Module 4: Exploratory Data Analysis

4.1 Need of Exploratory data analysis

Originally developed by American mathematician John Tukey in the 1970s, EDA techniques continue to be a widely used method in the data discovery process today.

Exploratory data analysis (EDA) is used by data scientists to analyze and investigate data sets and summarize their main characteristics, often employing data visualization methods.

It helps determine how best to manipulate data sources to get the answers you need, making it easier for data scientists to discover patterns, spot anomalies, test a hypothesis, or check assumptions.

EDA is primarily used to see what data can reveal beyond the formal modeling or hypothesis testing task and provides a provides a better understanding of data set variables and the relationships between them.
It can also help determine if the statistical techniques you are considering for data analysis are appropriate.
It is crucial to understand it in depth before you perform data analysis and run your data through an algorithm. You need to know the patterns in your data and determine which variables are important and which do not play a significant role in the output.

some variables may have correlations with other variables. You also need to recognize errors in your data.

All of this can be done with Exploratory Data Analysis. It helps you gather insights and make better sense of the data, and removes irregularities and unnecessary values from data.

- Helps you prepare your dataset for analysis.
- Allows a machine learning model to predict our dataset better.
- Gives you more accurate results.
- It also helps us to choose a better machine learning model

4.2 Cleaning and preparing the data

Data cleaning is the process of fixing or removing incorrect, corrupted, incorrectly formatted, duplicate, or incomplete data within a dataset. When combining multiple data sources, there are many opportunities for data to be duplicated or mislabeled. If data is incorrect, outcomes and algorithms are unreliable, even though they may look correct. There is no one absolute way to prescribe the exact steps in the data cleaning process because the processes will vary from dataset to dataset. But it is crucial to establish a template for your data cleaning process so you know you are doing it the right way every time.

Difference between data cleaning and data transformation

Data cleaning is the process that removes data that does not belong in your dataset. Data transformation is the process of converting data from one format or structure into another. Transformation processes can also be referred to as data wrangling, or data munging, transforming and mapping data from one "raw" data form into another format for warehousing and analyzing.

How to clean the data

Step 1: Remove duplicate or irrelevant observations

Remove unwanted observations from your dataset, including duplicate observations or irrelevant observations. Duplicate observations will happen most often during data collection. When you combine data sets from multiple places, scrape data, or receive data from clients or multiple departments, there are opportunities to create duplicate data. De-duplication is one of the largest areas to be considered in this process. Irrelevant observations are when you notice observations that do not fit into the specific problem you are trying to analyze. For example, if you want to analyze data regarding millennial customers, but your dataset includes older generations, you might remove those irrelevant observations. This can make analysis more efficient and minimize distraction from your primary target—as well as creating a more manageable and more performant dataset.

Step 2: Fix structural errors

Structural errors are when you measure or transfer data and notice strange naming conventions, typos, or incorrect capitalization. These inconsistencies can cause mislabeled categories or classes. For example, you may find “N/A” and “Not Applicable” both appear, but they should be analyzed as the same category.

Step 3: Filter unwanted outliers

Often, there will be one-off observations where, at a glance, they do not appear to fit within the data you are analyzing. If you have a legitimate reason to remove an outlier, like improper data-entry, doing so will help the performance of the data you are working with. However, sometimes it is the appearance of an outlier that will prove a theory you are working on. Remember: just because an outlier exists, doesn’t mean it is incorrect. This step is needed to determine the validity of that number. If an outlier proves to be irrelevant for analysis or is a mistake, consider removing it.

Step 4: Handle missing data

You can’t ignore missing data because many algorithms will not accept missing values. There are a couple of ways to deal with missing data. Neither is optimal, but both can be considered.

1. As a first option, you can drop observations that have missing values, but doing this will drop or lose information, so be mindful of this before you remove it.
2. As a second option, you can input missing values based on other observations; again, there is an opportunity to lose integrity of the data because you may be operating from assumptions and not actual observations.
3. As a third option, you might alter the way the data is used to effectively navigate null values.

Step 5: Validate and QA

At the end of the data cleaning process, you should be able to answer these questions as a part of basic validation:

- Does the data make sense?
- Does the data follow the appropriate rules for its field?
- Does it prove or disprove your working theory, or bring any insight to light?
- Can you find trends in the data to help you form your next theory?

- If not, is that because of a data quality issue?

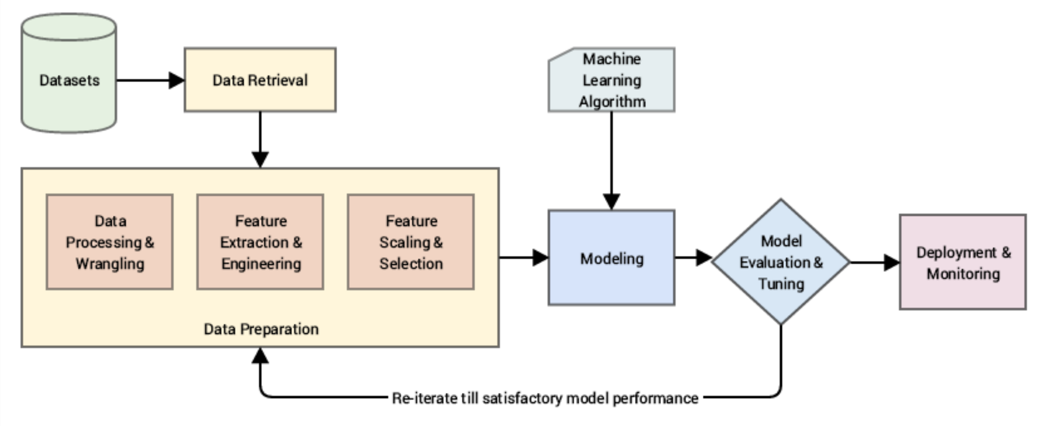
False conclusions because of incorrect or “dirty” data can inform poor business strategy and decision-making. False conclusions can lead to an embarrassing moment in a reporting meeting when you realize your data doesn’t stand up to scrutiny. Before you get there, it is important to create a culture of quality data in your organization. To do this, you should document the tools you might use to create this culture and what data quality means to you.

4.3 Feature Engineering

Feature engineering refers to the process of using domain knowledge to select and transform the most relevant variables from raw data when creating a predictive model using machine learning or statistical modeling. The goal of feature engineering and selection is to improve the performance of machine learning (ML) algorithms.

What is Feature Engineering?

The feature engineering pipeline is the preprocessing steps that transform raw data into features that can be used in machine learning algorithms, such as predictive models. Predictive models consist of an outcome variable and predictor variables, and it is during the feature engineering process that the most useful predictor variables are created and selected for the predictive model. Automated feature engineering has been available in some machine learning software since 2016. Feature engineering in ML consists of four main steps: Feature Creation, Transformations, Feature Extraction, and Feature Selection.


Datasets
Data Retrieval
Machine Learning Algorithm
Data Processing & Wrangling
Feature Extraction & Engineering
Feature Scaling & Selection
Data Preparation
Modeling
Model Evaluation & Tuning
Deployment & Monitoring
Re-iterate till satisfactory model performance

Feature engineering consists of creation, transformation, extraction, and selection of features, also known as variables, that are most conducive to creating an accurate ML algorithm. These processes entail:

Concepts in feature engineering

Feature Creation: Creating features involves identifying the variables that will be most useful in the predictive model. This is a subjective process that requires human intervention and creativity. Existing features are mixed via addition, subtraction, multiplication, and ratio to create new derived features that have greater predictive power.

Transformations: Transformation involves manipulating the predictor variables to improve model performance; e.g. ensuring the model is flexible in the variety of data it can ingest; ensuring variables are on the same scale, making the model easier to understand; improving accuracy; and avoiding computational errors by ensuring all features are within an acceptable range for the model.

Feature Extraction: Feature extraction is the automatic creation of new variables by extracting them from raw data. The purpose of this step is to automatically reduce the volume of data into a more manageable set for modeling. Some feature extraction methods include cluster analysis, text analytics, edge detection algorithms, and principal components analysis.

Feature Selection: Feature selection algorithms essentially analyze, judge, and rank various features to determine which features are irrelevant and should be removed, which features are redundant and should be removed, and which features are most useful for the model and should be prioritized.

Steps in Feature Engineering

The art of feature engineering may vary among data scientists, however steps for how to perform feature engineering for most machine learning algorithms include the following:

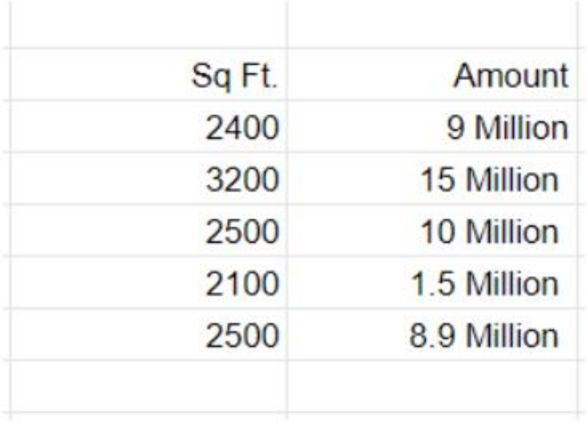
Data Preparation: This preprocessing step involves the manipulation and consolidation of raw data from different sources into a standardized format so that it can be used in a model. Data preparation may entail data augmentation, cleaning, delivery, fusion, ingestion, and/or loading.

Exploratory Analysis: This step is used to identify and summarize the main characteristics in a data set through data analysis and investigation. Data science experts use data visualizations to better understand how best to manipulate data sources, to determine which statistical techniques are most appropriate for data analysis, and for choosing the right features for a model.

Benchmark: Benchmarking is setting a baseline standard for accuracy to which all variables are compared. This is done to reduce the rate of error and improve a model’s predictability. Experimentation, testing and optimizing metrics for benchmarking is performed by data scientists with domain expertise and business users.

Examples of Feature Engineering

Now to understand it in a much easier way, let’s take a simple example. Below are the prices of properties in x city. It shows the area of the house and total price.

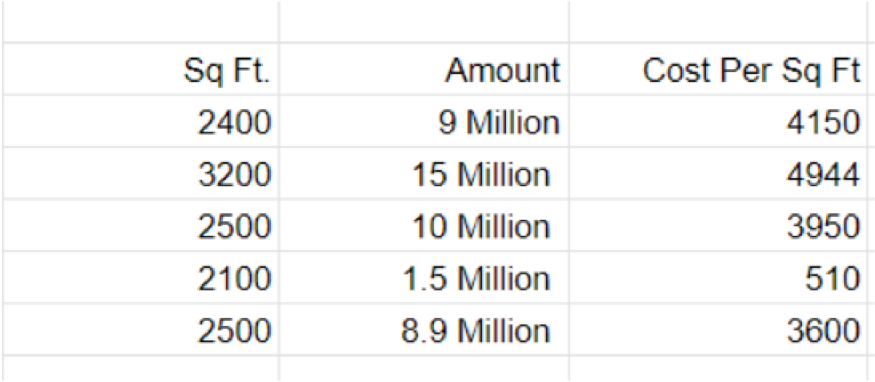


Sq Ft.	Amount
2400	9 Million
3200	15 Million
2500	10 Million
2100	1.5 Million
2500	8.9 Million

Sample Data

Sample Data

Now this data might have some errors or might be incorrect, not all sources on the internet are correct. To begin, we'll add a new column to display the cost per square foot.

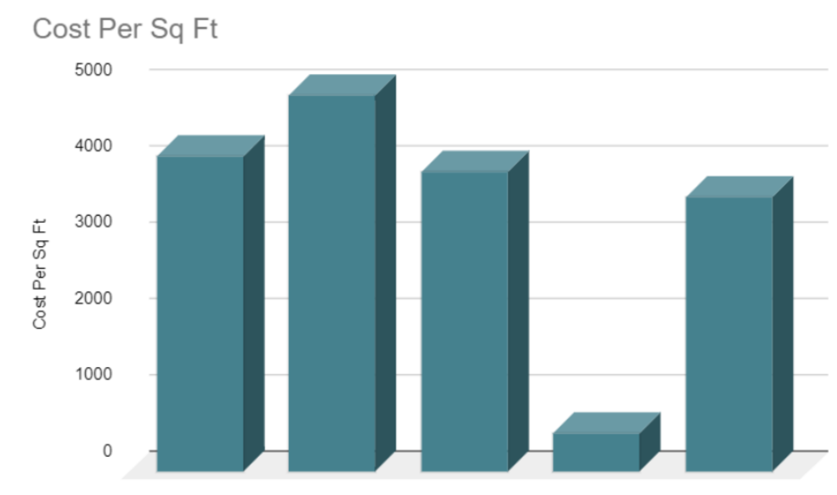


Sq Ft.	Amount	Cost Per Sq Ft
2400	9 Million	4150
3200	15 Million	4944
2500	10 Million	3950
2100	1.5 Million	510
2500	8.9 Million	3600

Sample Data

Sample Data

This new feature will help us understand a lot about our data. So, we have a new column which shows cost per square ft. There are three main ways you can find any error. You can use Domain Knowledge to contact a property advisor or real estate agent and show him the per square foot rate. If your counsel states that pricing per square foot cannot be less than 3400, you may have a problem. The data can be visualised.



When you plot the data, you’ll notice that one price is significantly different from the rest. In the visualisation method, you can readily notice the problem. The third way is to use Statistics to analyze your data and find any problem.

4.4 Missing Data

Missing data, or missing values, occur when you don’t have data stored for certain variables or participants. Data can go missing due to incomplete data entry, equipment malfunctions, lost files, and many other reasons.

In any dataset, there are usually some missing data. In quantitative research, missing values appear as blank cells in your spreadsheet.

Types of missing data

Missing data are errors because your data don’t represent the true values of what you set out to measure.

The reason for the missing data is important to consider, because it helps you determine the type of missing data and what you need to do about it.

There are three main types of missing data.

Type	Definition
Missing completely at random (MCAR)	Missing data are randomly distributed across the variable and unrelated to other variables.
Missing at random (MAR)	Missing data are not randomly distributed but they are accounted for by other observed variables.
Missing not at random (MNAR)	Missing data systematically differ from the observed values.

Missing completely at random

When data are missing completely at random (MCAR), the probability of any particular value being missing from your dataset is unrelated to anything else.

The missing values are randomly distributed, so they can come from anywhere in the whole distribution of your values. These MCAR data are also unrelated to other unobserved variables.

Example: MCAR data

You note that there are a few missing values in your holiday spending dataset. Some people started answering your survey but dropped out or skipped a question.However, you note that you have data points from a wide distribution, ranging from low to high values.Therefore, you conclude that the missing values aren't related to any specific holiday spending amount range.Data are often considered MCAR if they seem unrelated to specific values or other variables. In practice, it's hard to meet this assumption because "true randomness" is rare.When data are missing due to equipment malfunctions or lost samples, they are considered MCAR.

Missing at random

Data missing at random (MAR) are not actually missing at random; this term is a bit of a misnomer.This type of missing data systematically differs from the data you've collected, but it can be fully accounted for by other observed variables.The likelihood of a data point being missing is related to another observed variable but not to the specific value of that data point itself.

Example: MAR data

You repeat your data collection with a new group. You notice that there are more missing values for adults aged 18–25 than for other age groups.But looking at the observed data for adults aged 18–25, you notice that the values are widely spread. It's unlikely that the missing data are missing because of the specific values themselves.Instead, some younger adults may be less inclined to reveal their holiday spending amounts for unrelated reasons (e.g., more protective of their privacy).

Missing not at random

Data missing not at random (MNAR) are missing for reasons related to the values themselves.

Example: MNAR data

In the new dataset, you also notice that there are fewer low values. Some participants with low incomes avoid reporting their holiday spending amounts because they are low. This type of missing data is important to look for because you may lack data from key subgroups within your sample. Your sample may not end up being representative of your population.

How to deal with missing values

To tidy up your data, your options usually include accepting, removing, or recreating the missing data.You should consider how to deal with each case of missing data based on your assessment of why the data are missing.

Acceptance

The most conservative option involves **accepting** your missing data: you simply leave these cells blank. It's best to do this when you believe you're dealing with MCAR or MAR values. When you have a small sample, you'll want to conserve as much data as possible because any data removal can affect your statistical power. You might also recode all missing values with labels of "N/A" (short for "not applicable") to make them consistent throughout your dataset. These actions help you retain data from as many research subjects as possible with few or no changes.

Deletion

You can remove missing data from statistical analyses using listwise or pairwise deletion.

Listwise deletion

Listwise deletion means deleting data from all cases (participants) who have data missing for any variable in your dataset. You'll have a dataset that's complete for all participants included in it.

A downside of this technique is that you may end up with a much smaller and/or a biased sample to work with. If significant amounts of data are missing from some variables or measures in particular, the participants who provide those data might significantly differ from those who don't.Your sample could be biased because it doesn't adequately represent the population.

Example: Listwise deletionYou decide to remove all participants with missing data from your survey dataset. This reduces your sample from 114 to 77 participants. You notice that most of the participants with missing data left a specific question about their opinions unanswered. Many of those participants were also women, so your sample now mainly consists of men.

Pairwise deletion

Pairwise deletion lets you keep more of your data by only removing the data points that are missing from any analyses. It conserves more of your data because all available data from cases are included. It also means that you have an uneven sample size for each of your variables. But it's helpful when you have a small sample or a large proportion of missing values for some variables. When you perform analyses with multiple variables, such as a correlation, only cases (participants) with complete data for each variable are included.Example: Pairwise deletionYou decide to only remove missing values, while retaining the other data points for these participants. This does not reduce your overall sample size.

- 12 people didn't answer a question about their gender, reducing the sample size from 114 to 102 participants for the variable "gender."
- 3 people didn't answer a question about their age, reducing the sample size from 114 to 11 participants for the variable "age."

You are able to retain more values this way, but the sample size now differs across variables.

Imputation

Imputation means replacing a missing value with another value based on a reasonable estimate. You use other data to recreate the missing value for a more complete dataset. You can choose from several imputation methods. The easiest method of imputation involves replacing missing values with the mean or median value for that variable.

Hot-deck imputation

In **hot-deck imputation**, you replace each missing value with an existing value from a similar case or participant within your dataset. For each case with missing values, the missing value is replaced by a value from a so-called "donor" that's similar to that case based on data for other variables.

Example: Hot-deck imputationIn a survey, you ask participants to answer questions about how they rate a new shopping app from 1 to 5. You notice that two participants skipped Question 3, so these cells are empty. You sort the data based on other variables and search for participants who responded similarly to other questions compared to your participants with missing values. You take the answer to Question 3 from a donor and use it to fill in the blank cell for each missing value.

Cold-deck imputation

Alternatively, in **cold-deck imputation**, you replace missing values with existing values from similar cases from other datasets. The new values come from an unrelated sample.

Example: Cold-deck imputationInstead of replacing the missing values with answers from participants from the same sample, you open a different dataset from a coworker. They conducted a similar survey but used a different sample. You search for participants who responded similarly to other questions compared to your participants with missing values. You take the answer to Question 3 from the other dataset and use it to fill in the blank cell for each missing value.

Use imputation carefully

Imputation is a complicated task because you have to weigh the pros and cons.

Although you retain all of your data, this method can create research bias and lead to inaccurate results. You can never know for sure whether the replaced value accurately reflects what would have been observed or answered. That’s why it’s best to apply imputation with caution.

4.5 Understand datasets through various plots and graphs

Data visualization is the graphical representation of information and data. By using visual elements like charts, graphs, and maps, data visualization tools provide an accessible way to see and understand trends, outliers, and patterns in data.

In simple words, it is very difficult to gain knowledge from a large amount of data and this is where data visualization comes into the picture. Be it numerical or categorical or mixed type of data, visualization techniques help see the trends, outliers, or any kind of patterns in the data. All this information helps data scientists or anyone in any field of work to make better decisions to achieve their objective.

There is a well-known fact that we can grasp and retain knowledge from pictures and visuals way better than any numbers or text. With the evolution of big data in recent years visualization techniques becomes far more important than ever before because it is almost impossible for anyone to extract any information from terabytes or petabytes of data without using visualization techniques.

Most of the time data scientists deal with structured data in most of their data analysis journey and familiar with the concepts of the structured query language(SQL) tables. SQL tables represent data in terms of rows and columns and make it convenient to explore and apply transformations. Data visualization techniques are also helpful for the feature engineering part in machine learning.

4.6 Draw conclusions

As you begin to analyze the data you collected through experiments, make sure your team sets aside time to review the information with your Team Advisor and discuss how best to showcase your results and conclusions.

Within your Mission Folder, you should state whether your hypothesis was true or false, what you learned from your experiments and how your project could be improved.

Below you will find helpful questions for you and your teammates to consider as you review your data.

1. **Data Analysis:** Review data and results critically
   a. Is the data complete and accurate?
   b. Do I need to collect more data?
   c. Did I make any mistakes in my research or experimentation?
2. **Summarize Data:** What is the best way to summarize the data?
   a. Calculate an average of data collected?
   b. Summarize the results as a ratio or percentage?
   c. Display data clearly and concisely?
3. **Display Data as a Graph or Table**
   a. Place independent variable on the x-axis of a graph
   b. Place dependent variable on the y-axis of a graph
   c. Label axes
   d. Include units of measurement
   e. Show each set of data in a different color or symbol
   f. Include a legend
   g. Convert data to show all units of measurement on the same scale

Now that you have analyzed your data, the last step is to draw your conclusions. Conclusions summarize whether the experiment or survey results support or contradict the original hypothesis. Teams should include key facts from your team's background research to help explain the results.

If the results of your experiment support that your hypothesis is **TRUE**, summarize how this occurred by comparing the relationship between the independent and dependent variables.

If the results of the experiments or surveys do NOT support the hypothesis and prove the hypothesis is **FALSE,** you should not change or manipulate the results to fit the original hypothesis. Simply explain why things did not go as expected. Scientists often find that results do not support their hypothesis. They use those unexpected results as the first step in constructing a new hypothesis. If you think you need additional experimentation, you should describe what you think should happen next.

4.7 How to select ML algorithms?

Simple Steps to Choose Best Machine Learning Algorithm:

Understand Your Problem : Begin by gaining a deep understanding on the problem you are trying to solve. What is your goal? What is the problem all about classification, regression , clustering, or something else? What kind of data you are working with?

Process the Data: Ensure that your data is in the right format for your chosen algorithm. Process and prepare your data by cleaning, Clustering, Regression.

Exploration of Data: Conduct data analysis to gain insights into your data. Visualizations and statistics helps you to understand the relationships within your data.

Metrics Evaluation: Decide on the metrics that will measure the success of model. You must choose the metric that should align with your problem.

Simple models: One should begin with the simple easy-to-learn algorithms. For classification, try regression, decision tree. Simple model provides a baseline for comparison.

Use Multiple Algorithms: Try to use multiple algorithms to check that one performs on your dataset. That may include:

Decision Trees

Gradient Boosting(XGBoost, LightGBM)

Random Forest

k-Neasrest Neighbors(KNN)

Naive Bayes

Support Vector Machines(SVM)

Neural Networks(Deep Learning)

Hyperparameter Tuning: Grid Search and Random Search can helps with adjusting parameters choose algorithm that find best combination.

Cross- Validation: Use cross- validation to get assess the performance of your models. This helps prevent overfiting .

Comparing Results: Evaluate the models's performance by using the metrics evaluation. Compare their performance and choose that best one that align with problem's goal.

Consider Model Complexity: Balance complexity of model and their performance. Compare their performance and choose that one best algorithm to generalize better.

Most used Machine Learning Algorithms

Linear Regression: It is essential in searching for the relationship between two continuous variables. One is an independent variable and other is the dependent variable.

Logistic Regression: Logistic regression is one of the common methods to analyse the data and explain the relationship between one dependent binary variable and one or more independent variables of the nominal, ordinal, interval, or ratio level.

KNN: KNN can be used for classification and regression predictive problems.

K-means: K-means clustering is an unsupervised learning algorithm, which is used when we are dealing with the data which is not labelled(without proper categories or groups). The aim of the algorithm is to search the groups in the data set, with the number of groups being represented by the variable K.

Support Vector Machines(SVM): It is a supervised machine learning algorithm which can be used for classification or regression tasks. It uses a technique called the kernel trick to transform your data and then based on these transformations it finds an optimal boundary between the possible outputs.

Random Forest: It can be used for regression and classifications task. It results in greater accuracy. Random forest classifier can manage the missing values and hold the accuracy for a significant proportion of the data. If there are more number of trees, then it won't permit the trees in the machine learning model that are overfitting.

Factors to Choose Correct Algorithm

The kind of model in use (problem)

Analyzing the available Data (size of training set)

The accuracy of the model

Time taken to train the model (training time)

Number of parameters

Number of features

Linearity

Conclusion

By selecting the best machine learning algorithm for your problem is a crucial step in building effective predictive models. It involves a systematic approach that starts with understanding your problem, preprocessing your data, exploring the dataset, and selecting appropriate evaluation metrics.

Note: Extra Reference to select best machine learning algorithms. This provide good explanation for ML algorithm selection based on various factor

https://www.analyticsvidhya.com/blog/2021/07/how-to-choose-an-appropriate-ml-algorithm-data-science-projects/

Prepared by:

Prof. Neha Kudu

INFT