababa	9.02	Ethiopia

```
months = {'january':1, 'february':2, 'march':3, 'april':4, 'may':5, 'june'
:6, 'july':7, 'august':8, 'september':9, 'october':10, 'november':11, 'dec
ember':12}
meteor_showers.bestmonth = meteor_showers.bestmonth.map(months)
meteor_showers.startmonth = meteor_showers.startmonth.map(months)
meteor_showers.endmonth = meteor_showers.endmonth.map(months)
moon_phases.month = moon_phases.month.map(months)
constellations.bestmonth = constellations.bestmonth.map(months)
```

```
meteor_showers.head()
```

	name	radiant	bestmonth	startmonth	startday	endmonth	endday	hemisphere	preferredhemisphere
0	Lyrids	Lyra	4	4	21	4	22	northern	northern
1	Eta Aquarids	Aquarius	5	4	19	5	28	northern, southern	southern
2	Orionids	Orion	10	10	2	11	7	northern, southern	northern, southern
3	Perseids	Perseus	8	7	14	8	24	northern	northern
4	Leonids	Leo	11	11	6	11	30	northern, southern	northern, southern

```
meteor_showers['startdate'] =
pd.to_datetime(2020*10000+meteor_showers.startmonth*100+meteor_showers.sta
rtday,format='%Y%m%d')  meteor_showers['enddate'] =
```

```
pd.to_datetime(2020*10000+meteor_showers.endmonth*100+meteor_showers.endda
y,format='%Y%m%d')
```

```
moon_phases['date'] = pd.to_datetime(2020*10000+moon_phases.month*100+moon
_phases.day,format='%Y%m%d')
```

```
hemispheres = {'northern':0, 'southern':1, 'northern, southern':3}
meteor_showers.hemisphere = meteor_showers.hemisphere.map(hemispheres)
constellations.hemisphere = constellations.hemisphere.map(hemispheres)
```

```
phases = {'new moon':0,'third quarter':0.5, 'first quarter':0.5,'full moon
':1.0}
moon_phases['percentage'] = moon_phases.moonphase.map(phases)
moon_phases.head()
```

	month	day	moonphase	specialevent	date	percentage
0	1	1	NaN	NaN	2020-01-01	NaN
1	1	2	first quarter	NaN	2020-01-02	0.5
2	1	3	NaN	NaN	2020-01-03	NaN
3	1	4	NaN	NaN	2020-01-04	NaN
4	1	5	NaN	NaN	2020-01-05	NaN

```
meteor_showers = meteor_showers.drop(['startmonth', 'startday', 'endmonth'
, 'endday', 'hemisphere'], axis=1)
moon_phases = moon_phases.drop(['month','day','moonphase','specialevent'],
axis=1)
constellations = constellations.drop(['besttime'], axis=1)
```

```
lastPhase = 0

for index, row in moon_phases.iterrows():
    if pd.isnull(row['percentage']):
        moon_phases.at[index,'percentage'] = lastPhase
    else:
        lastPhase = row['percentage']
```

Microsoft

6. Write the predictor function

```
def predict_best_meteor_shower_viewing(city):
    # Create an empty string to return the message back to the user
    meteor_shower_string = ""

    if city not in cities.values:
        meteor_shower_string = "Unfortunately, " + city + " isn't availabl
e for a prediction at this time."
        return meteor_shower_string

    # Get the latitude of the city from the cities dataframe
    latitude = cities.loc[cities['city'] == city, 'latitude'].iloc[0]

    # Get the list of constellations that are viewable from that latitude
    constellation_list = constellations.loc[(constellations['latitudestart
'] >= latitude) & (constellations['latitudeend'] <= latitude), 'constellat
ion'].tolist()

    # If no constellations are viewable, let the user know
    if not constellation_list:
        meteor_shower_string = "Unfortunately, there are no meteor showers
viewable from "+ city + "."

        return meteor_shower_string

    meteor_shower_string = "In " + city + " you can see the following mete
or showers:\n"

    # Iterate through each constellation that is viewable from the city
    for constellation in constellation_list:
        # Find the meteor shower that is nearest that constellation
        meteor_shower = meteor_showers.loc[meteor_showers['radiant'] == co
nstellation, 'name'].iloc[0]

        # Find the start and end dates for that meteor shower
        meteor_shower_startdate = meteor_showers.loc[meteor_showers['radia
nt'] == constellation, 'startdate'].iloc[0]
        meteor_shower_enddate = meteor_showers.loc[meteor_showers['radiant
'] == constellation, 'enddate'].iloc[0]

        # Find the moon phases for each date within the viewable timeframe
of that meteor shower
        moon_phases_list = moon_phases.loc[(moon_phases['date'] >= meteor_
shower_startdate) & (moon_phases['date'] <= meteor_shower_enddate)]

        if meteor_shower == 'Chang\'e':
```

```
            # For the film meteor shower, find the date where the moon is
the most visible
            best_moon_date = moon_phases_list.loc[moon_phases_list['percen
tage'].idxmax()]['date']

            # Add that date to the string to report back to the user
            meteor_shower_string += "Though the moon will be bright, the "
+ meteor_shower + " is best seen if you look towards the " + constellation
+ " constellation on " +  best_moon_date.to_pydatetime().strftime("%B %d,
%Y") + ".\n"
        else:
            # Find the first date where the moon is the least visible
            best_moon_date = moon_phases_list.loc[moon_phases_list['percen
tage'].idxmin()]['date']

            # Add that date to the string to report back to the user
            meteor_shower_string += meteor_shower + " is best seen if you
look towards the " + constellation + " constellation on " +  best_moon_dat
e.to_pydatetime().strftime("%B %d, %Y") + ".\n"

    return meteor_shower_string
```

```
print(predict_best_meteor_shower_viewing('Beijing'))
```

```
In Beijing you can see the following meteor showers:
Lyrids is best seen if you look towards the Lyra constellation on April 22, 2020.
Eta Aquarids is best seen if you look towards the Aquarius constellation on April 22, 2020.
Orionids is best seen if you look towards the Orion constellation on October 16, 2020.
Perseids is best seen if you look towards the Perseus constellation on July 20, 2020.
Though the moon will be bright, the Chang'e is best seen if you look towards the Draco constellation on October 01, 2020.
```

This is the final output for the project. You have now successfully predicted when the meteors will be seen in the city of Beijing.

You can also try to predict other cities across the world like New Delhi and Abu Dhabi.

Recap

- In this exercise you learnt how to use Visual Studio Code and Python.
- You also understood how to combine data from multiple sources and create a predictor function in Python.

CHAPTER

9

Final Project II

Studying this chapter should enable you to understand:

- How to use Pandas for data analysis
- How to use machine learning to impute missing values
- How to use the CodeTour extension of VS Code
- How to predict player stats

1. Introduction

In the previous project, we learned how to clean raw data and make predictions based on historic data. In this project, which is inspired by the new film *Space Jam: A New Legacy*, we will see how basketball stats can help us to gain an understanding of data science and coding.

2. Introduction to the Project

In this project, we will use some tools and techniques from Data Science and machine learning as given below.

- Use Python, Pandas, and Visual Studio Code to understand basketball stats
- Use machine learning to cleanse and impute missing data from datasets
- Discover how to identify bimodal data across human and Tune Squad basketball players

- Explore the CodeTour extension in Visual Studio Code for code writing guidance

To complete the project, the students must complete the 12 units of the course as given below. Each module builds on the previous module and introduces new concepts to the students gradually so please complete the modules in the given order.

The twelve modules are as follows.

1. Introduction
2. Set up your local environment for data science coding
3. Data cleansing part 1 - Find missing values
4. Data cleansing part 2 - Drop columns and rows
5. Data exploration part 1 - Check for outliers
6. Data exploration part 2 - Check the distribution of the data
7. Data exploration part 3 - Discover data that represents more than one population
8. Data manipulation part 1 - Add qualifying player information
9. Data manipulation part 2 - Impute missing values for columns
10. Data manipulation part 3 - Impute missing values by using machine learning
11. Knowledge check
12. Summary

Please visit this link to access the modules.

https://docs.microsoft.com/en-us/learn/modules/predict-basketball-player-efficiency-ratings/1-introduction

Recap

- In this exercise you learnt how to use Pandas for data science.
- You also understood techniques to find out missing values.

.

References

Introduction to Process Modeling. 2021. Introduction to Process Modeling. [ONLINE] Available at: https://www.itl.nist.gov/div898/handbook/pmd/section1/pmd1.htm. [Accessed 03 March 2021].

Bargagliotti, A., Franklin, C., Arnold, P., and Gould, R., 2020. Pre-K-12 Guidelines for Assessment and Instruction in Statistics Education II (GAISE II). American Statistical Association

sguthals. 2021. Introduction - Learn | Microsoft Docs. [ONLINE] Available at: https://docs.microsoft.com/en-us/learn/modules/predict-basketball-player-efficiency-ratings/1-introduction. [Accessed 03 March 2021].

Freedman, D., 2009. Statistical Models. Cambridge University Press.