Data Exploration/Feature Engineering

- What is data exploration in machine learning

- How to do the data cleaning

- The need for data exploration

- What are the different data exploration techniques in machine learning, and a lot more

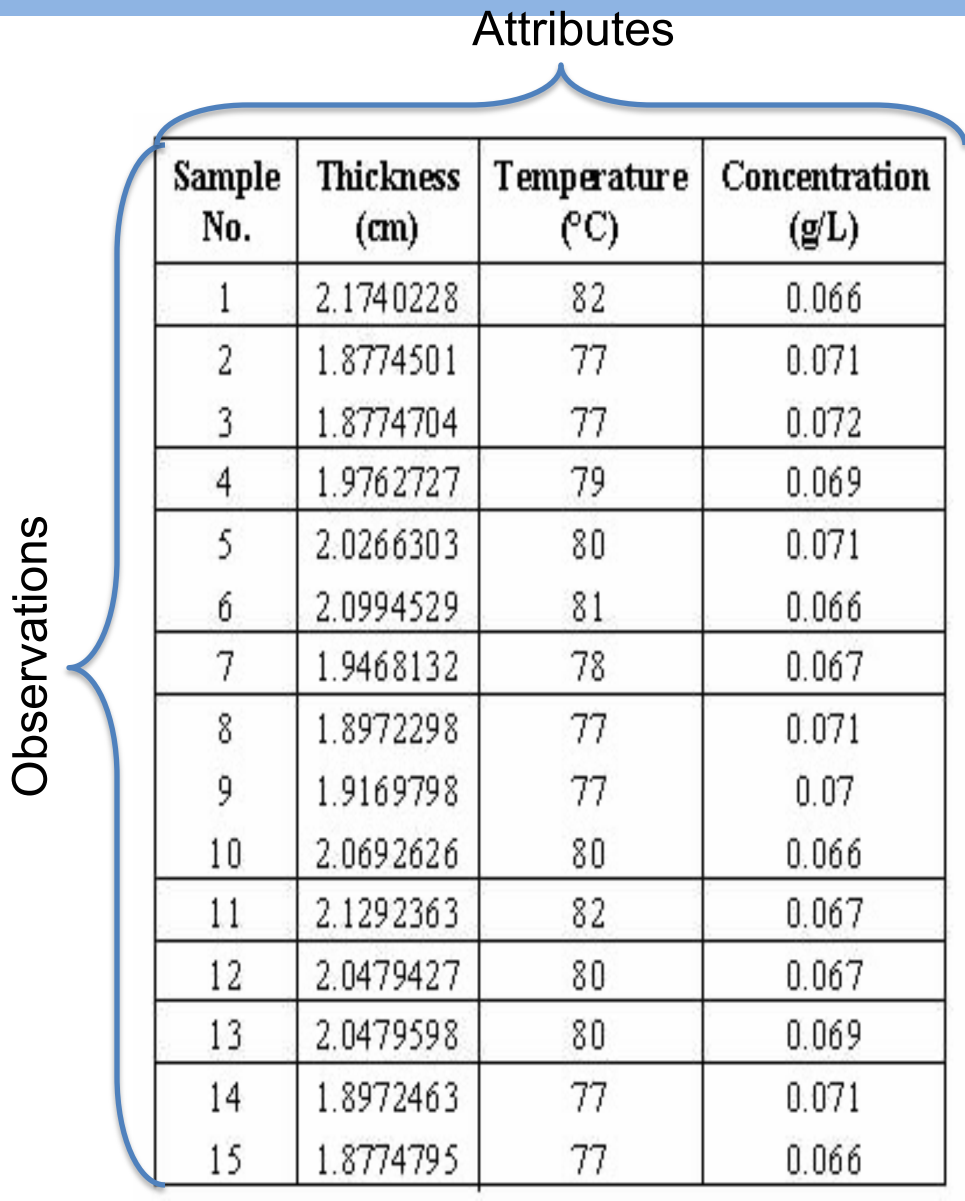
Data exploration definition: Data exploration refers to the initial step in data analysis in which data analysts use data visualization and statistical techniques to describe dataset characterizations, such as size, quantity, and accuracy, in order to better understand the nature of the data.

Data exploration techniques include both manual analysis and automated data exploration software solutions that visually explore and identify relationships between different data variables, the structure of the dataset, the presence of outliers, and the distribution of data values in order to reveal patterns and points of interest, enabling data analysts to gain greater insight into the raw data.

Data is often gathered in large, unstructured volumes from various sources and data analysts must first understand and develop a comprehensive view of the data before extracting relevant data for further analysis, such as univariate, bivariate, multivariate, and principal components analysis.

What is data?

- Data is a collection of facts, such as numbers, words, measurements, observations or even just descriptions of things
- Collection of data objects and their attributes/variable, characteristic or feature
- An attribute is a property of an object, e.g., color of a car, or temperature of furnace
- A collection of features describe an object. Object is also known as record, observation, sample etc.



Attributes

Observations

Sample No.	Thickness (cm)	Temperature (°C)	Concentration (g/L)
1	2.1740228	82	0.066
2	1.8774501	77	0.071
3	1.8774704	77	0.072
4	1.9762727	79	0.069
5	2.0266303	80	0.071
6	2.0994529	81	0.066
7	1.9468132	78	0.067
8	1.8972298	77	0.071
9	1.9169798	77	0.07
10	2.0692626	80	0.066
11	2.1292363	82	0.067
12	2.0479427	80	0.067
13	2.0479598	80	0.069
14	1.8972463	77	0.071
15	1.8774795	77	0.066

Data Representation

- Data has form: $\{(x_1,y_1),\ldots,(x_n,y_n)\}$ (labeled), or $\{x_1,\ldots,x_n\}$ (unlabeled)
- What the label y looks like is task-specific
- What about x which denotes a real-world object (e.g., image or text document)?
- Each example x is a set of (numeric) features/attributes/dimensions
- Features encode properties of the object which x represents
- x is commonly represented as a $D \times 1$ vector
- Representing a 28×28 image: x can be a 784×1 vector of pixel values
- Representing a text document: x can be a vector of word-counts of words appearing in that document

Cont’d

Attirbutes						
Observations		x_1	x_2	x_3	…	x_p
	1	x_{11}	x_{12}	x_{13}	…	x_{1p}
	2	x_{21}	x_{22}	x_{23}	…	x_{2p}
	3	x_{31}	x_{23}	x_{33}	…	x_{3p}
	…	…	…	…	…	…
	n	x_{n1}	x_{n2}	x_{n3}		x_{np}

- You got your data: what’s next:

What kind of analysis do you need which model is more appropriate for it? …

What is feature and why we need engineering of it

All machine learning algorithms use some input data to create outputs. This input data comprise features, which are usually in the form of structured columns. Algorithms require features with some specific characteristic to work properly.

Here, the need for feature engineering arises.

Preprocessing

- Real-world databases are highly susceptible to noisy, missing and inconsistent data due their typically huge size (often several gigabytes or more) and their likely origin from multiple, heterogeneous sources.

- The data that we work with is raw, it is not clean and needs processing to be ready to be passed to a machine learning model or get some useful insights.

- Low quality data will lead to a low-quality mining results.

- A Machine Learning project is as good as the foundation of data on which it is built. In order to perform well, machine learning data exploration models must ingest large quantities of data, and model accuracy will suffer if that data is not thoroughly explored first.

- *How can the data be preprocessed in order to help improve the quality of the data and, consequently, of the mining results? Moreover, how can the data be preprocessed so as to improve the efficiency and ease of mining process*

Most statistical method focuses on data modeling, prediction and statistical inference while it is usually assumed that data are in the correct state for data analysis. In practice, a data analyst/modeler spends much if not most of his time on preparing the data before doing any statistical operation.

It is very rare that the raw data one works with are in the correct format, are without errors, are complete and have all the correct labels and codes that are needed for analysis.

Data Preprocessing/Feature Engineering

Data preparation is one of the most difficult and important steps in any machine learning project.

The reason is that each dataset is different and highly specific to the project. Nevertheless, there are enough commonalities across predictive modeling projects that we can define a loose sequence of steps and subtasks that you are likely to perform.

There are number of preprocessing techniques which can be applied to improve the quality of the data.

Data Preprocessing/Feature Engineering

Each machine learning project is different because the specific data at the core of the project is different.

The right features can only be defined in the context of both the model and the data; since data and models are so diverse, it is difficult to generalize the practice of feature engineering across projects.

Even though project is unique, the steps on the path to a good or even the best result are generally the same from project to project. This is known as “data science process”.

The process consists of a sequence of steps. The steps are the same, but the names of the steps and takes performed may differ from description to description.

Need for data Preprocessing

The need for data preprocessing is there because good data is undoubtedly more important than good models and for which the quality of the data is of paramount importance. Therefore, companies and individuals invest a lot of their time in cleaning and preparing the data for modeling.

•Data preprocessing is very important and comprises majority of the work in a data mining/modeling process (about 80% of the work is done in this stage)

•You may have heard that 80% of a data scientist's time goes into data preprocessing and 20% of the time for model building. This isn't false and is actually the case.

The data present in the real world contains a lot of quality issues, noise, inaccurate, and not complete. It may not contain relevant, specific

Need for Data Preprocessing?

To improve the quality of the data preprocessing is essential.

The preprocessing helps to make the data consistent by eliminating any duplicates, irregularities in the data, normalizing the data to compare, and improving the accuracy of the results.

The machines understand the language of numbers, primarily binary numbers 1s and 0s.

Need For Data Pre-Processing

- You want to get the best accuracy from machine learning algorithms on your datasets.

- Some machine learning algorithms require the data to be in a specific form. Whereas other algorithms can perform better if the data is prepared in a specific way, but not always.

- It is important to prepare your data in such a way that it gives various different machine learning algorithms the best chance on your problem.

- You need to pre-process your raw data as part of your machine learning project.

- Some data preparation is needed for all mining tools. The purpose of preparation is to transform data sets so that their information content is best exposed to the mining tool

- Preparing data also prepares the scientist so that when using prepared data the scientist produces better models, and faster.

- Good data is a prerequisite for producing effective models of any type.

- Several data mining methods are sensitive to the scale and/or type of the variables

- Different variables (columns of our data sets) may have rather different scales

- Some methods are not able to handle either nominal or numeric Variables

- We may need to “create” new variables to achieve our objectives

- Sometimes we are more interested in relative values (variations) than absolute values

- We may be aware of some domain-specific mathematical relationship among two or more variables that is important for the task

- Frequently we have data sets with unknown variable values

- Our data set may be too large for some methods to be applicable

- If the user believe that the data are dirty, they are unlikely to trust the results of any data mining that has been applied to it. Furthermore, dirty data can cause confusion for the mining procedure, resulting in unreliable output.

Data Preprocessing

There may also be interplay between the data preparation step and the evaluation of models.

Information known about the choice of algorithms and the discovery of well-performing algorithms can also inform the selection and configuration of data preparation methods.

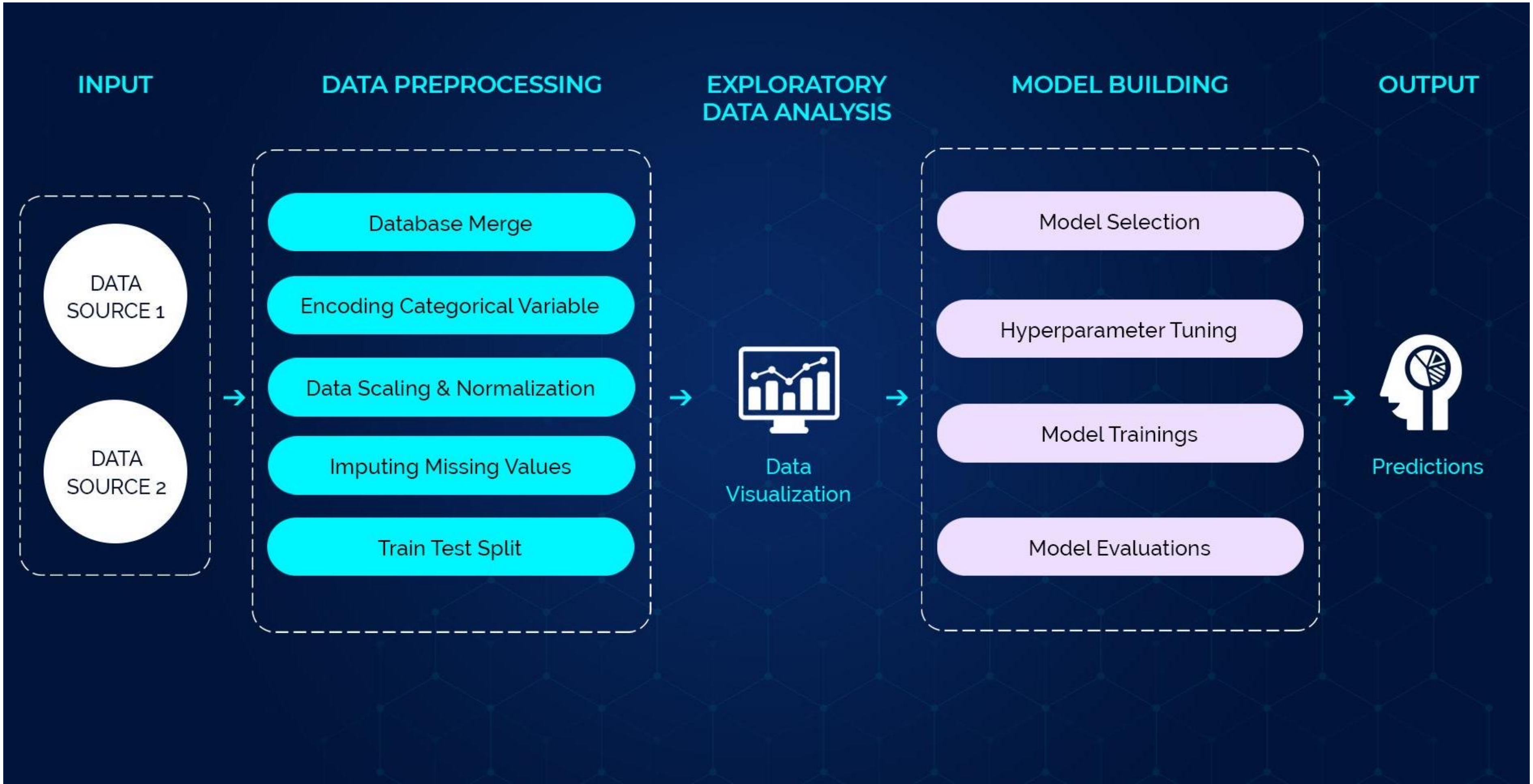
For example, the choice of algorithms may impose requirements and expectations on the type and form of input variables in the data. This might require variables to have a specific probability distribution, the removal of correlated input variables, and/or the removal of variables that are not strongly related to the target variable.

What is Data Preprocessing?

Data preprocessing in machine learning is the process of preparing the raw data to make it ready for model making. It is the first and the most crucial step in any machine learning model process.

What is Data Preprocessing?

Post the collection and combining the different data sources, data preprocessing in machine learning comes first in its pipeline.



Data cleaning, or data preparation is an essential part of statistical analysis. In fact, in practice it is often more time-consuming than the statistical analysis itself. These techniques cover technical as well as subject-matter aspects of data cleaning.

Technical aspects include data reading, type conversion and string matching and manipulation. Subject-matter related aspects include topics like data checking, error localization and an introduction to imputation methods.

If there is much irrelevant and redundant information present or noisy and unreliable data, then knowledge during the training phase is more difficult. Data preparation can take considerable amount of time. The product of the data pre-processing is the final training data set.

The Python provides a good environment for reproducible data cleaning since all cleaning actions can be scripted and therefore reproduced.

Raw data is highly susceptible to noise, missing values, and inconsistency. The quality of data affects the data mining results. In order to improve the quality of the data and, consequently of the mining results raw data is pre-processed so as to improve the efficiency and ease of the mining process. Data preparation is one of the most steps in data science project which deals with the preparation and transformation of raw dataset.

Data Preparation

- Preparing the data is one of the most time-consuming parts of data analysis.
- The way in which the data is collected and prepared is critical to the confidence with which decisions can be made.
- The data needs to be merged into a table and this may involve integration of the data from multiple sources.
- Once the data is in a tabular format, it should be fully characterized.
- The data should be cleaned by resolving ambiguities and errors, removing redundant and problematic data, and eliminating columns of data irrelevant to the analysis.
- This whole process is called *pre-processing. It goes by other names, such as "data wrangling", "data cleaning", and feature engineering*
- Data pre-processing techniques generally refer to the addition, deletion, or transformation of the dataset.

Measure of Data Quality

- Raw data is often not useful without some kind of organization or manipulation. Raw data seems to be just a bunch of meaningless values without any context or some level of organization.
- In recent years, data quality has gained more an more attention due to extended use of data warehouse systems and a higher relevance of customer relationship management.
- Due to this fact for decision makers, the benefits of data depends heavily on their completeness, correctness, and timeliness, respectively. Such properties are known as data quality dimensions.
- The consequences of poor data quality are manifold: They range from worsening customer relationships and customer satisfaction by falsely addressing customers to insufficient decision support for managers.

Measure of Data Quality

- The following measure can be used to test the quality of data
  - *Completeness: The proportion of stored data against the potential of "100% complete"*
  - *Uniqueness: No observation will be recorded more than once based upon how that observation is identified.*
  - *Velocity: The rate at which data is coming especially for streaming data*
  - *Accuracy: The degree to which data correctly describes the "real world" event being described*
  - *Consistency: The absence of difference, when comparing two or more representations of a thing against a definition.*
  - *Accessibility: How easy it is to access the data.*

Data Preprocessing/Feature Engineering

How do we know what data preparation techniques to use in our data?

As with many questions of statistics, the answer to “which feature engineering methods are the best?” is that it depends. Specifically, it depends on the model being used and the true relationship with the outcome.

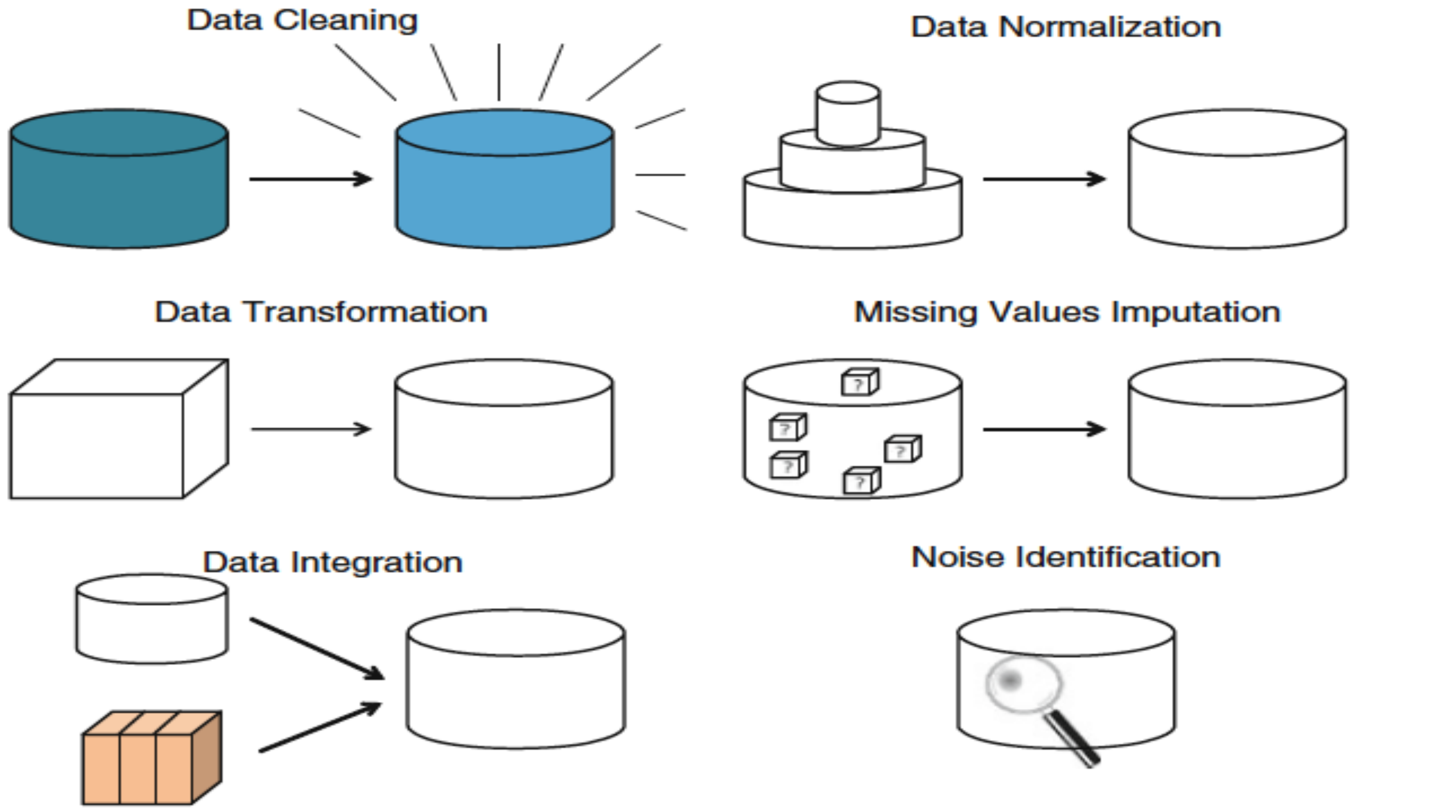
On the surface, this is a challenging question, but if we loo at the data preparation step in the context of the whole project, it becomes more straightforward.

The step before data preparation involves defining the problem

Major tasks involved

- **Integration of data**
  - Integration of data from multiple databases, or files
- **Data Cleaning**
  - Fill in missing values, smooth noisy data, remove outliers and resolve inconsistencies
- **Feature Selection**
  - Identifying features that are most relevant to the task
- **Feature Engineering**
  - Deriving new features from available data
- **Data Transformation**
  - Scaling/normalization and aggregation. Normalization may improve the accuracy and efficiency of mining algorithms involving distance measurements.
- **Data reduction**
  - Optimize the features/attributes and obtain reduced representation in volume.

- Forms of data preparation



Cont’d

- These techniques are not mutually exclusive; they may work together. For example, data cleaning can involve transformations to correct wrong data, such as by transforming all entries for a date field to a common format.

- Each of these tasks is a whole field of study with specialized algorithms.

- Data preparation is not performed blindly

- Data cleaning techniques, when applied before mining, can substantially improve the overall quality of the modeling.

- Data cleaning routines work to “clean” the data by filling in missing values, smoothing noisy data, identifying o removing outliers, and resolving inconsistencies.

How to Prepare Data

Below are the steps involved to understand, clean and prepare the data for building the predictive model:

1. Variable Identification
2. Univariate Analysis
3. Bi-variate Analysis
4. Missing values treatment
5. Outlier treatment
6. Variable transformation
7. Variable creation

Finally, we will need to iterate over steps 4 – 7 multiple times before we come up with our refined model.

What are the ways to manipulate data

- Missing values
- Data Summarization
- Group by factors
- Aggregate
- Subset/Exclude
- Bucketing Values
- Rearrange (Shape)
- Merge Datasets

Cont’d

The steps in data preprocessing in machine learning are:

1. Consolidation after acquisition of the data
2. Data Cleaning:

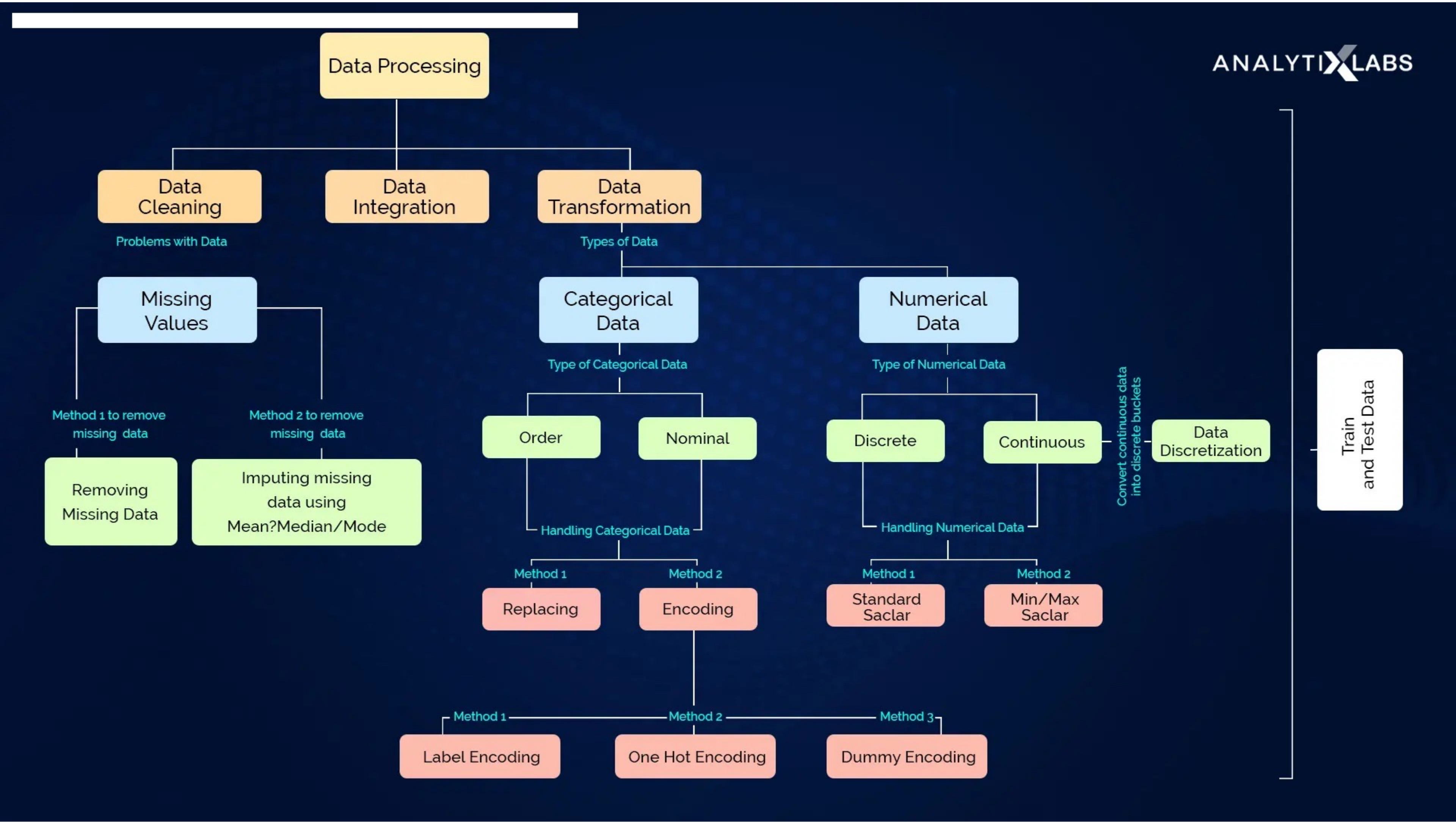
   Convert the data types if any mismatch present in the data types of the variables

   Change the format of the date variable to the required format

   Replace the special characters and constants with the appropriate values
3. Detection and treatment of missing values
4. Treating for negative values, if any present depending on the data
5. Outliers detection and treatment
6. Transformation of variables
7. Creation of new derived variables
8. Scale the numerical variables
9. Encode the categorical variables
10. Split the data into training, validation, and test set

Data Preprocessing Techniques

The data preprocessing techniques in machine learning can be broadly



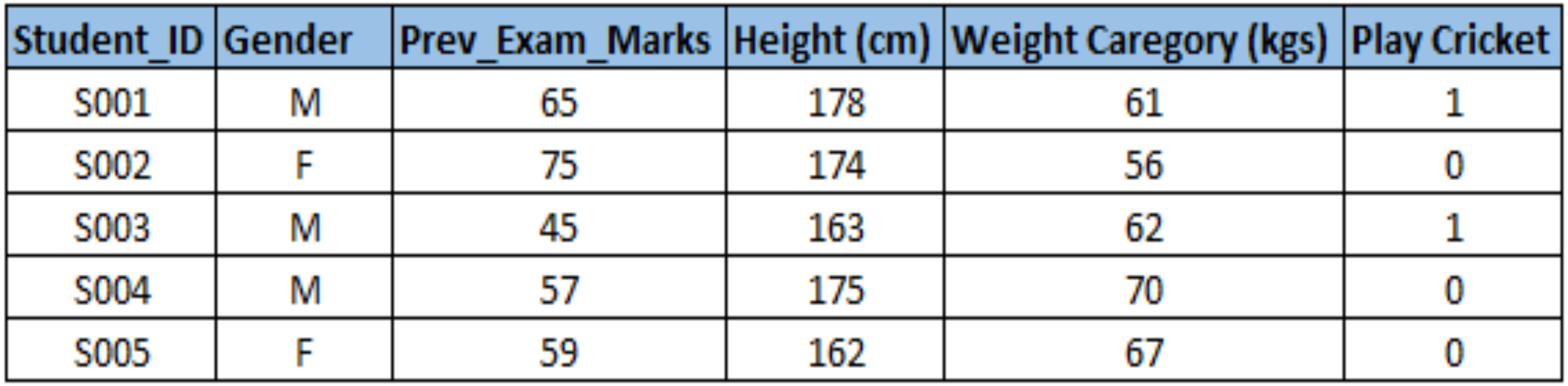
Data Integration

- Data integration is the process of combining data from multiple sources. These sources may include multiple databases, data cubes or flat files
- The data may also need to be transformed into forms appropriate for mining.
- Integrate metadata from different sources
- Removing duplicates and redundant data
- Detect and resolve data value conflicts
  - For the same real world entity, attribute values from different sources are different, e.g., different scales, different units (miles vs. km)

Variable Identification

- First, identify Predictor (Input) and Target (output) variables. Next, identify the data type and category of the variables. Let’s understand this step more clearly by taking an example.

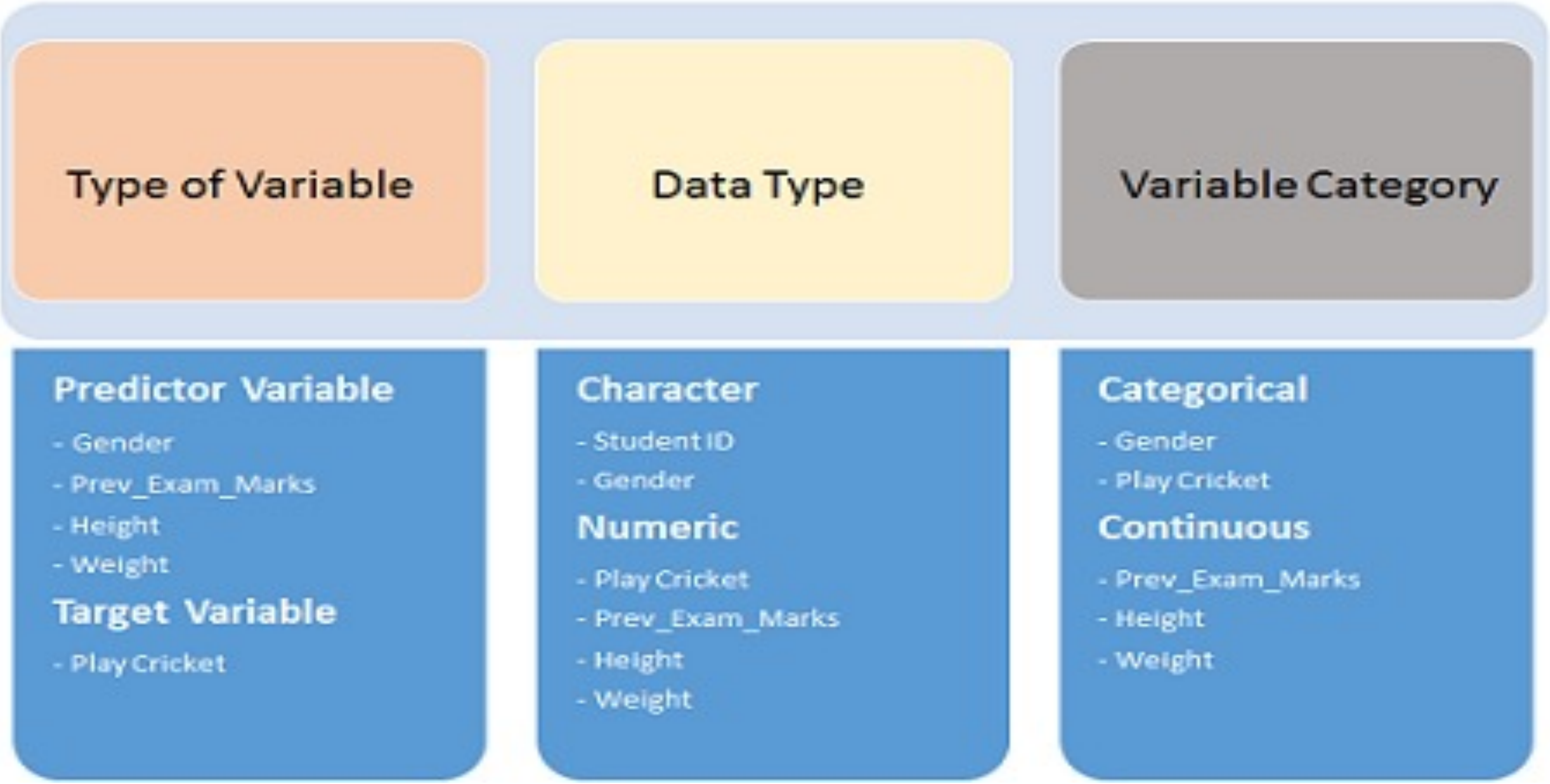
Example:- Suppose, we want to predict, whether the students will play cricket or not (refer below data set). Here you need to identify predictor variables, target variable, data type of variables and category of variables



Student_ID	Gender	Prev_Exam_Marks	Height (cm)	Weight Caregory (kgs)	Play Cricket
S001	M	65	178	61	1
S002	F	75	174	56	0
S003	M	45	163	62	1
S004	M	57	175	70	0
S005	F	59	162	67	0

Variable Identification

The variables have been defined in different category:

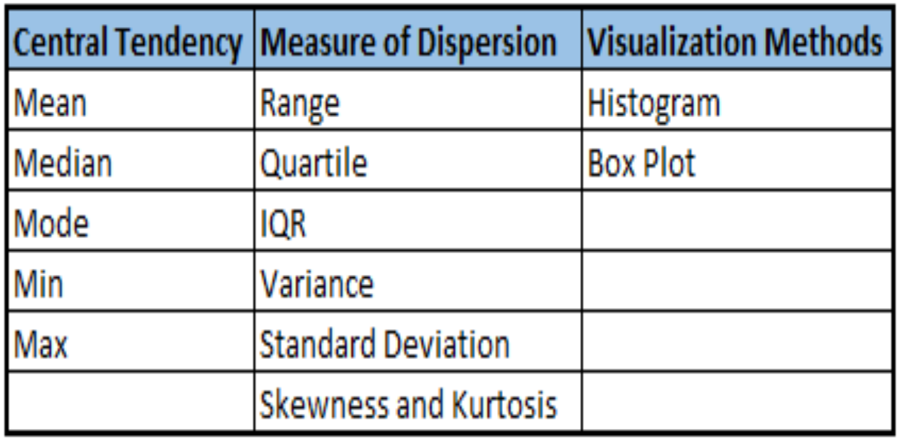


Type of Variable	Data Type	Variable Category
Predictor Variable - Gender - Prev_Exam_Marks - Height - Weight **Target Variable** - Play Cricket	**Character** - Student ID - Gender **Numeric** - Play Cricket - Prev_Exam_Marks - Height - Weight	**Categorical** - Gender - Play Cricket **Continuous** - Prev_Exam_Marks - Height - Weight

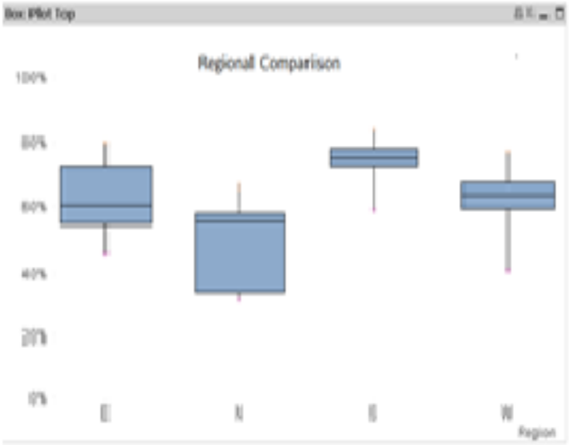
Univariate Analysis

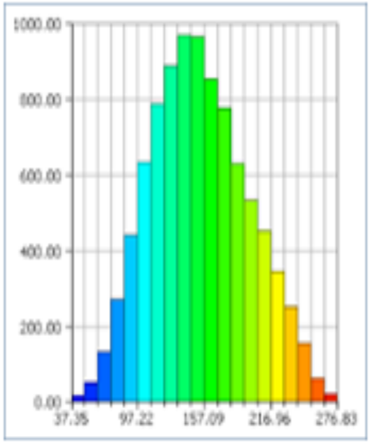
In univariate analysis, we explore variable one by one. Analysis method will depend on whether the variable is categorical or continuous.

Continuous Variables: For continuous variables, we need to understand the central tendency and spread of the variable. These are measured using various statistical metrics and visualization methods. Analysis can be used to highlight missing and outlier values.



Central Tendency	Measure of Dispersion	Visualization Methods
Mean	Range	Histogram
Median	Quartile	Box Plot
Mode	IQR	
Min	Variance	
Max	Standard Deviation	
	Skewness and Kurtosis	





Univariate Analysis

Categorical Variables: For categorical variables, we'll use frequency table to understand distribution of each category. We can also read as percentage of values under each category. Bar chart can be used visually to see the distribution.

Bivariate Analysis

Bivariate analysis finds out the relationship between two variables. Association and disassociation between variables is analyzed at a pre-defined significance level. One of the most fundamental questions in statistical learning is the relationship between variables. Estimating a measure of association between two variables can help make the right decisions in everyday data science-related problems for different reasons:

- Reduce model complexity — by implementing correlation-based feature selection.
- Reduce multicollinearity — by manipulating highly correlated variables.
- Reduce model uncertainty and infer insights.

Bivariate Analysis

Measuring correlation is tricky since there are many correlation coefficients. Furthermore, each coefficient suits different setups and assumptions.

Any combination of categorical and continuous variables can be analyzed. The combination can be :

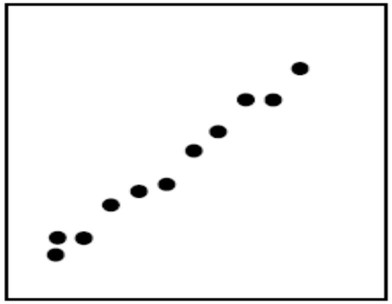
Continuous & Continuous

Continuous & Categorical

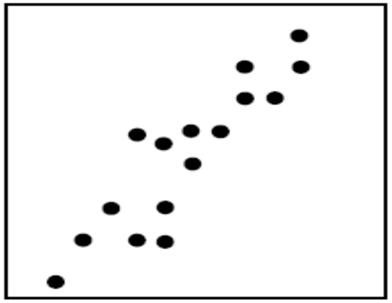
Categorical and Categorical

Different methods are used to tackle these combinations during analysis process.

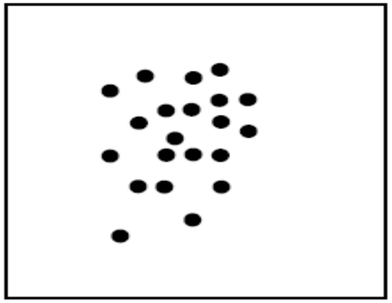
Continuous & Continuous: Scatter Plots should be looked at. The pattern of the scatter plot indicates the relationship between variables. The relationship can be linear of non-linear



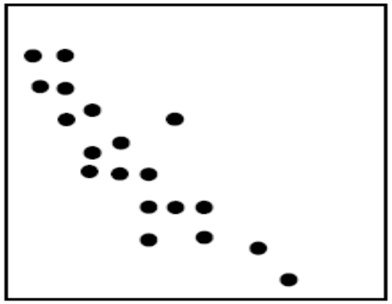
Strong positive correlation



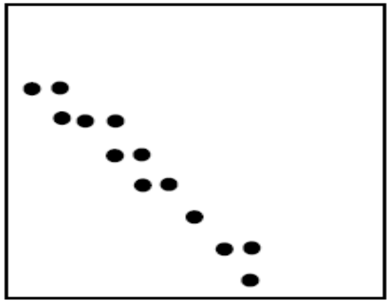
Moderate positive correlation



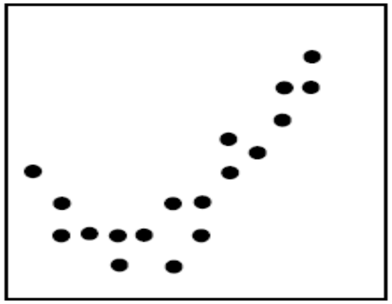
No correlation



Moderate negative correlation



Strong negative correlation



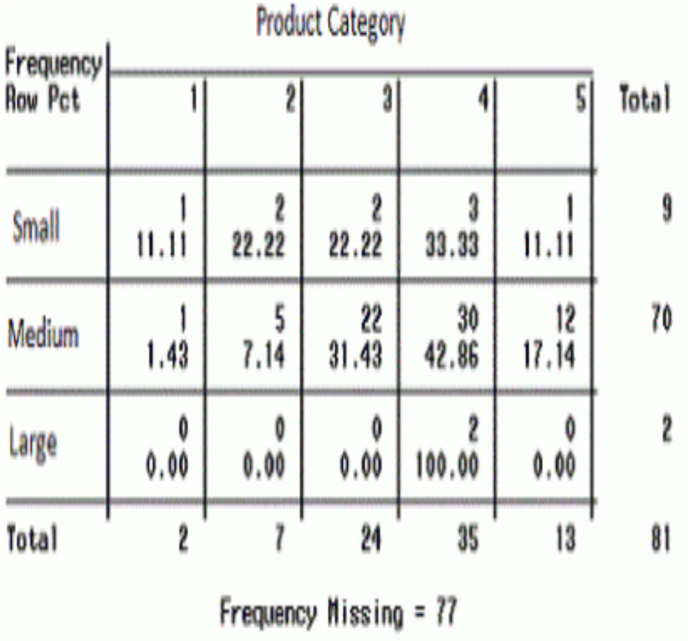
Curvilinear relationship

- Scatter plot shows the relationship between two variable but does not indicate the strength of relationship amongst them. To find the strength of the relationship, we use Correlation. Correlation varies between -1 and +1.
- -1: perfect negative linear correlation
- +1:perfect positive linear correlation and
- 0: No correlation

Categorical & Categorical:

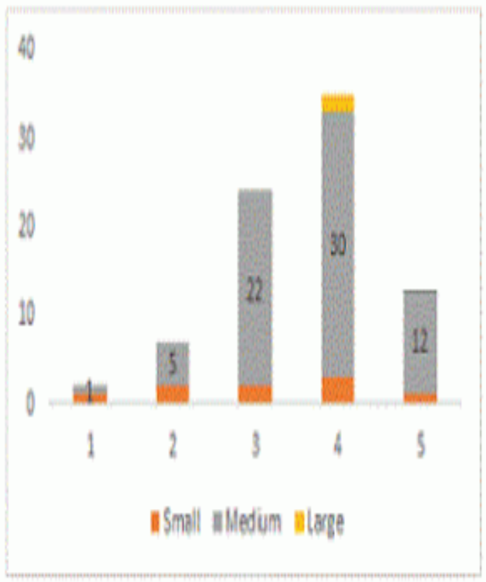
Two-way table: Two-way table of frequency and relative frequency.

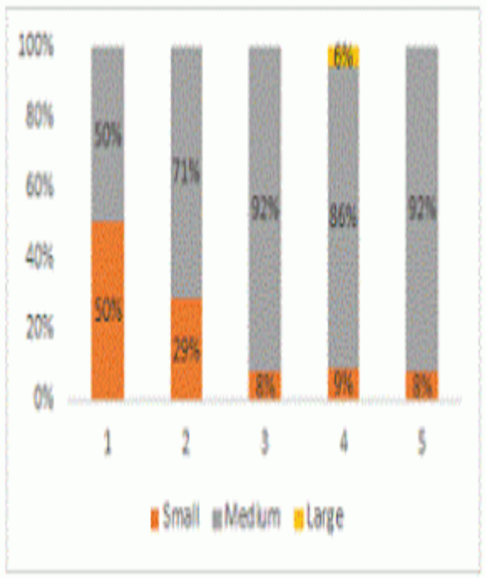
Stacked Column Chart: This method is more of a visual form of the two-way table



	Product Category					
Frequency Row Pct	1	2	3	4	5	Total
Small	1 11.11	2 22.22	2 22.22	3 33.33	1 11.11	9
Medium	1 1.43	5 7.14	22 31.43	30 42.86	12 17.14	70
Large	0 0.00	0 0.00	0 0.00	2 100.00	0 0.00	2
Total	2	7	24	35	13	81

Frequency Missing = 77



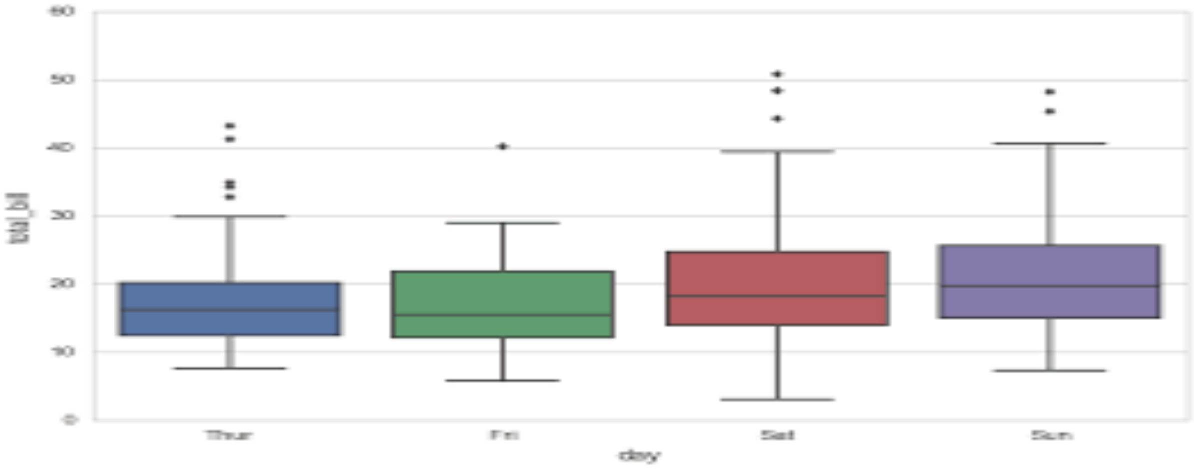


- **Chi-Square Test:** This test is used to derive the statistical significance of relationship between the variables. Also, it tests whether the evidence in the sample is strong enough to generalize that the relationship for a larger population as well. Chi-square is based on the difference between the expected and observed frequencies in one or more categories in the two-way table. It returns probability for the computed chi-square distribution with the degree of freedom.
- Probability of 0: It indicates that both categorical variable are dependent
- Probability of 1: It shows that both variables are independent.
- Probability less than 0.05: It indicates that the relationship between the variables is significant at 95% confidence.

Categorical & Continuous:

Z-test/T-test: Either test assess whether mean of two groups are statistically different from each other or not. The T-test is very similar to Z-test but it is used when number of observation for both categories is less than 30.

Box plot: This method is more of a visual form for each level of categorical variables



Data Cleaning

Data cleaning refers to identifying and correcting errors in the dataset that may negatively impact a predictive model.

Data cleaning is used to refer to all kinds of tasks and activities to detect and repair errors in the data.

Although critically important, data cleaning is not exciting, not does it involve fancy techniques. Just a good knowledge of the dataset.

Cleaning up your data is not the most glamorous of tasks, but it’s an essential part of data wrangling. Knowing how to properly clean and assemble your data will set you miles apart from others in your field.

Data Cleaning

Data cleaning involves fixing systematic problems or errors in "messy" data that may negatively impact a predictive model.

The most useful data cleaning involves deep domain expertise and could involve identifying and addressing specific observations that may be incorrect.

There are many reasons data may have incorrect values, such as being mistyped, corrupted, duplicated, and so on. Domain expertise may allow obviously erroneous observations to be identified as they are different from what is expected, such as a person's height of 200 feet.

Once messy, noisy, corrupt, or erroneous observations are identified, they can be addressed. This might involve removing a row or a column. Alternately, it might involve replacing observations with new values.

Data Cleaning

Nevertheless, there are general data cleaning operations that can be performed, such as:

- Using statistics to define normal data and identify outliers.
- Identifying columns that have the same value or no variance and removing them.
- Identifying duplicate rows of data and removing them.
- Marking empty values as missing. Imputing missing values using statistics or a learned model.

Data cleaning is an operation that is typically performed first, prior to other data preparation operations.

Data Cleaning (consistence)

Making the data consistent across the values, which can mean:

- The attributes may have incorrect data types and are not in sync with the data dictionary. Correction of the data types is a must before proceeding with any type of data cleaning.

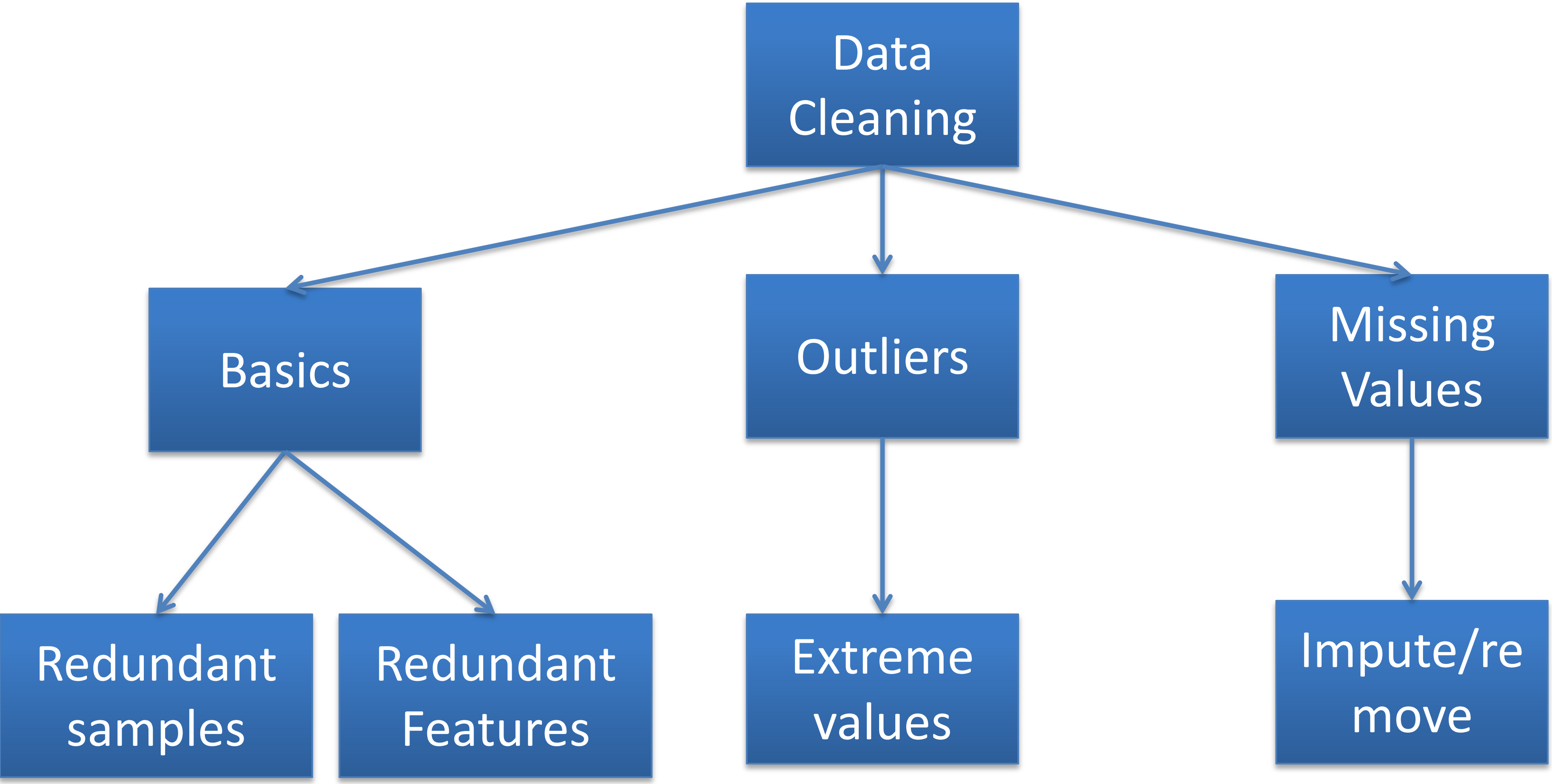
- Replace the special characters for example: replace $ and comma signs in the column of Sales/Income/Profit i.e making $10,000 as 10000.

- Making the format of the date column consistent with the format of the tool used for data analysis.

Data Cleaning (consistence)

Please note the above steps are not comprehensive. The data cleaning steps vary and depend on the nature of the data. For instance, text data consisting of, say, reviews, or tweets would have to be cleaned to make the cases of the words the same, remove punctuation marks, any special characters, remove common words, and differentiate words based on the parts of speech.

Data Cleaning



Data Cleaning

Basics

Outliers

Missing Values

Redundant samples

Redundant Features

Extreme values

Impute/re move

Data Cleaning

Data cleaning is a critically important step in any machine learning project.

In tabular data, there are many different statistical analysis and data visualization techniques you can use to explore your data in order to identify data cleaning operations you may want to perform.

Before jumping to the sophisticated methods, there are some very basic data cleaning operations that you probably should perform on every single machine learning project. These are so basic that they are often overlooked by seasoned machine learning practitioners, yet are so critical that if skipped, models may break or report overly optimistic performance results.

Data Cleaning

There are many types of errors that exist in a dataset, although some of the simplest errors include columns that don’t contain much information and duplicated rows.

- **Identify columns that contain a single value**
    - Columns that have a single observation or value are probably useless for modeling
    - These columns are referred to zero-variance features because there truly is no variance displayed by the feature.
    - Columns that have a single value do not contain any information for modeling

You can detect this by using ***unique()*** function and can be removed by using ***drop()*** function.

Data Cleaning

- **Identify columns that contain very few values**
  - There may be columns in the dataset which have few unique values. This might make sense for ordinal or categorical features. In this case, the dataset only contains numerical variables.
  - These columns are near-zero variance features, as their variance is not zero, but a very small number close to zero.

These columns may or may not contribute to the model. We can’t assume that they are useless to modeling. This does not mean that these columns should be deleted, but they require further attention. For example:

- The unique values can be encoded as ordinal values
- Unique values can be encoded as categorical values
- Compare model with each variable removed from the dataset.

Identify rows that contain Duplicate Data

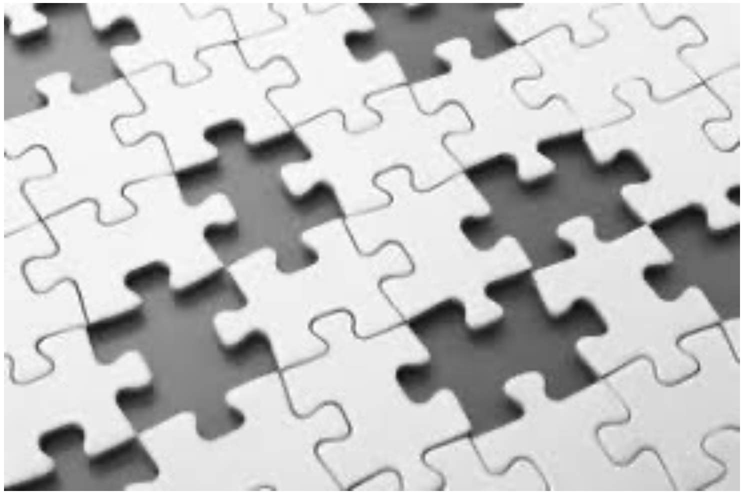
Rows that have identical data are probably useless

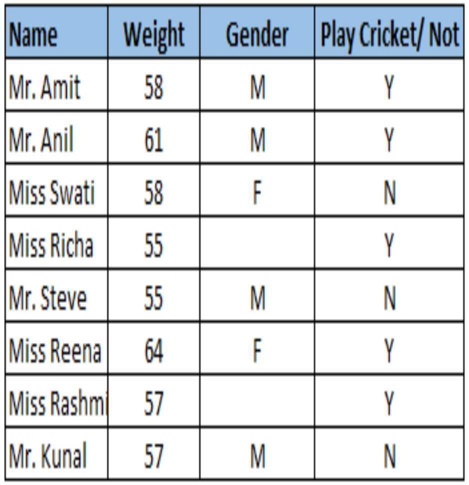
If you have used raw data that may have duplicate entries, removing duplicate data will be an important step in ensuring that data can be used accurately.

The pandas function ***duplicated()*** can be used to find the duplicated values and ***drop_duplicates()*** function can be used to remove duplicate rows.

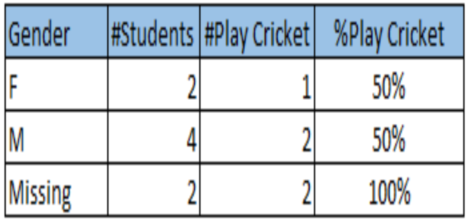
Missing Data

- Missing data in the data set can reduce the power/fit of a model or can lead to a biased model. It can lead to wrong prediction or classification.
- Data is not always available
  - e.g., many records have not values for attribute, such as customer age or income in sales data

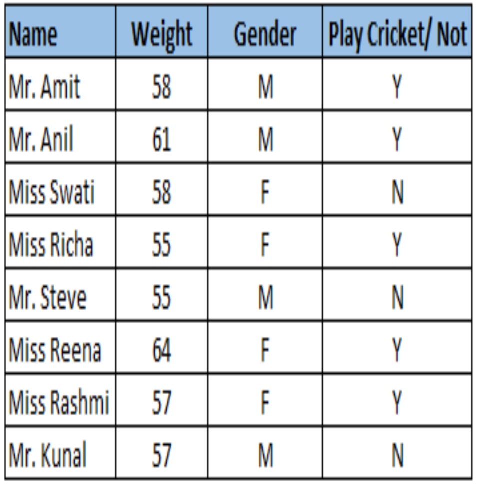




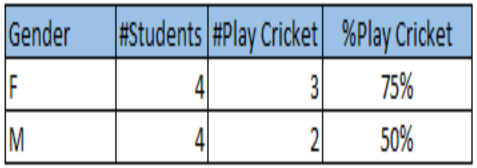
Name	Weight	Gender	Play Cricket/ Not
Mr. Amit	58	M	Y
Mr. Anil	61	M	Y
Miss Swati	58	F	N
Miss Richa	55		Y
Mr. Steve	55	M	N
Miss Reena	64	F	Y
Miss Rashmi	57		Y
Mr. Kunal	57	M	N



Gender	#Students	#Play Cricket	%Play Cricket
F	2	1	50%
M	4	2	50%
Missing	2	2	100%



Name	Weight	Gender	Play Cricket/ Not
Mr. Amit	58	M	Y
Mr. Anil	61	M	Y
Miss Swati	58	F	N
Miss Richa	55	F	Y
Mr. Steve	55	M	N
Miss Reena	64	F	Y
Miss Rashmi	57	F	Y
Mr. Kunal	57	M	N



Gender	#Students	#Play Cricket	%Play Cricket
F	4	3	75%
M	4	2	50%

Missing Data

- Missing data may be due to the following reasons:
  - Equipment malfunction
  - Data not entered properly
  - Certain data may not be considered important at the time of data entry or collection
  - Deleted due to inconsistent
  - Information is not collected (e.g. people decline to give their age)
  - NA, Inf, NaN, NULL
  - Attributes may not be applicable to all cases
    (e.g., annual income is not applicable to children)

Handling missing values

- *Ignore the Missing Value During Analysis*: This is usually done when class label is missing (assuming the mining task involves classification). This method is not very effective, unless the record contains several attributes with missing values. It is especially poor when the percentage of missing values per attribute varies considerably.
- *Fill in the missing value manually:* In general, this approach is time-
- consuming and may not be feasible given a large data set with many missing values
- Use a global constant/mean or median to fill in the missing value: R Imputation is a method to fill in the missing values with the estimated one. The objective is to employ known relationships that can be identified in the valid values of the data set to assist in estimating the missing values. Mean/Mode/Median imputation is one of the most frequently used methods. It consists of replacing the missing data for a given feature by the mean or median (quantitative feature) or mode (qualitative feature) of all known values of that variable. by the same constant/mean or median of the attribute based on the type of the data that is missing:

Handling missing values

Generalized Imputation: In this case, we calculate the mean or median for all non missing values of that variable then replace missing value with mean or median.

Similar case Imputation: In this case, we calculate average for each category and replace accordingly. For example, “**Male”** (29.75) and “**Female**” (25) individually of non missing values then replace the missing value based on gender. For “**Male**“, we will replace missing values of manpower with 29.75 and for “**Female**” with 25.

Handling missing values

Prediction Model: Prediction model is one of the sophisticated method for handling missing data. Here, we create a predictive model to estimate values that will substitute the missing data. In this case, we divide our data set into two sets: One set with no missing values for the variable and another one with missing values. First data set become training data set of the model while second data set with missing values is test data set and variable with missing values is treated as target variable. Next, we create a model to predict target variable based on other attributes of the training data set and populate missing values of test data set. We can use regression, ANOVA, Logistic regression and various modeling technique to perform this.

Handling missing values

There are 2 drawbacks for this approach:

The model estimated values are usually more well-behaved than the true values

If there are no relationships with attributes in the data set and the attribute with missing values, then the model will not be precise for estimating missing values.

- Note that, it is as important to avoid adding bias and distortion to the data as it is to make the information available.

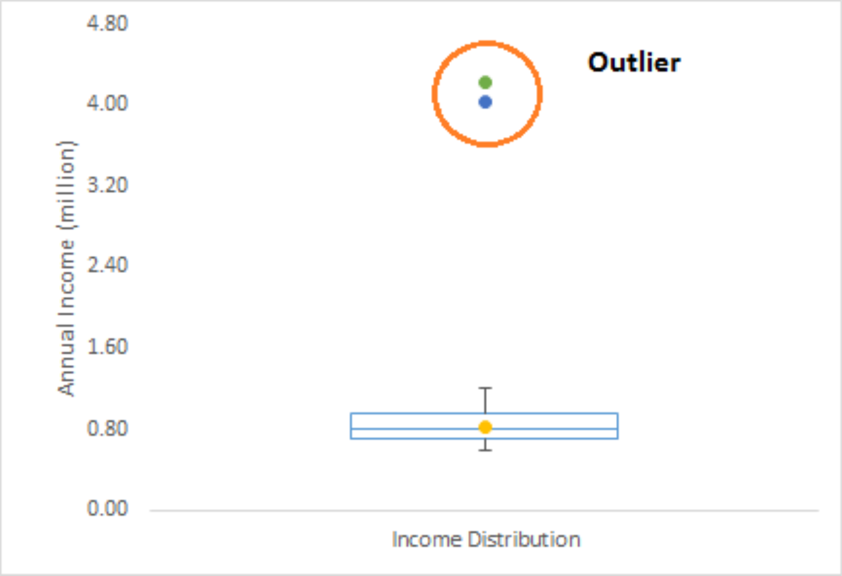
  - bias is added when a wrong value is filled-in

- No matter what techniques you use to conquer the problem, it comes at a price. The more guessing you have to do, the further away from the real data the database becomes. Thus, in turn, it can affect the accuracy and validation of the mining results.

Outliers

- Outlier is a commonly user term as it needs attention otherwise it can result in wildly wrong estimations.
- Simply speaking, outlier is an observation that appears far away and diverges from an overall pattern in a sample.
- Outliers are data objects with characteristics that are considerably different than most of the other data objects in the data set
- Data points inconsistent with the majority of data
- Outlier detection can be used for fraud detection or data cleaning

Let’s take an example, we do customer profiling and find out that the average annual income of customers is $0.8 million. But, there are two customers having annual income of $4 and $4.2 million. These two customers annual income is much higher than rest of the population. These two observations will be seen as Outliers.

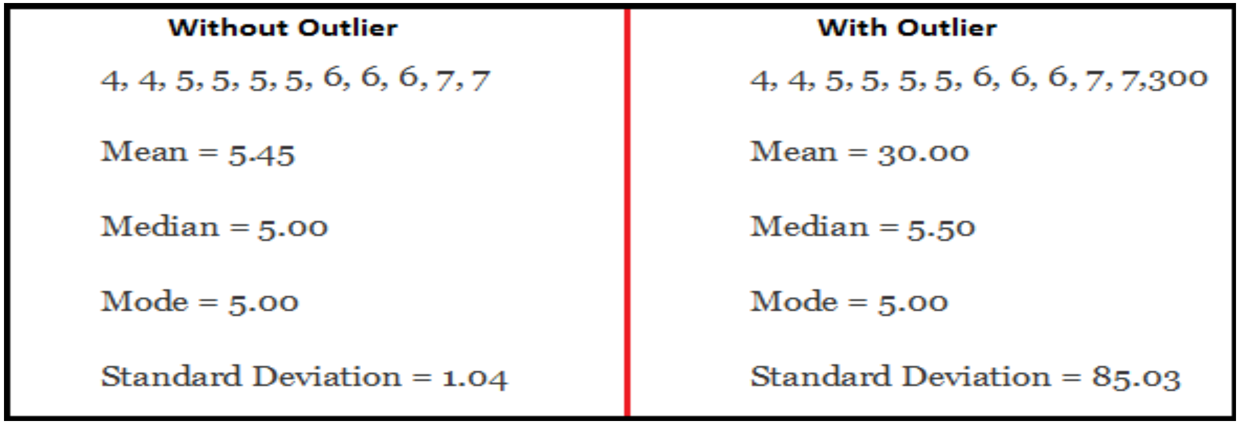


Impact of Outliers

Outliers can drastically change the results of the data analysis and statistical modeling. There are numerous unfavourable impacts of outliers in the data set:

- It increases the error variance and reduces the power of statistical tests
- If the outliers are non-randomly distributed, they can decrease normality
- They can bias or influence estimates that may be of substantive interest
- They can also impact the basic assumption of Regression, ANOVA and other statistical model assumptions.

To understand the impact deeply, let’s take an example to check what happens to a data set with and without outliers in the data set.

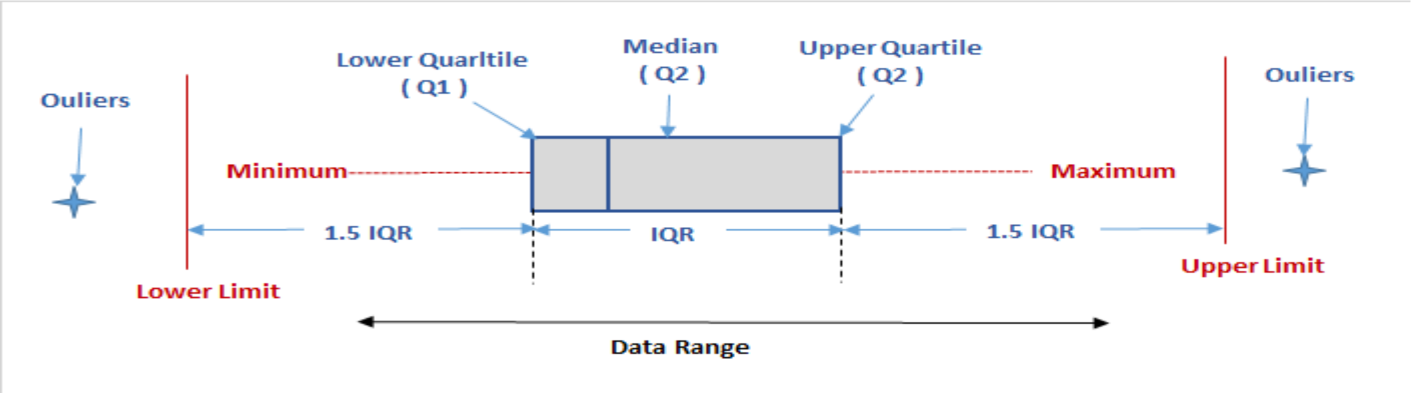


Without Outlier	With Outlier
4, 4, 5, 5, 5, 5, 6, 6, 6, 7, 7	4, 4, 5, 5, 5, 5, 6, 6, 6, 7, 7,300
Mean = 5.45	Mean = 30.00
Median = 5.00	Median = 5.50
Mode = 5.00	Mode = 5.00
Standard Deviation = 1.04	Standard Deviation = 85.03

How to detect Outliers

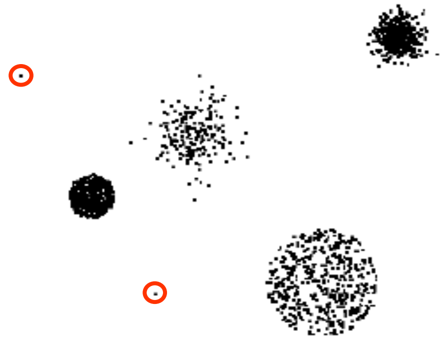
Most commonly used method to detect outliers is visualization. We use various visualization methods, like *Box-plot, Histogram, Scatter plot*. Some analysts also various thumb rules to detect outliers. Some of them are:

- Any value, which is beyond the range of -1.5 x IQR to 1.5 x IQR
- Use capping methods. Any value which out of range of 5th and 95th percentile can be considered as outlier
- Data points, three or more standard deviation away from mean are considered outlier
- Outlier detection is merely a special case of the examination of data for influential data points and it also depends on the business understanding

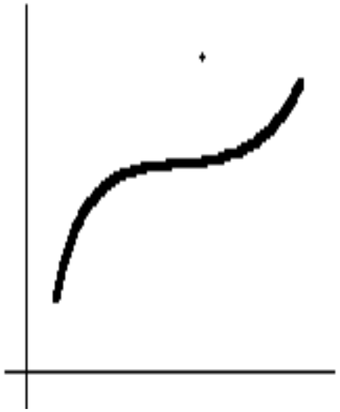


How to detect outliers

- Clustering
  - Outliers may be detected by clustering, where similar values are organized into groups, or “clusters”, Intuitively, values that fall outside of the set of clusters may be considered outliers or very small clusters are outliers



- Combined computer and human inspection
  - Tedious and time consuming
- Curve fitting



How to remove outliers

Deleting observations: We delete outlier values if it is due to data entry error, data processing error or outlier observations are very small in numbers. We can also use trimming at both ends to remove outliers.

Imputing: Like imputation of missing values, we can also impute outliers. We can use mean, median, mode imputation methods. Before imputing values, we should analyse if it is natural outlier or artificial. If it is artificial, we can go with imputing values. We can also use statistical model to predict values of outlier observation and after that we can impute it with predicted values.

Capping the data:

Treat separately: If there are significant number of outliers, we should treat them separately in the statistical model. One of the approach is to treat both groups as two different groups and build individual model for both groups and then combine the output.

Feature Engineering

Feature engineering is the science (and art) of extracting more information from existing data. You are not adding any new data here, but you are actually making the data you already have more useful.

You perform feature engineering once you have completed the first 5 steps in data exploration – Variable Identification, Univariate, Bivariate Analysis, Missing Values Imputation and Outliers Treatment. Feature engineering itself can be divided in 2 steps:

- Variable transformation.
- Variable / Feature creation.

These two techniques are vital in data exploration and have a remarkable impact on the power of prediction.

Data Transformation

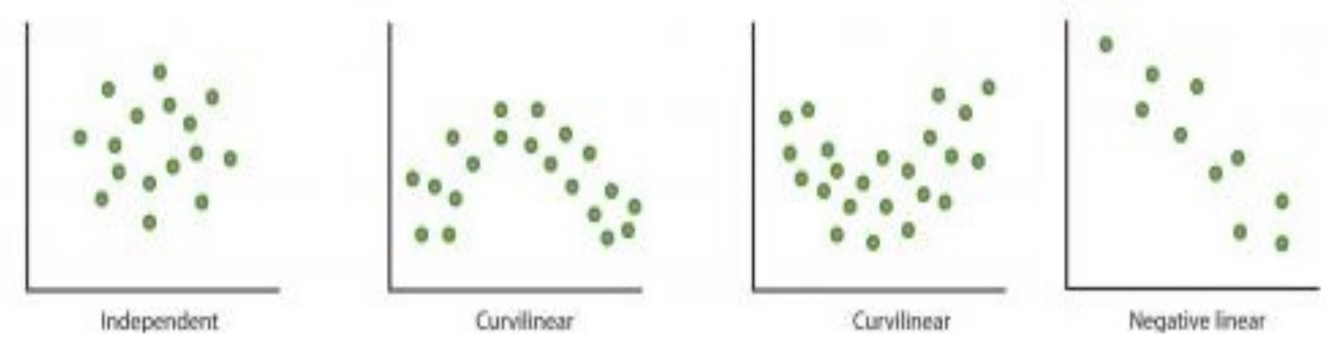
In data modelling, transformation refers to the replacement of a variable by a function. For instance, replacing a variable x by the square / cube root or logarithm x is a transformation. In other words, transformation is a process that changes the distribution or relationship of a variable with others.

When should we use Variable Transformation?

- When we want to **change the scale** of a variable or standardize the values of a variable for better understanding. While this transformation is a must if you have data in different scales, this transformation does not change the shape of the variable distribution

Data Transformation

When we can **transform complex non-linear relationships into linear relationships**. Existence of a linear relationship between variables is easier to comprehend compared to a non-linear or curved relation. Transformation helps us to convert a non-linear relation into linear relation. Scatter plot can be used to find the relationship between two continuous variables. These transformations also improve the prediction. Log transformation is one of the commonly used transformation technique used in these situations.



Data Transformation

When we can **transform complex non-linear relationships into linear relationships**. Existence of a linear relationship between variables is easier to comprehend compared to a non-linear or curved relation. Transformation helps us to convert a non-linear relation into linear relation. Scatter plot can be used to find the relationship between two continuous variables. These transformations also improve the prediction. Log transformation is one of the commonly used transformation technique used in these situations.

Symmetric distribution is preferred over skewed distribution as it is easier to interpret and generate inferences. Some modeling techniques requires normal distribution of variables. So, whenever we have a skewed distribution, we can use transformations which reduce skewness. For right skewed distribution, we take square / cube root or logarithm of variable and for left skewed, we take square / cube or exponential of variables.

Data Transformation

What are the common methods of Variable Transformation?

There are various methods used to transform variables. As discussed, some of them include square root, cube root, logarithmic, binning, reciprocal and many others. Let’s look at these methods in detail by highlighting the pros and cons of these transformation methods.

Logarithm: Log of a variable is a common transformation method used to change the shape of distribution of the variable on a distribution plot. It is generally used for reducing right skewness of variables. Though, It can’t be applied to zero or negative values as well.

Square / Cube root: The square and cube root of a variable has a sound effect on variable distribution. However, it is not as significant as logarithmic transformation. Cube root has its own advantage. It can be applied to negative values including zero. Square root can be applied to positive values including zero.

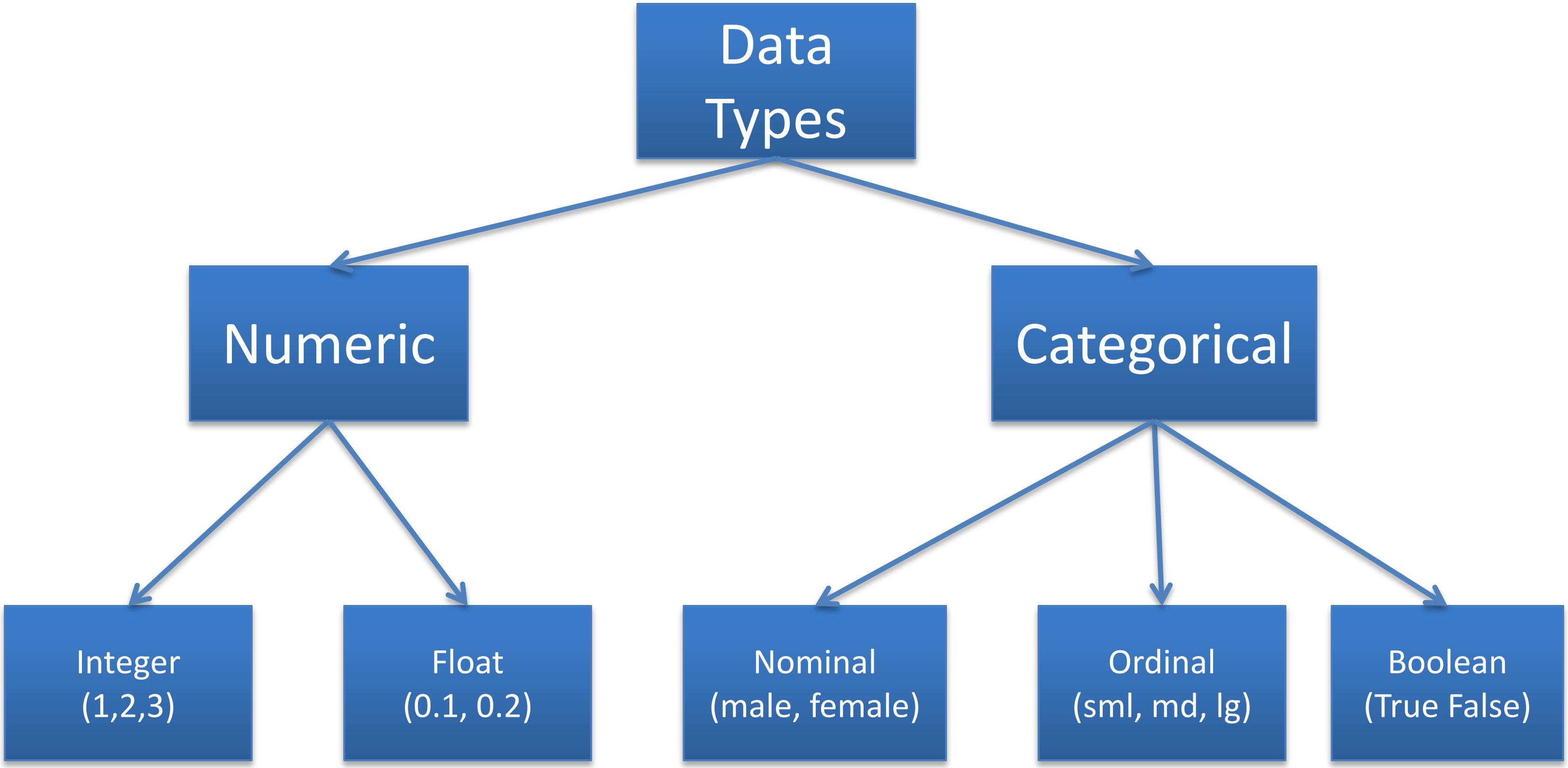
Data Transformation

Binning: It is used to categorize variables. It is performed on original values, percentile or frequency. Decision of categorization technique is based on business understanding. For example, we can categorize income in three categories, namely: High, Average and Low. We can also perform co-variate binning which depends on the value of more than one variables.

Cont'd

Data Transforms

Data transforms are used to change the type or distribution of data variables. Recall that the data may have one of a few types, such as numeric or categorical, with subtypes for each, such as integer and real-valued for numeric, and nominal, ordinal, and boolean for categorical.



Cont'd

- *Normalization* uses a mathematical function to transform numeric columns to a new range. Normalization is important in preventing certain data analysis methods from giving some variables undue influence over others because of differences in the range of their values. In other words, normalization helps to prevent that attributes with large ranges out-weight attributes with small ranges

- There are various ways to do this.

Min-max normalization

$$v' = \frac{v - min_A}{max_A - min_A}(new\_max_A - new\_min_A) + new\_min_A$$

Cont’d

- *Z-Score normalization (Standardization):* On the basis of the Z-score, the numerical data is scaled using the formula of calculating Z values). The data ranges in the interval of -3 to 3.

$$v' = \frac{v - mean_A}{stand\_dev_A}$$

Cont’d

- ***Categorical Data:***

The categorical data can not be directly fed into the model. We have seen machines are black and white, either 1 or 0. So, to use the categorical data for our model building process, we need to create dummy variables. Dummy variables are binary; they can take either the value as 1 or as 0. If we have n types of sub-categories within a categorical column, we must employ n-1 dummy variables. There are two ways to create dummy variables:

Pandas’ function: pd.get_dummies, and

sklearn’s in-built function of OneHotEncoder

Data Preprocessing Steps

There is one more way of dealing with the categorical data, which is to use label encoding. The label encoder does not create dummy variables. However, it labels the categorical variable by numbers like below:

New York –> 1

London –> 2

San Francisco –> 3

There is a limitation of label encoding: it converts the nominal data, which is the categorical data without any order, into ordinal data having order. In the above example, the three cities did not have order. However, the post applying label encoder has values 1,2,3, respectively. The machine will treat this data by giving precedence and treat the numbers as weights like 3 > 2 > 1 will make San Francisco > London > New York. Hence, due to this limitation of label encoding, handling the categorical data is by creating the dummy variables.

Cont'd

We may wish to convert a numeric variable to an ordinal variable in a process called discretization. Alternatively, we may encode a categorical variable as integers or boolean variables, required on most classification tasks.

Discretization Transform: Encode a numeric variable as an ordinal variable.

Ordinal Transform: Encode a categorical variable into an integer variable.

One-Hot Transform: Encode a categorical variable into binary variables.

Feature Selection

Feature selection refers to techniques for selecting a subset of input features that are most relevant to the target variable that is being predicted.

This is important as irrelevant and redundant input variables can distract or mislead learning algorithms possibly resulting in lower predictive performance. Additionally, it is desirable to develop models only using the data that is required to make a prediction, e.g. to favor the simplest possible well performing model.

Data Reduction

- Data is too big to work with. Complex data analysis and mining on huge amounts of data can take a long time. Making such analysis impractical or infeasible.

- Data reduction techniques can be applied to obtain a reduced representation of the data set that is much smaller in volume, yet closely maintains the integrity of the original data, i.e, mining on the reduced data set should be more efficient yet produce the same (or almost the same) analytical results.

- Data reduction strategies
  - Dimensionality reduction — remove unimportant attributes
  - Aggregation and clustering

What is the Best Language for Data Exploration?

The most popular programming tools for data science are currently R and Python, both highly flexible, open source data analytics languages. R is generally best suited for statistical learning as it was built as a statistical language. Python is generally considered the best choice for machine learning with its flexibility for production. The best language for data exploration depends entirely on the application at hand and available tools and technologies.

In this course, we will be concentrating on Python.

Data exploration with python has the advantage in ease of learning, production readiness, integration with common tools, an abundant library, and support from a huge community. Nearly every tool kit and functionality is packaged and can be executed by simply calling the name of a method.

Summary

Data preprocessing in machine learning is the process of preparing the raw data in the form to feed the data into the machine learning model.

The need of data preprocessing is required due to the following reasons:

- The data is more relevant depending on the nature of the business problem.
- It makes the data more reliable and accurate by removing the incorrect, missing or the negative values (based on the domain of the data).
- The data is also more complete after treating for the missing values.
- The data becomes more consistent by eliminating any data quality issues and inconsistencies present in the data.
- The data is in a format that can be parsed to a machine.
- The features of the algorithm are much more interpretable, readability and interpretability of the data improve.

Summary

- Data preparation is a big issue for data mining
- Data preparation includes
  - Data cleaning and data integration
    - Data reduction and feature selection
    - Discretization
- Many methods have been proposed but still an active area of research

- Data preparation is a large subject that can involve a lot of iterations, exploration and analysis. Getting good at data preparation will make life easy for machine learning.