CHAPTER 3: DATA CLEANING AND PREPARATION

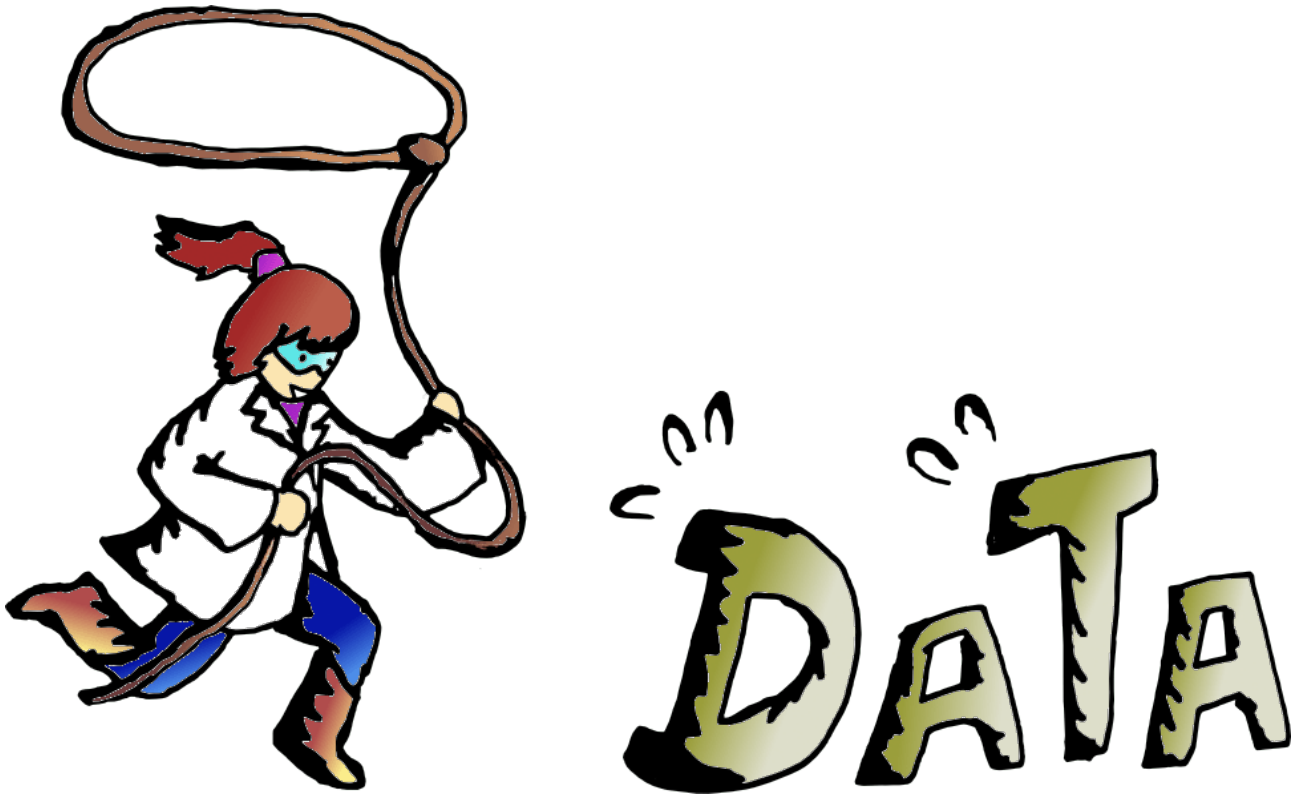
DR. MOHD KHAIRUL BAZLI MOHD AZIZ

PUSAT SAINS MATEMATIK, UMP

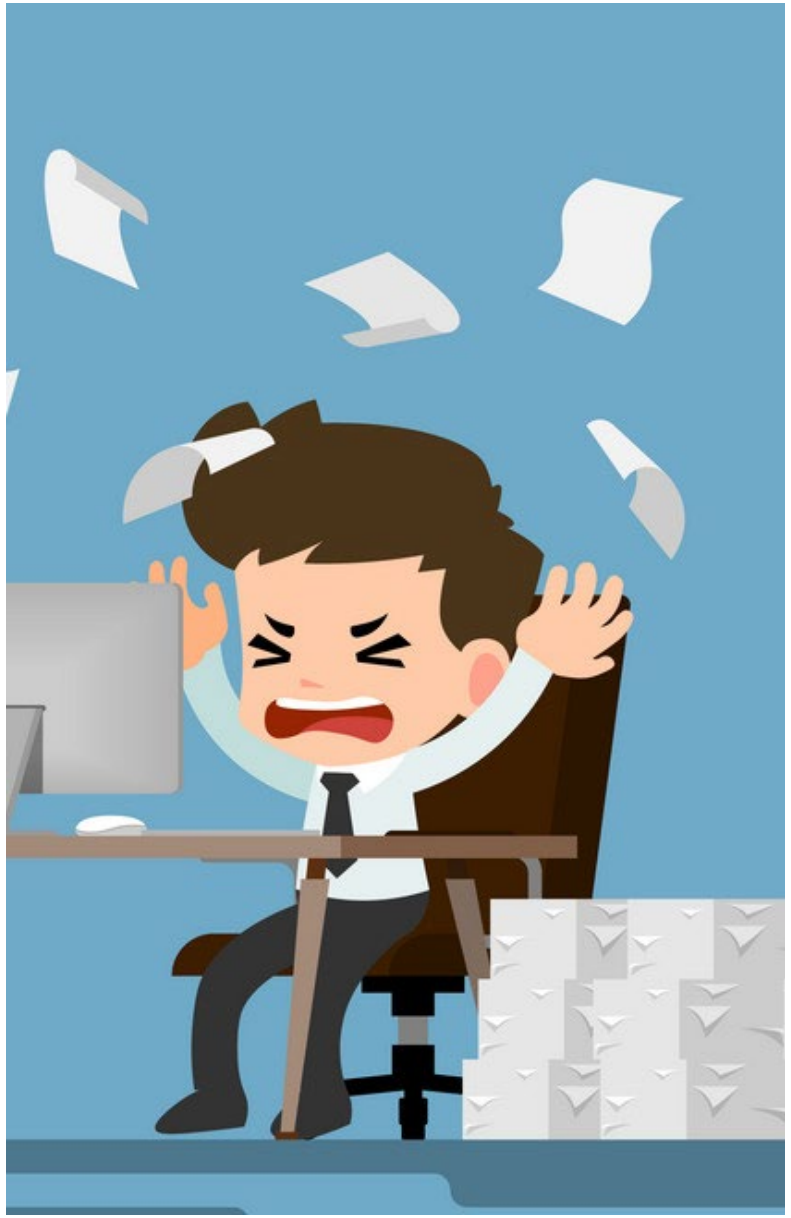
CONTENT

- 3.1 Handling Missing Data
- 3.2 Filtering Out Missing Data
- 3.3 Filling In Missing Data
- 3.4 Identifying and Cleaning Outliers
- 3.5 Normalizing and Standardizing Your Data
- 3.6 Testing with New Data

INTRODUCTION



Suppose you are working on some training data set. You decide to use your favorite classification algorithm only to realize that the training data set contains a mixture of continuous and categorical variables and you'll need to transform some of the variables into a suitable format. You realize that the raw data you have can't be used for your analysis without some manipulation — what you'll soon know as data preparation. You'll need to clean this messy data to get anywhere with it.



WHY DATA PREPARATION?

- Data comes from multitude of sources; it can be high in volume and have variety of attributes.
- Real-world data is generally noisy, incomplete and inconsistent. It implies that raw data tends to be corrupt, have missing values or attributes, outliers or conflicting values.
- Data preparation stage resolves such kinds of data issues to ensure the dataset used for modelling stage is acceptable and of improved quality.
- Analytical models fed with poor quality data can lead to misleading predictions.

WHY IS DATA PREPARATION IMPORTANT?

Preparation of data is mainly to check the data quality. The quality can be checked by the following

- Accuracy: To check whether the data entered is correct or not.
- Completeness: To check whether the data is available or not recorded.
- Consistency: To check whether the same data is kept in all the places that do or do not match.
- Timeliness: The data should be updated correctly.
- Believability: The data should be trustable.
- Interpretability: The understandability of the data.

DATA CLEANING

- Data cleaning is the process of preparing data for analysis by removing or modifying data that is incorrect, incomplete, irrelevant, duplicated, or improperly formatted.
- Data cleaning (or data cleansing), is a pre-processing data and this process will ensure the data is correct, consistent and useable by identifying any errors or corruptions in the data, correcting or deleting them, or manually processing them as needed to prevent the error from happening again.

GOAL OF DATA CLEANING

- The goal of data cleaning is to address data quality issues and to transform the raw data to make it suitable for analysis.
- The data cleaning is aimed to create data sets that are standardized and uniform to allow business intelligence and data analytics tools to easily access and find the right data for each query.
- Quality data will ensure the analysis is valid and the result is reliable. Hence, it will produce correct business decision which increased productivity and in general improved business performances.

DATA QUALITY

- The raw data that is obtained directly from any sources are never in the format that is ready to be analysed.
- Data quality is important since inaccurate data can have an impact on results or business decision.
- The input from domain knowledge expert is also essential to making informed decisions on how to handle incomplete or incorrect data.
- Example of data quality:
  1. inconsistent data e.g. one customer with two different address (duplication records)
  2. missing data e.g. missing customer age in demographic study
  3. invalid data e.g. invalid IC number 6611-11-7042

DIRTY DATA

Dirty data refers to any data that takes away the data integrity of the entire dataset. Below are some of the examples.

- Data errors such as misspelled data, typos, duplicate data, erroneously parsed data.
- Data that violate business rules may not be easily fixed even if it is identified. More often than not, the business needs to review such data.
- Data can be consistently generated by systems that have entity constraint issues, bugs, and legacy "patching" placed inside the systems. Often, this data will look "consistent" with all other data. But, upon close inspection, this data is simply wrong.
- Data can be collected using wrong method, or wrong population. This data often comes from asking the wrong business question.
- Data can be calculated using inconsistent codebase, modules, and Application Programming Interface (API). The magnitude of the data errors might not be large for individual transactions.

Id	Name	DoB	Age	Gender	Phone	Country
1801	Shah Rukh Khan	11/12/1984	34	Male	5551212	India
1802	Roselinda	14/13/1986	32	Female	4568765	Kuala Lumpur
1803	Muhammad Ali Jannah	31/08/1983	35	M	5678900	Jordan
1804	Lynda Carter	12/30/1980	38	Female	9999999	America
1805	Smith, Tracy	23/08/1981	37	2	6856262	UK
1806	Ng Chee Chin	3/10/1989	19	Fenale	3209876	Malaysia
1807	Fatoush Olkan	18/07/1982	36	-	2348765	Turkey
1808	John Doe	20/11/1987	31	Male	7735075	USA
1809	Tracy Smith	23/08/1981	37	2	8356753	UK
1809	Ibrar Yaacob	18/09/1974	44	Male	6544321	Pakistan

IS THIS AN ACCEPTABLE DATASET?

Id	Name	DoB	Age	Gender	Phone	Country
1801	Shah Rukh Khan	11/12/1984	34	Male	5551212	India
1802	Roselinda	14/13/1986	32	Female	4568765	Kuala Lumpur
1803	Muhammad Ali Jannah	31/08/1983	35	M	5678900	Jordan
1804	Lynda Carter	12/30/1980	38	Female	9999999	America
1805	Smith, Tracy	23/08/1981	37	2	6856262	UK
1806	Ng Chee Chin	3/10/1989	19	Fenale	3209876	Malaysia
1807	Fatoush Olkan	18/07/1982	36	-	2348765	Turkey
1808	John Doe	20/11/1987	31	Male	7735075	USA
1809	Tracy Smith	23/08/1981	37	2	8356753	UK
1809	Ibrar Yaacob	18/09/1974	44	Male	6544321	Pakistan

ISSUES TO HIGHLIGHT

There are some issues raise here:

1. DoB column, there are unacceptable for 14/13/1986 and 12/30/1980.
2. Gender column, doesn't synchronously written for gender type.
3. Country column, Kuala Lumpur doesn't correctly represent the Country.

COMMON TYPES OF DIRTY DATA

1. Incomplete data: Most common occurrence of dirty data. Important fields on master data records, useful to the business, are often left blank. For example, if you haven't classified your customers by industry, you cannot segment your sales and marketing initiatives by industry.
2. Duplicate data: Very common. Most companies deal with issues such as duplicate customer records, but duplicate materials are also very common. This can be costly to companies due to excess in inventory and sub-optimal procurement decisions.
3. Incorrect data: Incorrect data can occur when field values are created outside of the valid range of values. For example, the value in a month field should range from 1 to 12 or a street address should be a real address.

COMMON TYPES OF DIRTY DATA

4. Inaccurate data: It is possible or data to be technically correct but inaccurate given the business context. Costly business interruptions are often rooted in inaccurate data. For example, minor errors in customer addresses can result in deliveries at wrong locations even though the addresses are actual addresses.
5. Business rule violations: There are often large collections of poorly documented business rules associated with master data that are specific to the industry or business context. For example, beverage products should have a Unit of Measure in 'fl. oz.' or payment terms for a certain type of customers should always be 'Net 30.'
6. Inconsistent data: Data redundancy–i.e., the same field values stored in different places-often leads to inconsistencies. For example, most companies have customer information in multiple systems and the data is often not kept in sync.

NOISY DATA

Noisy data is a meaningless data that can't be interpreted by machines. It can be generated due to faulty data collection, data entry errors etc. It can be handled in following ways :

1. Binning Method:

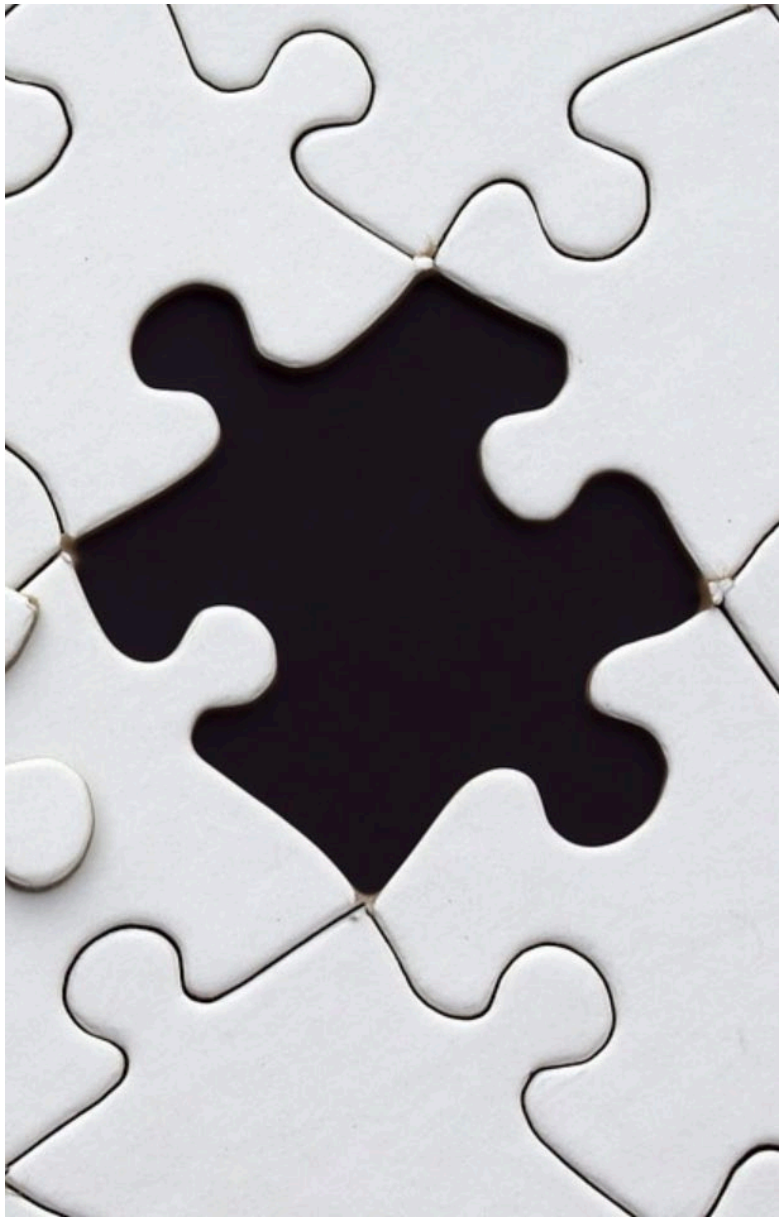
   This method works on sorted data in order to smooth it. The whole data is divided into segments of equal size and then various methods are performed to complete the task. Each segmented is handled separately. One can replace all data in a segment by its mean or boundary values can be used to complete the task.

2. Regression:

   Here data can be made smooth by fitting it to a regression function. The regression used may be linear (having one independent variable) or multiple (having multiple independent variables).

3. Clustering:

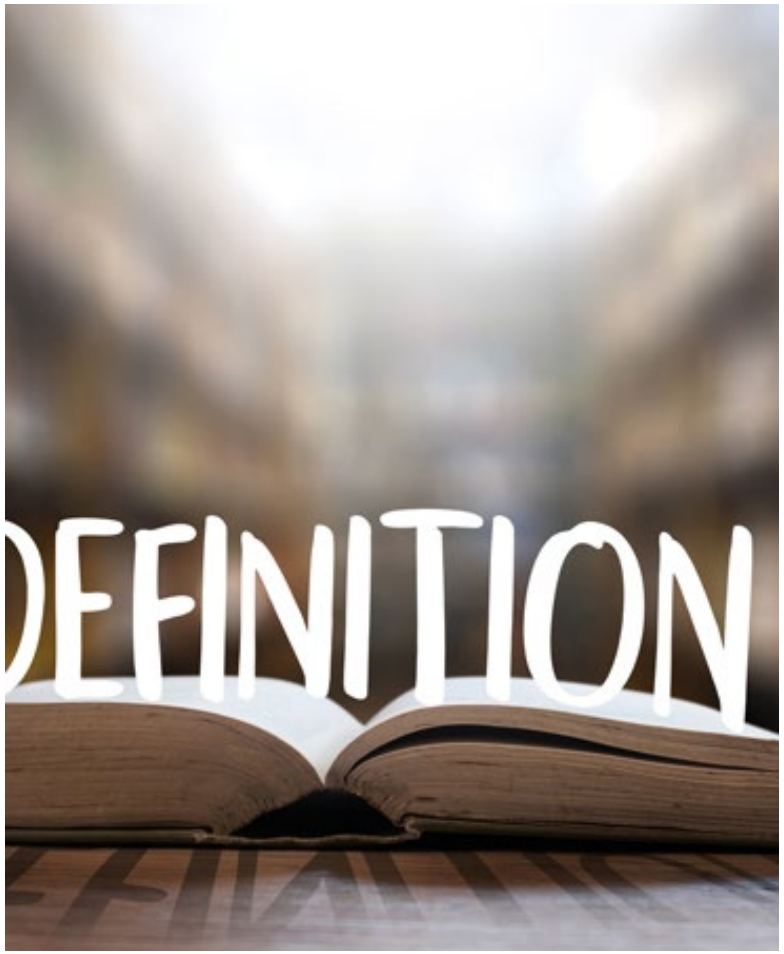
   This approach groups the similar data in a cluster. The outliers may be undetected or it will fall outside the clusters.



MISSING DATA

- This situation arises when some data is missing in the data. It can be handled in various ways.

3.1 HANDLING MISSING DATA



Missing data – Why does it matter so much?

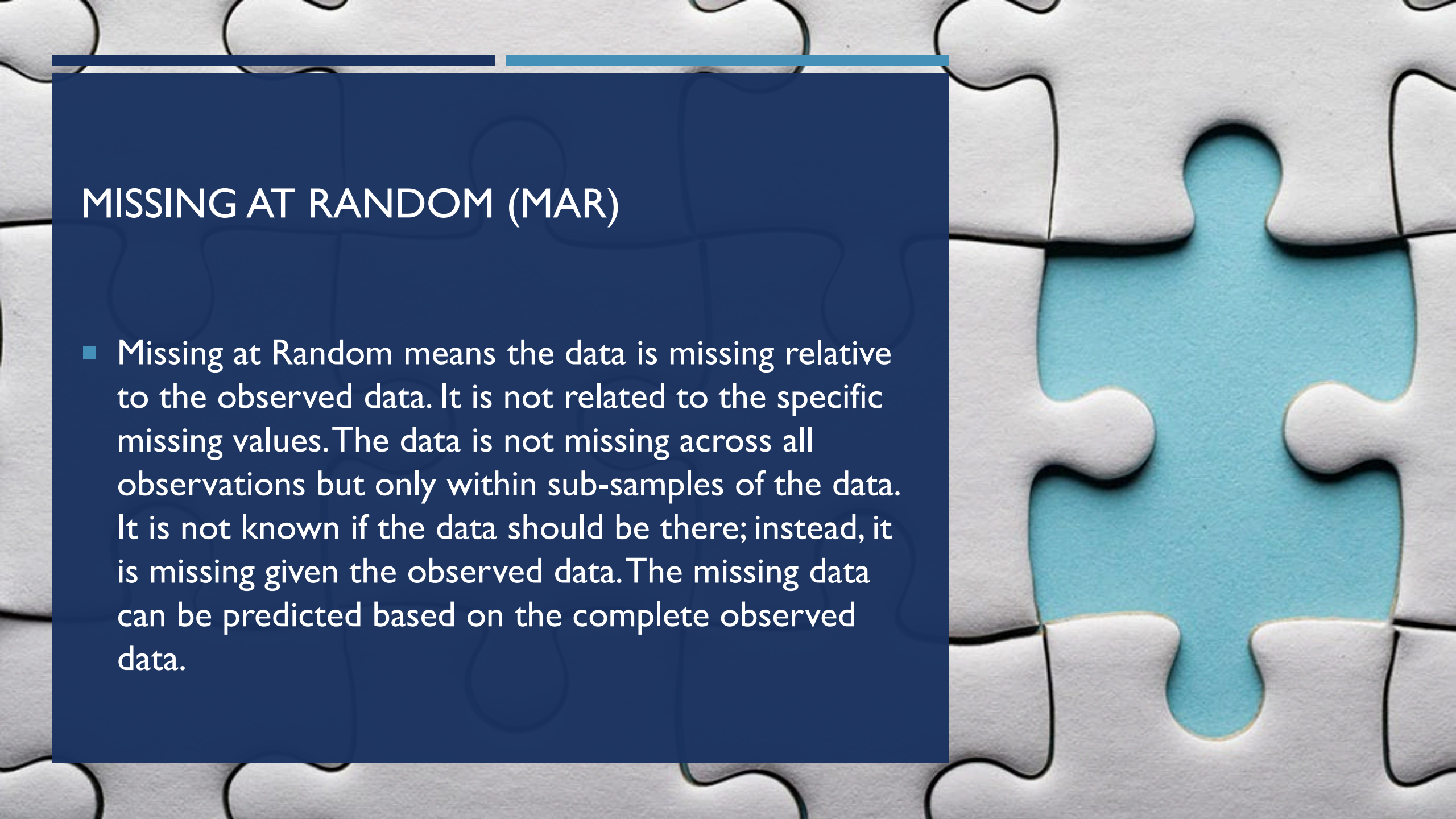
- Missing data is a huge problem for data analysis because it distorts findings. It's difficult to be fully confident in the insights when you know that some entries are missing values. Hence, why they must be addressed.
- According to data scientists, there are three types of missing data.
  1. Missing Completely at Random (MCAR)
  2. Missing At Random (MAR)
  3. Missing Not at Random (MNAR)

MISSING COMPLETELY AT RANDOM (MCAR)

- In the MCAR situation, the data is missing across all observations regardless of the expected value or other variables. Data scientists can compare two sets of data, one with missing observations and one without. Using a t-test, if there is no difference between the two data sets, the data is characterized as MCAR.
- Data may be missing due to test design, failure in the observations or failure in recording observations. This type of data is seen as MCAR because the reasons for its absence are external and not related to the value of the observation.

MISSING AT RANDOM (MAR)

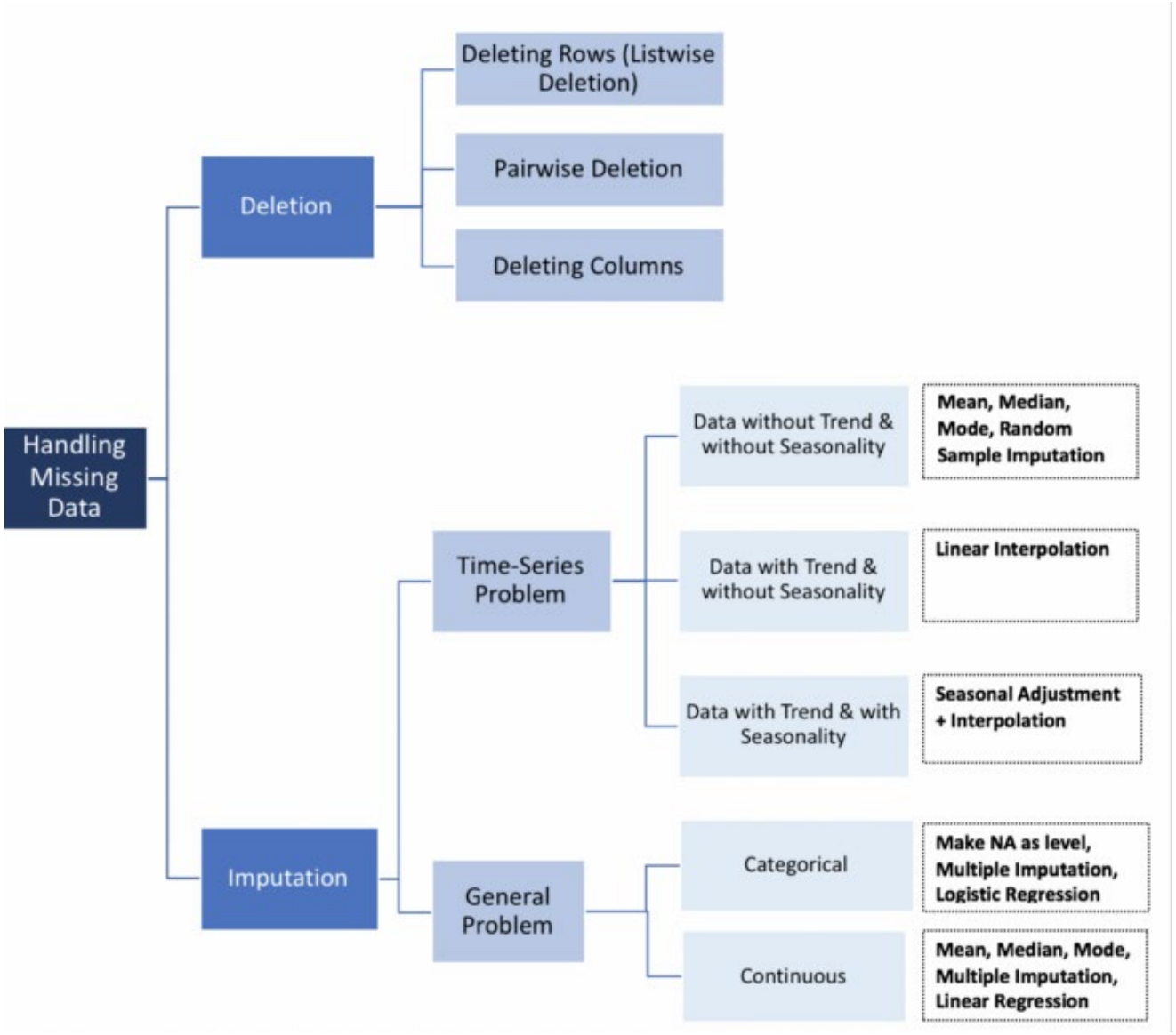
- Missing at Random means the data is missing relative to the observed data. It is not related to the specific missing values. The data is not missing across all observations but only within sub-samples of the data. It is not known if the data should be there; instead, it is missing given the observed data. The missing data can be predicted based on the complete observed data.



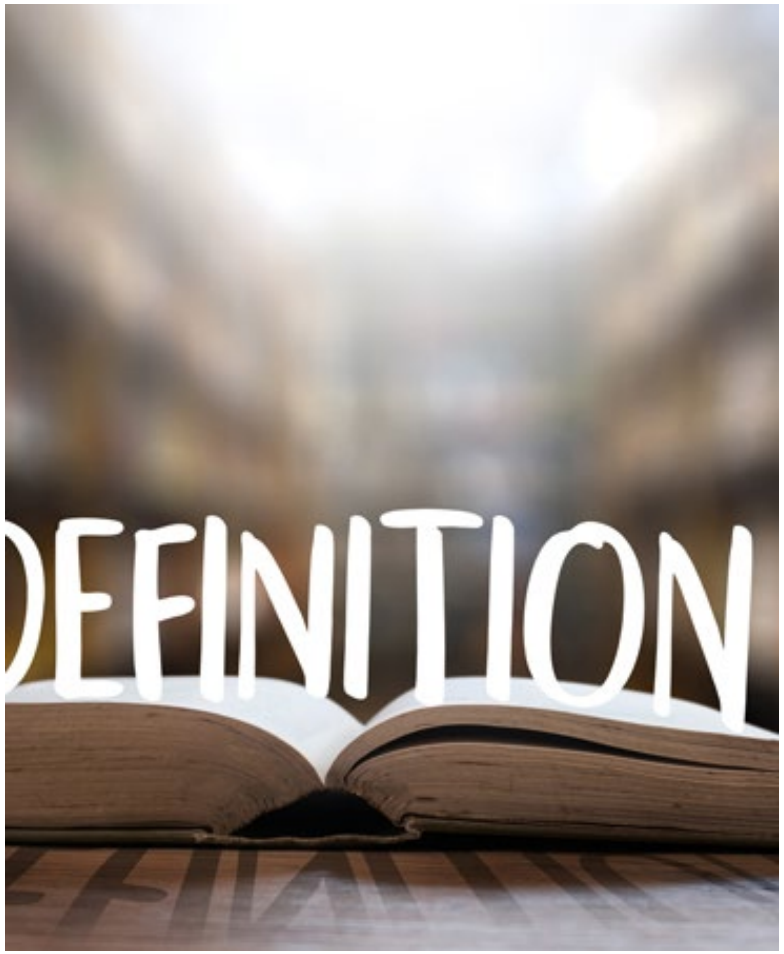
MISSING NOT AT RANDOM (MNAR)

- The MNAR category applies when the missing data has a structure to it. In other words, there appear to be reasons the data is missing. In a survey, perhaps a specific group of people – say women ages 45 to 55 – did not answer a question. Like MAR, the data cannot be determined by the observed data, because the missing information is unknown. Data scientists must model the missing data to develop an unbiased estimate. Simply removing observations with missing data could result in a model with bias.

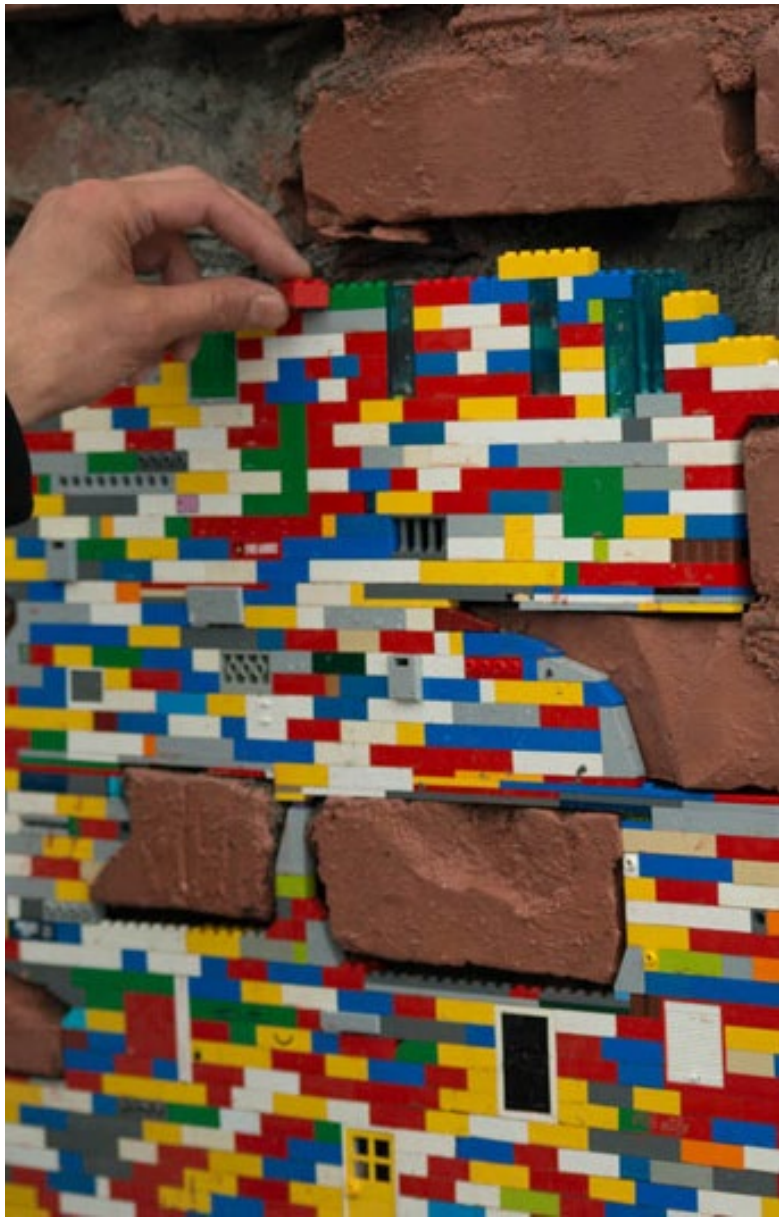
HANDLING MISSING DATA



3.2 FILTERING OUT MISSING DATA



- Find columns with missing data
- Get a list of columns with missing data
- Get the number of missing data per column
- Get the column with the maximum number of missing data
- Get the number total of missing data in the DataFrame
- Remove columns that contains more than 50% of missing data
- Find rows with missing data
- Get a list of rows with missing data
- Get the number of missing data per row
- Get the number of missing data for a given row
- Get the row with the largest number of missing data
- Remove rows with missing data

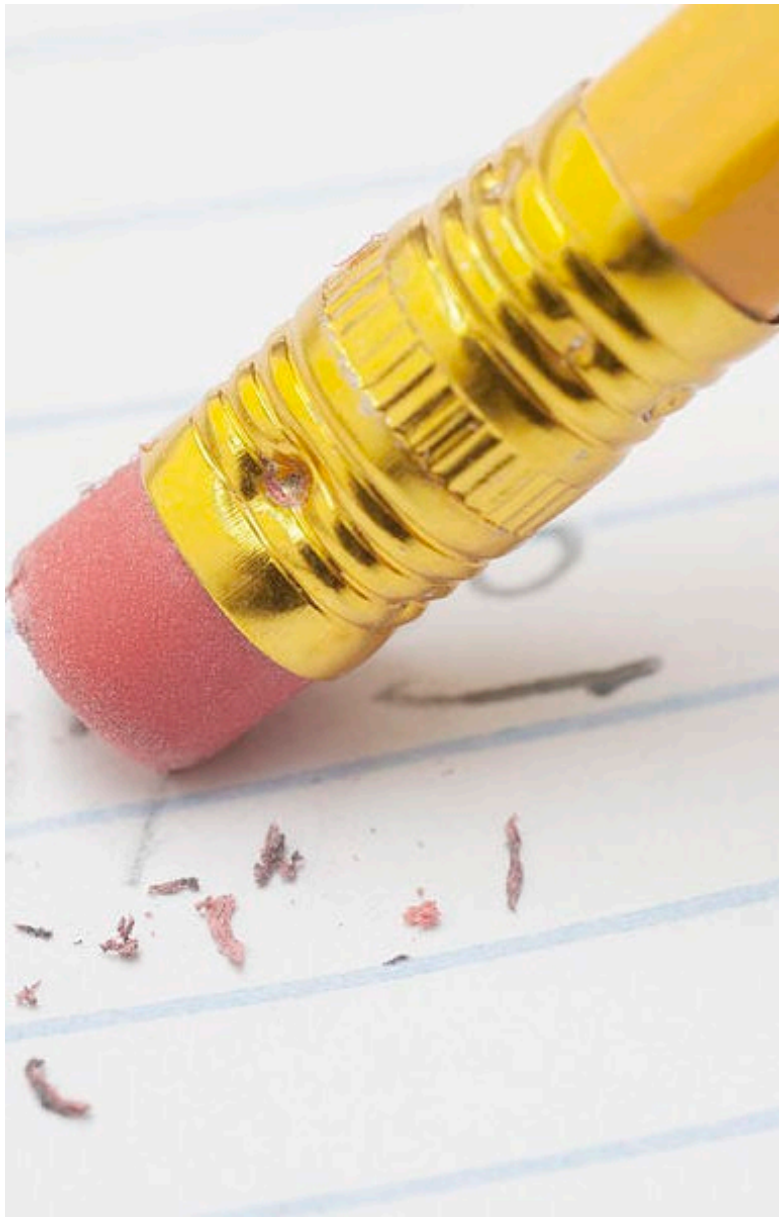


3.3 FILLING IN MISSING DATA

- For filling missing values, there are many methods available. For choosing the best method, you need to understand the type of missing value and its significance, before you start filling/deleting the data.
- When dealing with missing data, data scientists can use two primary methods to solve the error: imputation or the removal of data.

IMPUTATION VS. REMOVING DATA

- The imputation method develops reasonable guesses for missing data. It's most useful when the percentage of missing data is low. If the portion of missing data is too high, the results lack natural variation that could result in an effective model.
- The other option is to remove data. When dealing with data that is missing at random, related data can be deleted to reduce bias. Removing data may not be the best option if there are not enough observations to result in a reliable analysis. In some situations, observation of specific events or factors may be required.



REMOVE DATA

- There are two primary methods for deleting data when dealing with missing data: listwise and dropping variables.

Listwise

- In this method, all data for an observation that has one or more missing values are deleted. The analysis is run only on observations that have a complete set of data. If the data set is small, it may be the most efficient method to eliminate those cases from the analysis. However, in most cases, the data are not missing completely at random (MCAR). Deleting the instances with missing observations can result in biased parameters and estimates and reduce the statistical power of the analysis.

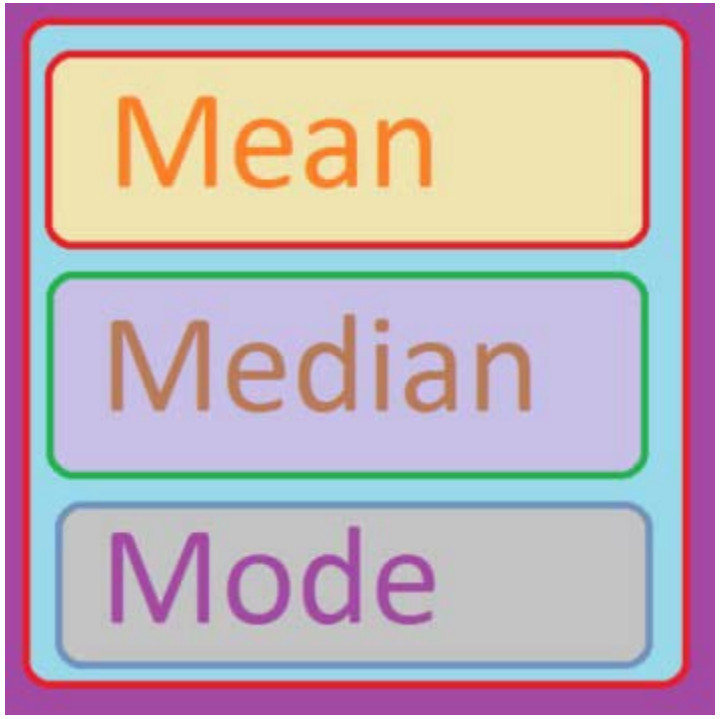
Pairwise

- Pairwise deletion assumes data are missing completely at random (MCAR), but all the cases with data, even those with missing data, are used in the analysis. Pairwise deletion allows data scientists to use more of the data. However, the resulting statistics may vary because they are based on different data sets. The results may be impossible to duplicate with a complete set of data.

IMPUTATION

- When data is missing, it may make sense to delete data, as mentioned above. However, that may not be the most effective option. For example, if too much information is discarded, it may not be possible to complete a reliable analysis. Or there may be insufficient data to generate a reliable prediction for observations that have missing data.
- Instead of deletion, data scientists have multiple solutions to impute the value of missing data. Depending why the data are missing, imputation methods can deliver reasonably reliable results. These are examples of single imputation methods for replacing missing data.

MEAN, MEDIAN AND MODE



- This is one of the most common methods of imputing values when dealing with missing data. In cases where there are a small number of missing observations, data scientists can calculate the mean or median of the existing observations. However, when there are many missing variables, mean or median results can result in a loss of variation in the data. This method does not use time-series characteristics or depend on the relationship between the variables.

REGRESSION ANALYSIS

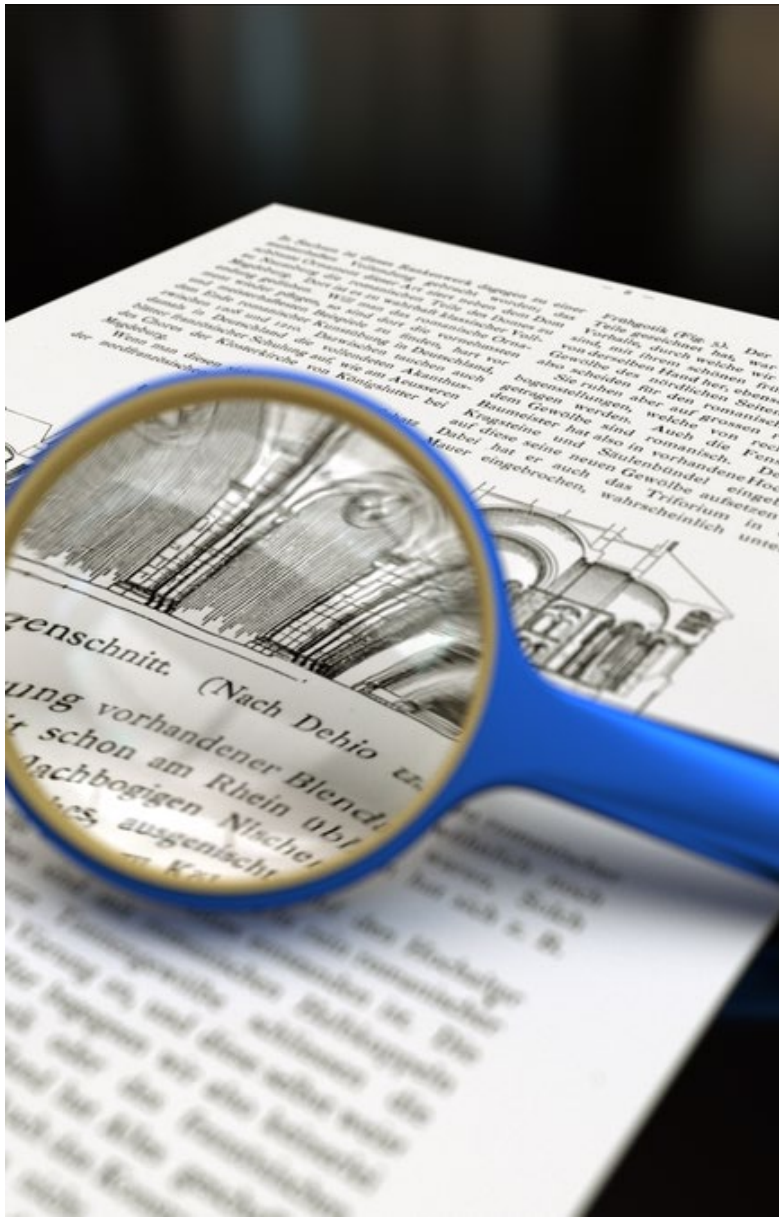
- Regression is useful for handling missing data because it can be used to predict the null value using other information from the dataset. There are several methods of regression analysis, like Stochastic regression. Regression methods can be successful in finding the missing data, but this largely depends on how well connected the remaining data is. Of course, the one drawback with regression analysis is that it requires significant computing power, which could be a problem if data scientists are dealing with a large dataset.

LINEAR REGRESSION

- Linear regression is often used to approximate a value of some function by using two known values of that function at other points. This formula can also be understood as a weighted average. The weights are inversely related to the distance from the end points to the unknown point. The closer point has more influence than the farther point.
- When dealing with missing data, you should use this method in a time series that exhibits a trend line, but it's not appropriate for seasonal data.

3.4 IDENTIFYING AND CLEANING OUTLIERS

- Outliers are values in data that differ extremely from a major sample of the data, the presence of outliers can significantly reduce the performance and accuracy of a predictable model.



OUTLIERS IDENTIFICATION

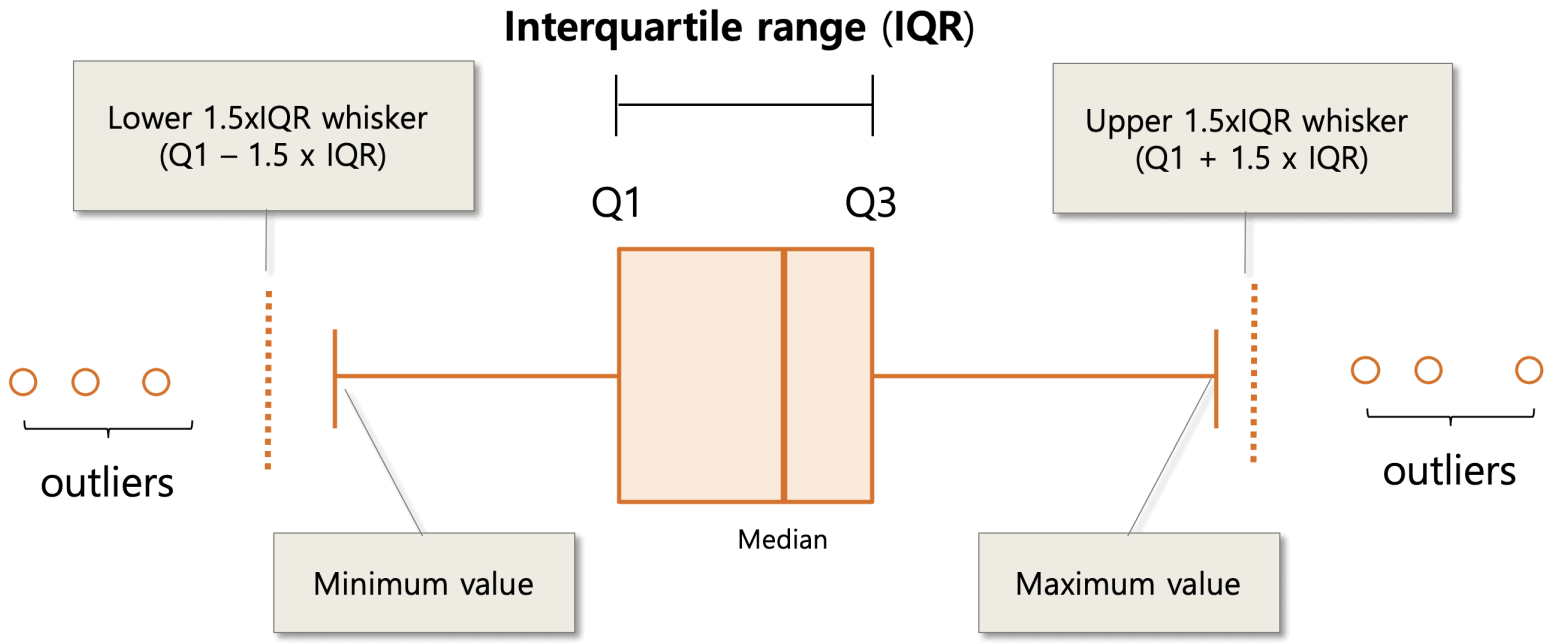
- There are different ways and methods of identifying outliers, but we are only going to use some of the most popular techniques:
  1. Visualization
  2. Skewness
  3. Interquartile Range
  4. Standard Deviation

VISUALIZATION

- Outliers can be detected using different visualization methods, we are going to use :
  1. Boxplot
  2. Histogram
  3. Scatter Plot

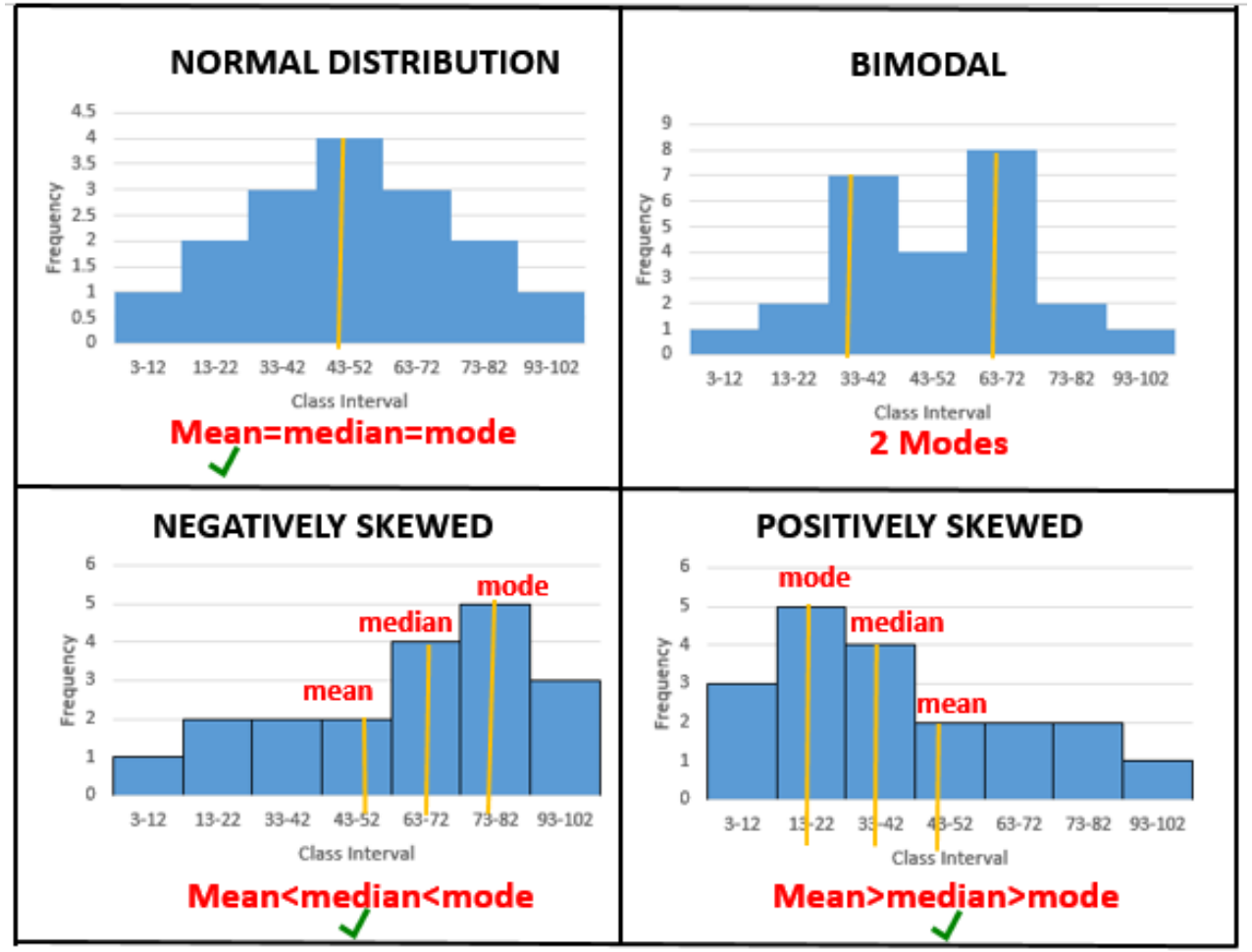
BOXPLOT

- Boxplot is a visualization tool for identifying outliers, it displays the distribution of statistical observations, its body is classified into four parts; the lowest and the highest(minimum and maximum), the 25 percentile(first quartile(Q1)), the median(50th percentile), the 75th percentile(third quartile(Q3)).
- Outliers appears above or below the minimum and maximum of the boxplot.



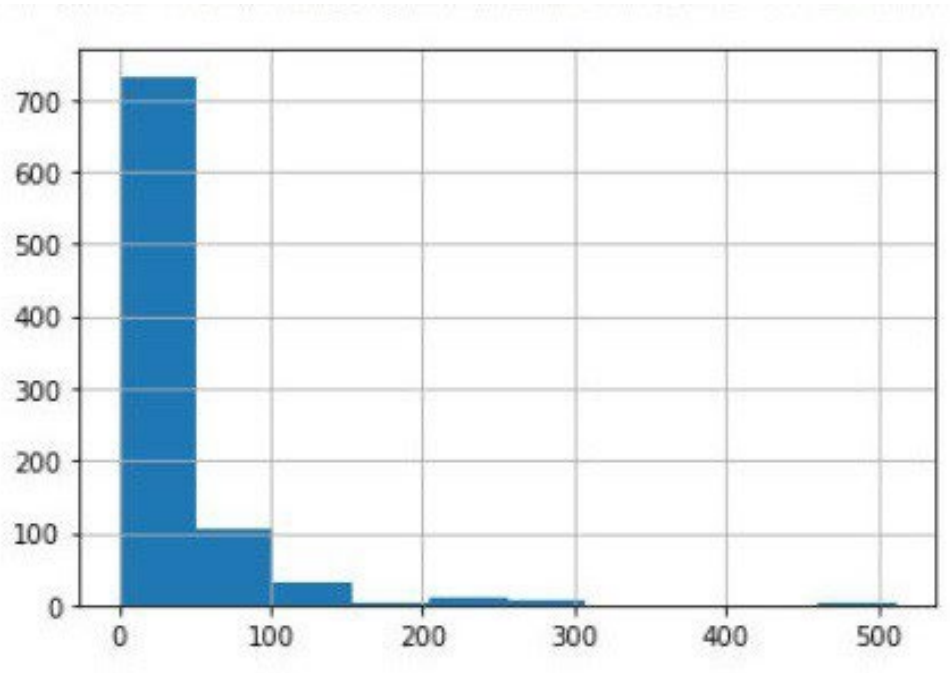
HISTOGRAM

- To visualize the distribution of a numerical variable, a histogram shows the direction in which these variables are distributed, outliers will appear outside the overall distribution of the data. if the histogram is right-skewed or left-skewed, it indicates the presence of extreme values or outliers.



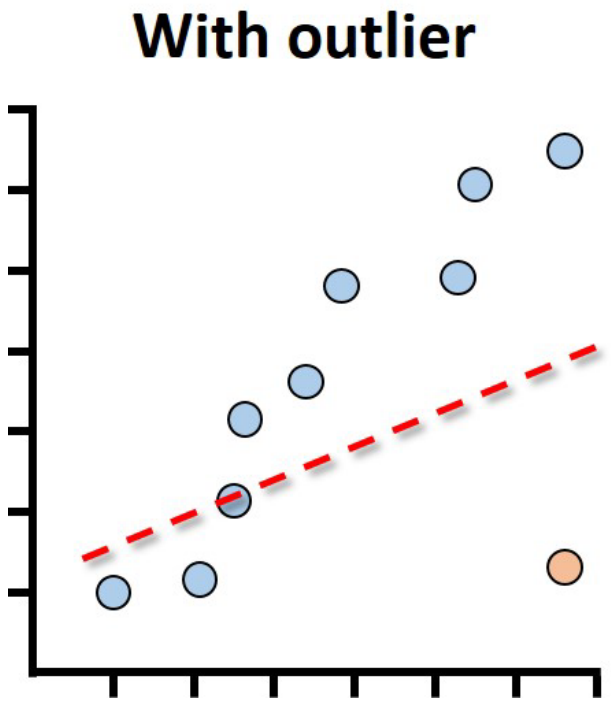
SKEWNESS

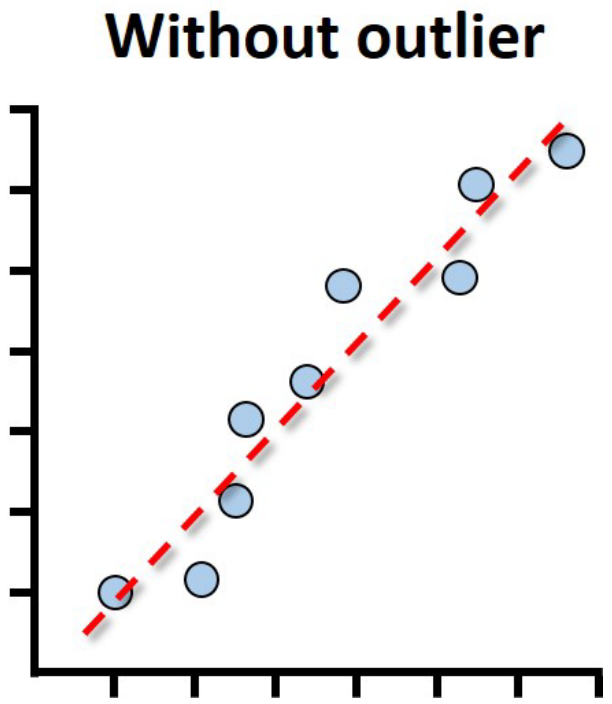
- The skewness value should be within the range of -1 to 1 for a normal distribution, any major changes from this value may indicate the presence of outliers.



SCATTER PLOT

- A scatter plot , is a type of plot or mathematical diagram using Cartesian coordinates to display values for typically two variables for a set of data. The data are displayed as a collection of points, each having the value of one variable determining the position on the horizontal axis and the value of the other variable determining the position on the vertical axis.
- Scatter plots often have a pattern. We call a data point an outlier if it doesn't fit the pattern.





INTERQUARTILE RANGE(IQR)

- Box plot use the IQR method to display data and outliers(shape of the data) but in order to be get a list of identified outlier, we will need to use the mathematical formula and retrieve the outlier data.
- IQR = Q3 − Q1

Z-SCORE

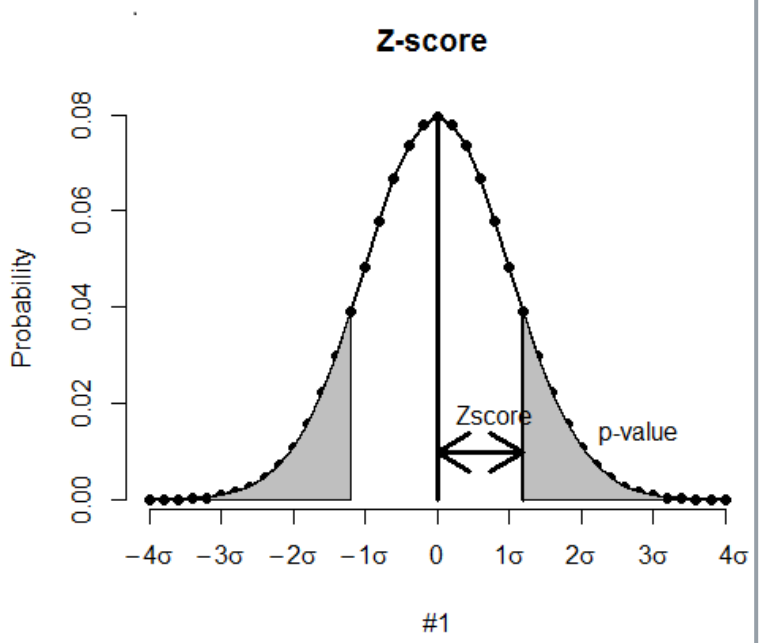
- Z-Score is the number of standard deviation by which the value of an observation or data point is above or below the observed mean value.
- The intuition behind the Z-Score is to describe any point by finding their relationship with standard deviation and mean of the group of data points.
- Z-Score ranges between 3 to -3,so, if Z-Score of a datapoint is greater than or lesser than the range that datapoint will be treated as an outlier.

Score

Mean

$$Z = \frac{x - \mu}{\sigma}$$

SD



CLEANING OUTLIERS

- Now that we know how to detect the outliers, it is important to understand if they needs to be removed or corrected. In this section we will consider a few methods of removing the outliers and if required imputing new values.

WORKING WITH OUTLIERS: CORRECTING, REMOVING

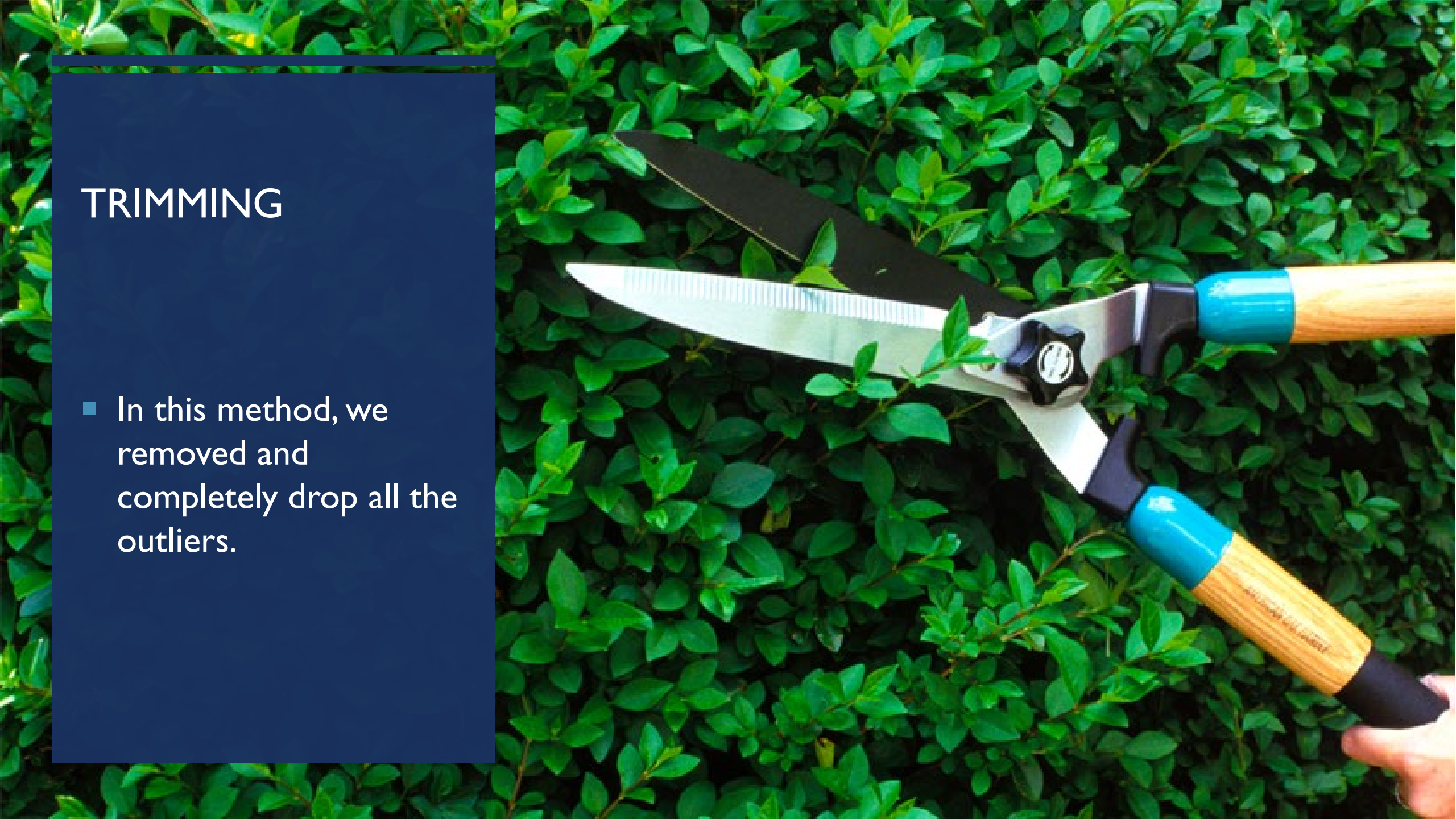
- Flooring and Capping.
- Trimming.
- Replacing outliers with the mean, median, mode, or other values.

FLOORING AND CAPPING

- In this quantile-based technique, we will do the flooring (e.g 25th percentile) for the lower values and capping(e.g for the 75th percentile) for the higher values. These percentile values will be used for the quantile-based flooring and capping.

TRIMMING

- In this method, we removed and completely drop all the outliers.



REPLACING OUTLIERS WITH THE MEAN, MEDIAN, MODE, OR OTHER VALUES

- In this technique, we replace the extreme values with the mode value, you can use median or mean value but it is advised not to use the mean values because it is highly susceptible to outliers.

3.5 NORMALIZING AND STANDARDIZING YOUR DATA

- In practice, we often encounter different types of variables in the same dataset. A significant issue is that the range of the variables may differ a lot. Using the original scale may put more weights on the variables with a large range. In order to deal with this problem, we need to apply the technique of features rescaling to independent variables or features of data in the step of data pre-processing. The terms normalisation and standardisation are sometimes used interchangeably, but they usually refer to different things.
- The goal of applying Feature Scaling is to make sure features are on almost the same scale so that each feature is equally important and make it easier to process by most ML algorithms.

NORMALIZATION

- Normalization is a scaling technique in which values are shifted and rescaled so that they end up ranging between 0 and 1. It is also known as Min-Max scaling.

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)}$$

- Here, Xmax and Xmin are the maximum and the minimum values of the feature respectively.
  - When the value of X is the minimum value in the column, the numerator will be 0, and hence X' is 0
  - On the other hand, when the value of X is the maximum value in the column, the numerator is equal to the denominator and thus the value of X' is 1
  - If the value of X is between the minimum and the maximum value, then the value of X' is between 0 and 1

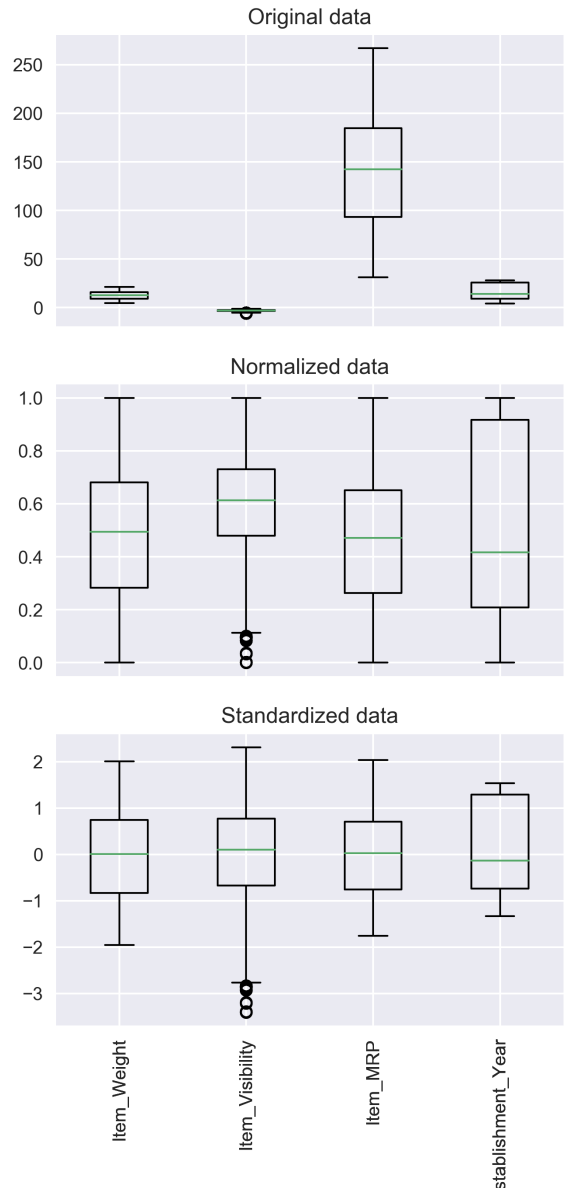
STANDARDIZATION

- Standardization is another scaling technique where the values are centered around the mean with a unit standard deviation. This means that the mean of the attribute becomes zero and the resultant distribution has a unit standard deviation.

$$x_{scaled} = \frac{x - mean}{sd}$$

THE BIG QUESTION – NORMALIZE OR STANDARDIZE?

- Normalization is good to use when you know that the distribution of your data does not follow a Gaussian distribution. This can be useful in algorithms that do not assume any distribution of the data like K-Nearest Neighbors and Neural Networks.
- Standardization, on the other hand, can be helpful in cases where the data follows a Gaussian distribution. However, this does not have to be necessarily true. Also, unlike normalization, standardization does not have a bounding range. So, even if you have outliers in your data, they will not be affected by standardization.
- However, at the end of the day, the choice of using normalization or standardization will depend on our problem and the machine learning algorithm we are using. There is no hard and fast rule to tell us when to normalize or standardize our data. We can always start by fitting your model to raw, normalized and standardized data and compare the performance for best results.



- We can see the comparison between our unscaled and scaled data using boxplots.
- You can notice how scaling the features brings everything into perspective. The features are now more comparable and will have a similar effect on the learning models.

EXAMPLE

- This is a dataset that contains an independent variable (Purchased) and 3 dependent variables (Country, Age, and Salary). We can easily notice that the variables are not on the same scale because the range of Age is from 27 to 50, while the range of Salary going from 48 K to 83 K. The range of Salary is much wider than the range of Age. This will cause some issues in our models since a lot of machine learning models such as k-means clustering and nearest neighbour classification are based on the Euclidean Distance.

	Country	Age	Salary	Purchased
1	France	44	72000	No
2	Spain	27	48000	Yes
3	Germany	30	54000	No
4	Spain	38	61000	No
5	Germany	40		Yes
6	France	35	58000	Yes
7	Spain		52000	No
8	France	48	79000	Yes
9	Germany	50	83000	No
10	France	37	67000	Yes

The range of Age: 27 - 50

The range of Salary:48,000 - 83,000

```
dataset['Age'].min()
```

```
27.0
```

```
dataset['Salary'].min()
```

```
48000.0
```

```
dataset['Age'].max()
```

```
50.0
```

```
dataset['Salary'].max()
```

```
83000.0
```

EXAMPLE

- When we calculate the equation of Euclidean distance, the number of $(x2-x1)^2$ is much bigger than the number of $(y2-y1)^2$ which means the Euclidean distance will be dominated by the salary if we do not apply feature scaling. The difference in Age contributes less to the overall difference. Therefore, we should use Feature Scaling to bring all values to the same magnitudes and, thus, solve this issue. To do this, there are primarily two methods called Standardisation and Normalisation.

	Country	Age	Salary	Purchased
1	France	44	72000	No
2	Spain	27	48000	Yes
3	Germany	30	54000	No
4	Spain	38	61000	No
5	Germany	40		Yes
6	France	35	58000	Yes
7	Spain		52000	No
8	France	48	79000	Yes
9	Germany	50	83000	No
10	France	37	67000	Yes

The range of Age: 27 - 50

The range of Salary:48,000 - 83,000

```
dataset['Age'].min()
```

```
27.0
```

```
dataset['Salary'].min()
```

```
48000.0
```

```
dataset['Age'].max()
```

```
50.0
```

```
dataset['Salary'].max()
```

```
83000.0
```

Standardisation

	Age	Salary
0	0.758874	7.494733e-01
1	-1.711504	-1.438178e+00
2	-1.275555	-8.912655e-01
3	-0.113024	-2.532004e-01
4	0.177609	6.632192e-16
5	-0.548973	-5.266569e-01
6	0.000000	-1.073570e+00
7	1.340140	1.387538e+00
8	1.630773	1.752147e+00
9	-0.258340	2.937125e-01

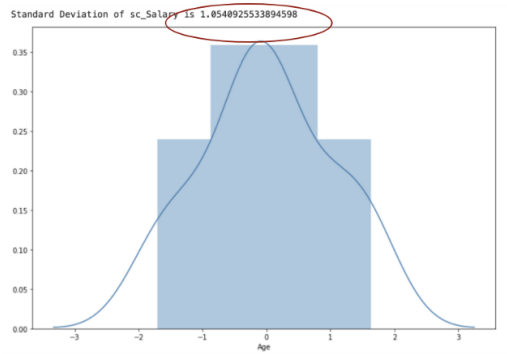
Max-Min Normalization

	Age	Salary
0	0.739130	0.685714
1	0.000000	0.000000
2	0.130435	0.171429
3	0.478261	0.371429
4	0.565217	0.450794
5	0.347826	0.285714
6	0.512077	0.114286
7	0.913043	0.885714
8	1.000000	1.000000
9	0.434783	0.542857

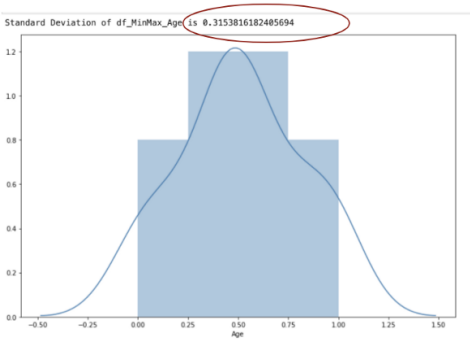
Column:Age

Standard Deviation (Age):
Max-Min Normalization (0.315) < Standardisation (1.05)

Standardisation



Max-Min Normalisation



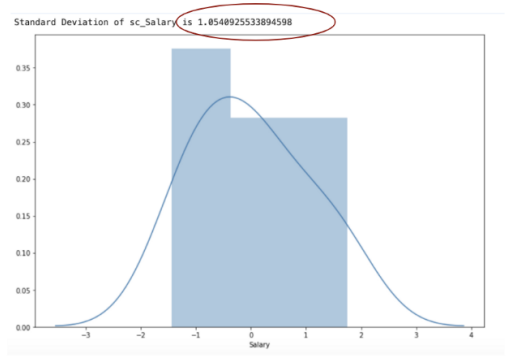
EXAMPLE

- From the above graphs, we can clearly notice that applying Max-Min Nomaralisation in our dataset has generated smaller standard deviations (Salary and Age) than using Standardisation method. It implies the data are more concentrated around the mean if we scale data using Max-Min Nomaralisation.
- As a result, if you have outliers in your feature (column), normalizing your data will scale most of the data to a small interval, which means all features will have the same scale but does not handle outliers well. Standardisation is more robust to outliers, and in many cases, it is preferable over Max-Min Normalisation.

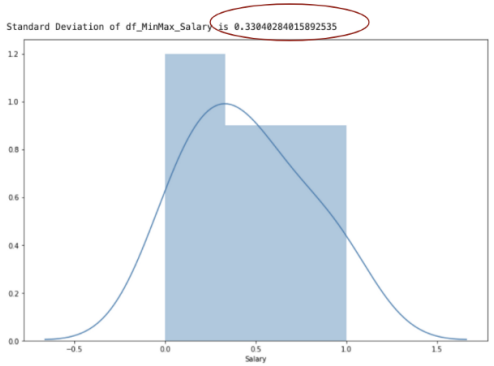
Column: Salary

Standard Deviation (Salary):
Max-Min Normalization (0.33) < Standardisation (1.05)

Standardisation



Max-Min Normalisation



WHAT ALGORITHMS NEED FEATURE SCALING

- Some machine learning models are fundamentally based on distance matrix, also known as the distance-based classifier, for example, K-Nearest-Neighbours, SVM, and Neural Network. Feature scaling is extremely essential to those models, especially when the range of the features is very different. Otherwise, features with a large range will have a large influence in computing the distance.
- Max-Min Normalisation typically allows us to transform the data with varying scales so that no specific dimension will dominate the statistics, and it does not require making a very strong assumption about the distribution of the data, such as k-nearest neighbours and artificial neural networks. However, Normalisation does not treat outliners very well. On the contrary, standardisation allows users to better handle the outliers and facilitate convergence for some computational algorithms like gradient descent. Therefore, we usually prefer standardisation over Min-Max Normalisation.

Algorithm(s)	Reason of applying feature scaling
1. K-Means	Use the Euclidean distance measure.
2. K-Nearest-Neighbours	Measure the distances between pairs of samples and these distances are influenced by the measurement units
3. Principal Component Analysis (PCA)	Try to get the feature with maximum variance
4. Artificial Neural Network	Apply Gradient Descent
5. Gradient Descent	Theta calculation becomes faster after feature scaling and the learning rate in the update equation of Stochastic Gradient Descent is the same for every parameter

NORMALIZATION VS STANDARDIZATION

	Normalization	Standardization
1.	Minimum and maximum value of features are used for scaling	Mean and standard deviation is used for scaling.
2.	It is used when features are of different scales.	It is used when we want to ensure zero mean and unit standard deviation.
3.	Scales values between [0, 1] or [-1, 1].	It is not bounded to a certain range.
4.	It is really affected by outliers.	It is much less affected by outliers.
5.	Scikit-Learn provides a transformer called MinMaxScaler for Normalization.	Scikit-Learn provides a transformer called StandardScaler for standardization.
6.	This transformation squishes the n-dimensional data into an n-dimensional unit hypercube.	It translates the data to the mean vector of original data to the origin and squishes or expands.
7.	It is useful when we don't know about the distribution	It is useful when the feature distribution is Normal or Gaussian.
8.	It is a often called as Scaling Normalization	It is a often called as Z-Score Normalization.

3.6 TESTING WITH NEW DATA

- Once we have pre-processed our data into a format that's ready to be used by our model, we need to split up our data into train and test sets. This is because our machine learning algorithm will use the data in the training set to learn what it needs to know. It will then make a prediction about the data in the test set, using what it has learned. We can then compare this prediction against the actual target variables in the test set in order to see how accurate our model is.
- We will do the train/test split in proportions. The larger portion of the data split will be the train set and the smaller portion will be the test set. This will help to ensure that we are using enough data to accurately train our model.
- In general, we carry out the train-test split with an 80:20 ratio, as per the Pareto principle. The Pareto principle states that "for many events, roughly 80% of the effects come from 20% of the causes." But if you have a large dataset, it really doesn't matter whether it's an 80:20 split or 90:10 or 60:40. (It can be better to use a smaller split set for the training set if our process is computationally intensive, but it might cause the problem of overfitting)

Data preparation plays a key role in earlier stages of machine learning and AI application development, as noted earlier. In an AI context, data preprocessing is used to improve the way data is cleansed, transformed and structured to improve the accuracy of a new model, while reducing the amount of compute required.

THANK YOU

khairulbazli@ump.edu.my