Chapter 2: Data preprocessing

outline

- Introduction to Data Quality Dimensions and Data Quality Assessments
  - ❖ Definition of Data Quality
  - ❖ Data Quality Dimensions
- ❑ Data pre-processing
  - ❖ Data Cleaning
    - ➢ Missing Data
    - ➢ Outliers
  - ❖ Data Transformation
    - ➢ Normalization
    - ➢ Encoding Categorical Features

Objectives

- Ddescribe importance of quality data for proper data use
- List data quality dimensions and data quality assessment ways
- Apply Python libraries to pre-process data.
- Apply data quality assurance techniques.
- Transform the data into form appropriate for Data Modeling.

Introduction to Data Quality Dimensions and Assessments

- Data is at most important in data science as it mainly works on data, and the quality of data is equally critical.
- Data quality dimensions are a notion that defines data quality using a set of data quality attributes.
- Data pre-processing is a crucial step in which data scientists select relevant data and address issues such as noise and redundancy

Introduction to Data Quality Dimensions and Assessments

- Data quality can be broadly defined as follows:
  - ✓ Fitness for use: Data must fit for its planned and intended uses.
- In order to satisfy the above-mentioned basic criteria's data must fulfills requirements to the following quality dimensions:
  - ✓ Completeness
  - ✓ Uniqueness
  - ✓ Timeliness
  - ✓ Validity
  - ✓ Accuracy
  - ✓ Consistency

Introduction to Data Quality Dimensions and Assessments

- Even though, there are multiple and advanced processes for assessing data quality, below is the generalized steps to assess data quality dimensions:

  Step1: Identify Data

  Step 2: Assess which data quality dimensions are/are not satisfied

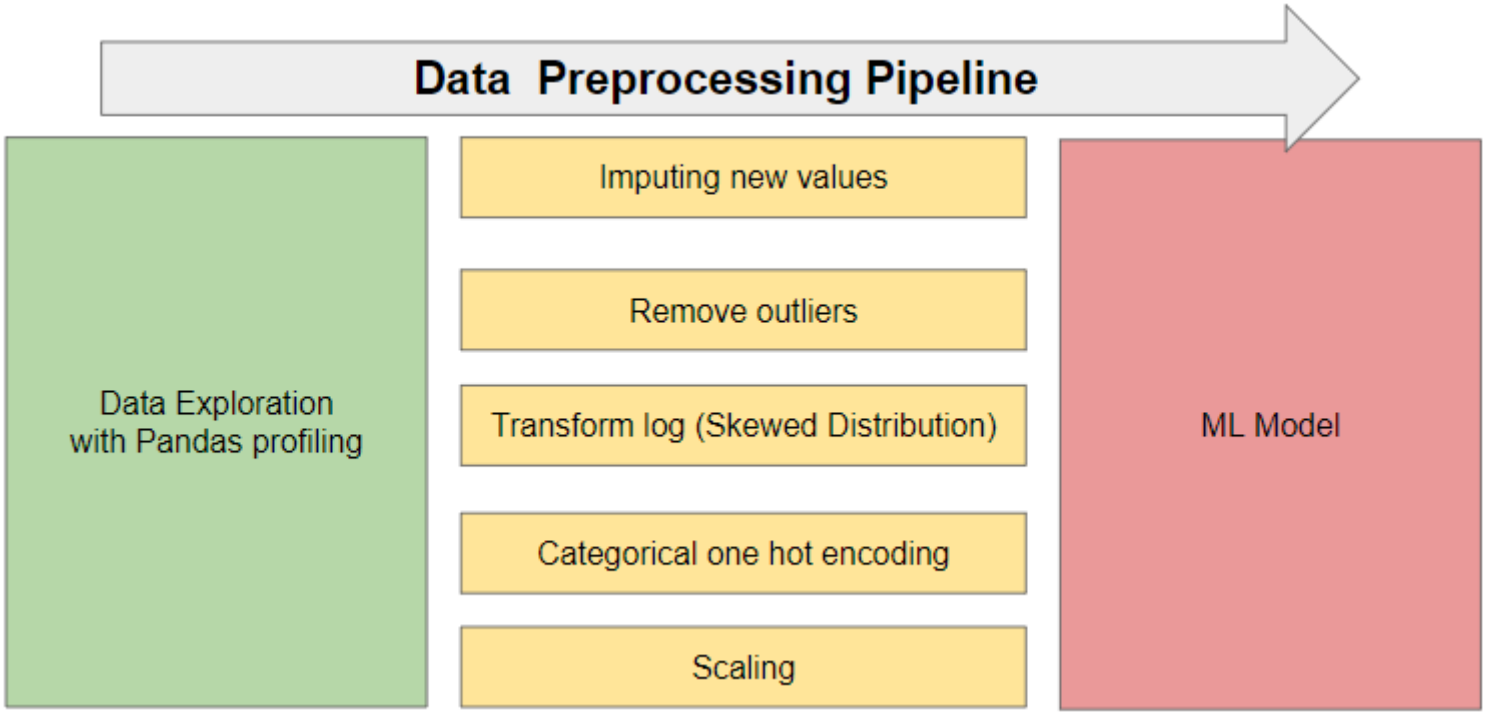
  Step 3: Define values or ranges representing good and bad quality

  Step 4: Review the results and determine if data quality is acceptable Step

  5: Take correctives measures

Data pre-processing

- Data Cleaning
  - Missing Data


Data Preprocessing Pipeline
Data Exploration
with Pandas profiling
Imputing new values
Remove outliers
Transform log (Skewed Distribution)
Categorical one hot encoding
Scaling
ML Model

Data pre-processing

- Data preprocessing is a technique that involves preparing the given raw data into suitable and understandable format.
- Real world data is often incomplete, inconsistent, and is likely to contain many errors.
- Such challenges are overcome through a set of tasks that are performed to **clean**, **integrate**, **transform**, and **reduce** the data before running any analysis

Data pre-processing

- It's crucial to remember that data preprocessing is typically the most time-consuming part of the data science lifecycle.
- And it should be addressed seriously because bad data leads to weak models, poor performance, and bad results.

Data Pre-processing|Data Cleaning

- Data in the real world is typically messy in the sense that:
  - ✓ it may be incomplete : e.g. missing data
  - ✓ noisy : e.g. random error or outlier values that deviate from the expected baseline.
  - ✓ or inconsistent : e.g. patient age 18 and admission service is neonatal intensive care unit.

Data Pre-processing|Data Cleaning

- The process of identifying, correcting, or eliminating observations that are outside the scope of the dataset is known as data cleaning.
- It refers to techniques to clean data by removing outliers and replacing missing values

Data Cleaning| Missing Data

- In real-world datasets, missing data is common, and it has a significant impact on the final analysis result, potentially making the conclusion inaccurate.
- Identifying the source of missing values is crucial, as it influences the choice the handling methods

Data Cleaning| Missing Data

- For example, the sources of missing data could be :
  - ✓The variable is measured, but the value is not recorded for unknown reasons.
  - ✓The variable is not measured during a certain period of time due to an identifiable reason.
  - ✓The variable is not measured because it is unrelated with the patient's condition and provides no clinical useful information to the physician

Missing Data | Dealing with Missing Data

- It is normally a reasonable rule of thumb to reject variables with an excessive amount of missing values (e.g. > 50%).
- However rejecting a variable can reduce predictive power and the capacity to discover statistically significant changes, as well as be a source of bias.
- For these reasons, variable selection needs to be tailored to the missing data mechanism.

Missing Data | Dealing with Missing Data

- Imputation can be done before and/or after variable selection.
- The general steps that should be followed for handling missing data are:
  - Identify patterns and reasons for missing data.
  - Analyze the proportion(percentage) of missing data .
  - Choose the best imputation method

Missing Data | Dealing with Missing Data

- The most widely used methods of handling missing data fall into three main categories:
  1. Deletion Methods
  2. Single Value Imputation Methods
  3. Model based Imputation Methods

Missing Data | Dealing with Missing Data

1. Deletion Methods

- The simplest way to deal with missing data is to discard the cases or observations that have missing values.
- There are two ways of doing this:
  - Complete case analysis (Listwise deletion): All the observations with at least one missing variable are discarded.
  - Available-Case Analysis (Pairwise deletion) : This method discards data only in the variables that are needed for a specific analysis. For example, if only 4 out of 20 variables are needed for a study, this method would only discard the missing observations of the 4 variables of interest

2. Single value imputation Method

- Missing values are filled by some type of "predicted" values.
- It ignores uncertainty and almost always underestimates the variance.
- The simplest imputation method is to substitute missing values by the **mean, mode** or the **median** of that variable.

3. Model based imputation methods (Reading assignment)

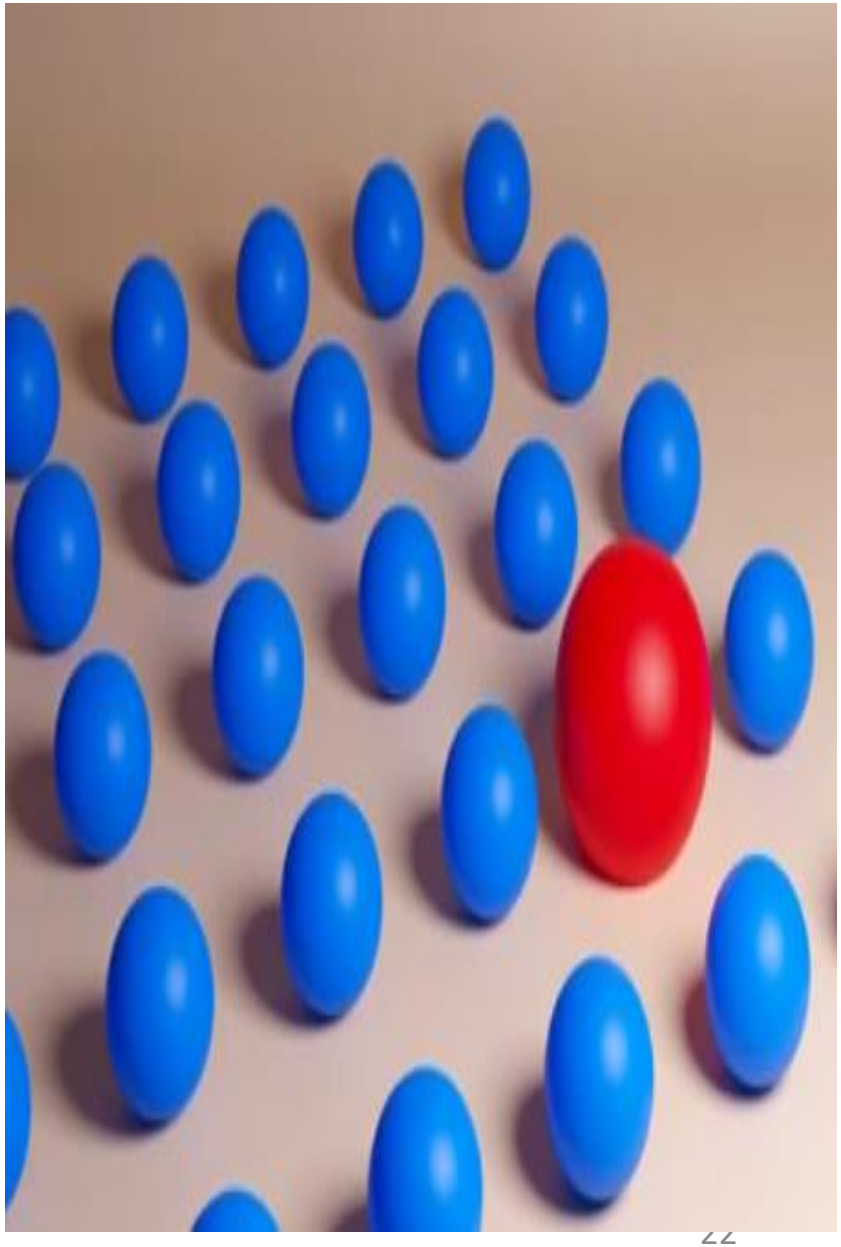
- In this method, a predictive model is created to estimate values that will substitute the missing data.
- The dataset is divided into training and test data.
- Several modeling methods can be used such as regression, logistic regression, Random forest, K-Nearest Neighbors

Data pre-processing

- Data Cleaning
  - Outliers

Data Cleaning| Outliers

- An outlier is a data point which is different from the remaining data.
- The main sources of outliers are:
  - ✓ Equipment malfunctions
  - ✓Human errors
  - ✓Anomalies arising from specific behaviors
  - ✓ Natural variation eg. within patients



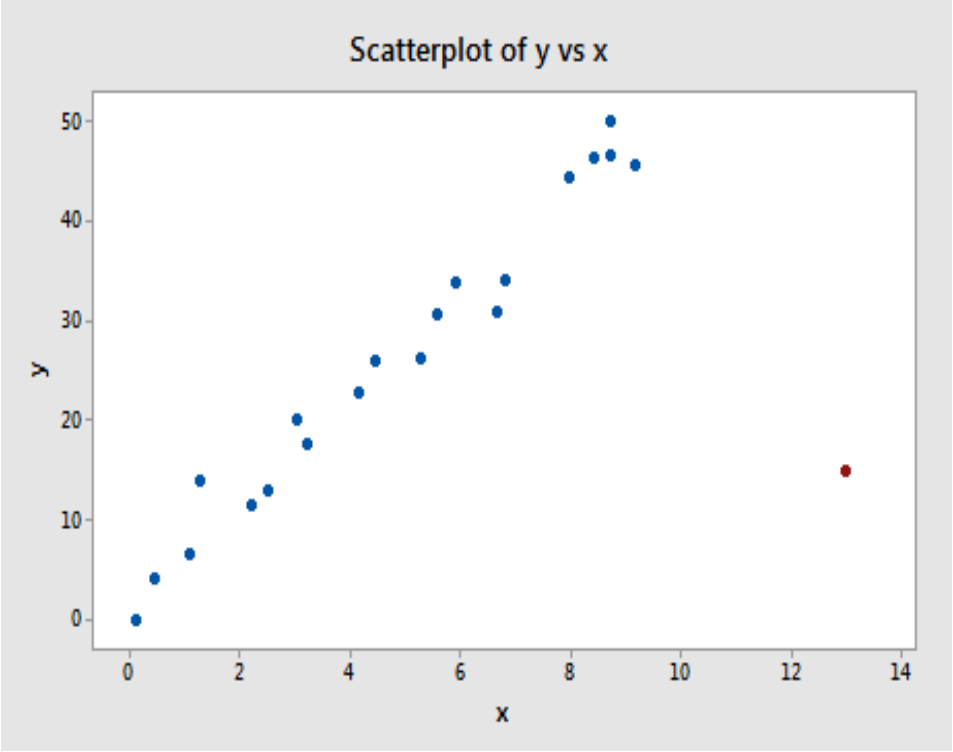
Data Cleaning| Outliers

- The negative effects of outliers can be summarized in:
  - ➢ Increase in error variance and reduction in statistical power.
  - ➢ Decrease in normality for the cases where the outliers are non-randomly distributed.
  - ➢ Model bias by corrupting the true relationship between exposure and outcome.

Finding Outliers | Using visualization tools

1. Scatter Plot

- It is a type of plot or mathematical diagram using Cartesian coordinates to display values for typically two variables for a set of data.



Finding Outliers | Using visualization tools

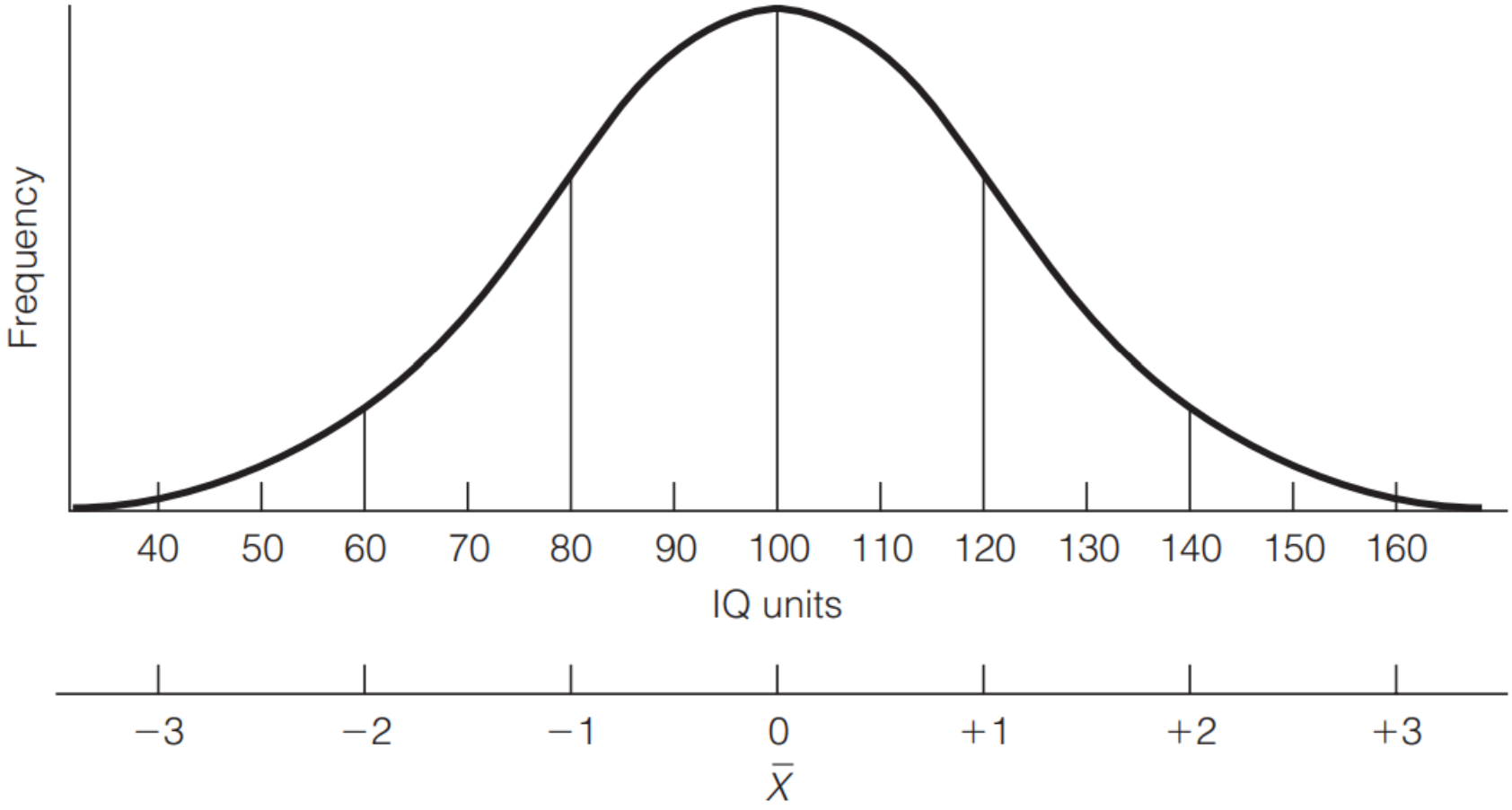
2. Z-Score (Reading assignment)

- The Z-value test computes the number of standard deviations by which the data varies from the mean.
- The Z-score find the distribution of data where mean is 0 and standard deviation is 1 i.e. normal distribution.
- While calculating the Z-score we rescale and center the data and look for data points which are too far from zero

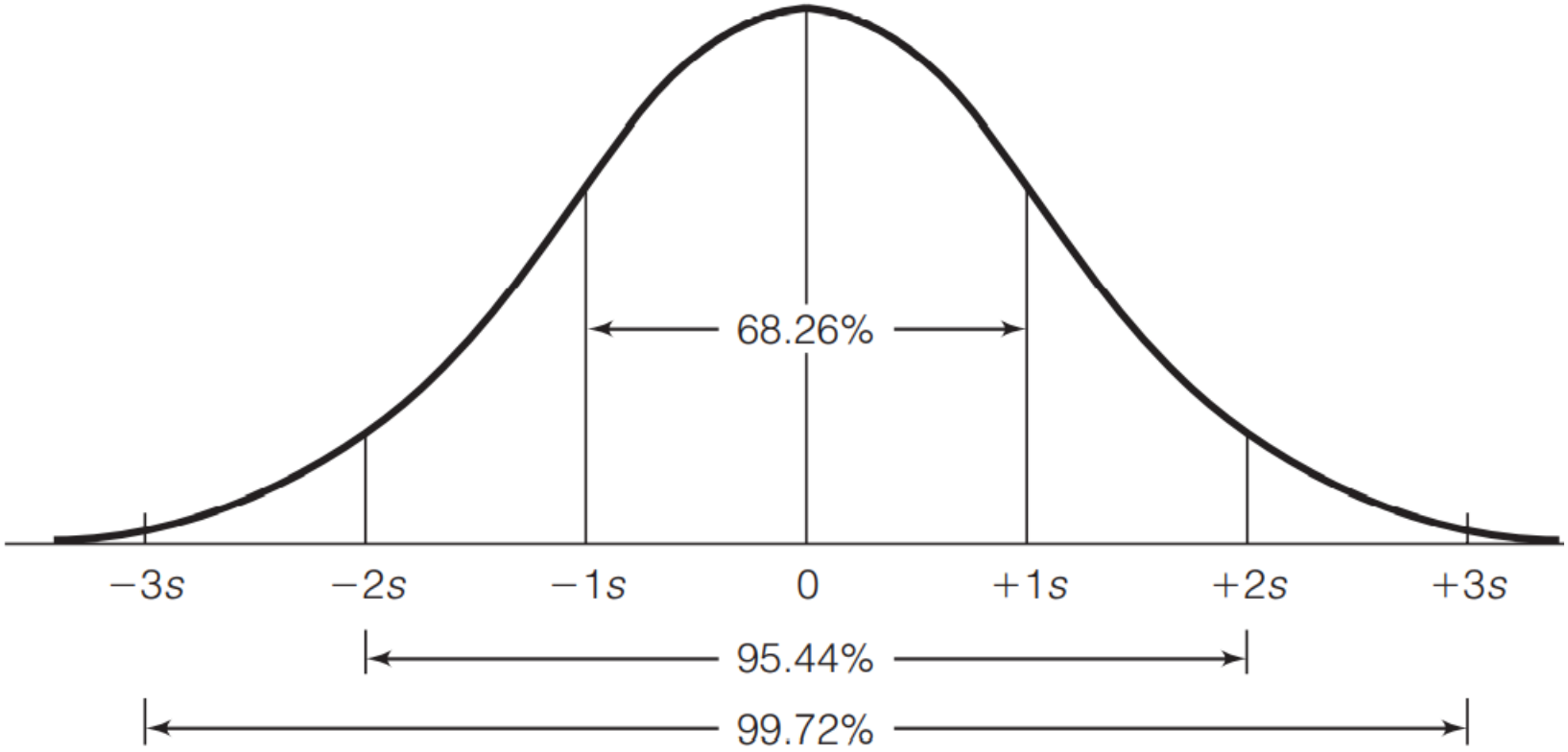
$$Z = \frac{X_i - \overline{X}}{s}$$

Xi=value of score
$\overline{X}$=mean
s=standard deviation

Z-score Vs Raw score



AREAS UNDER THE THEORETICAL NORMAL CURVE



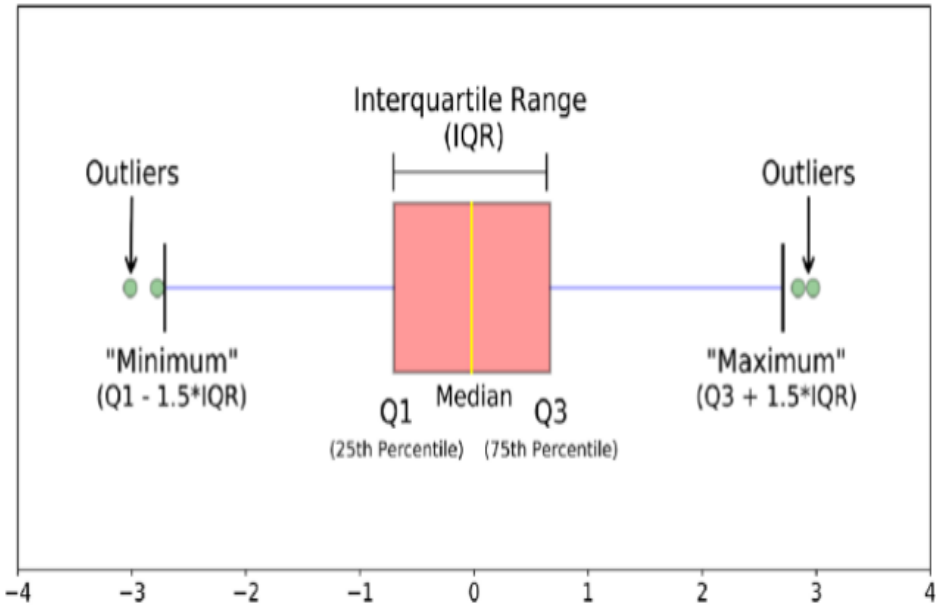
Finding Outliers | Using visualization tools

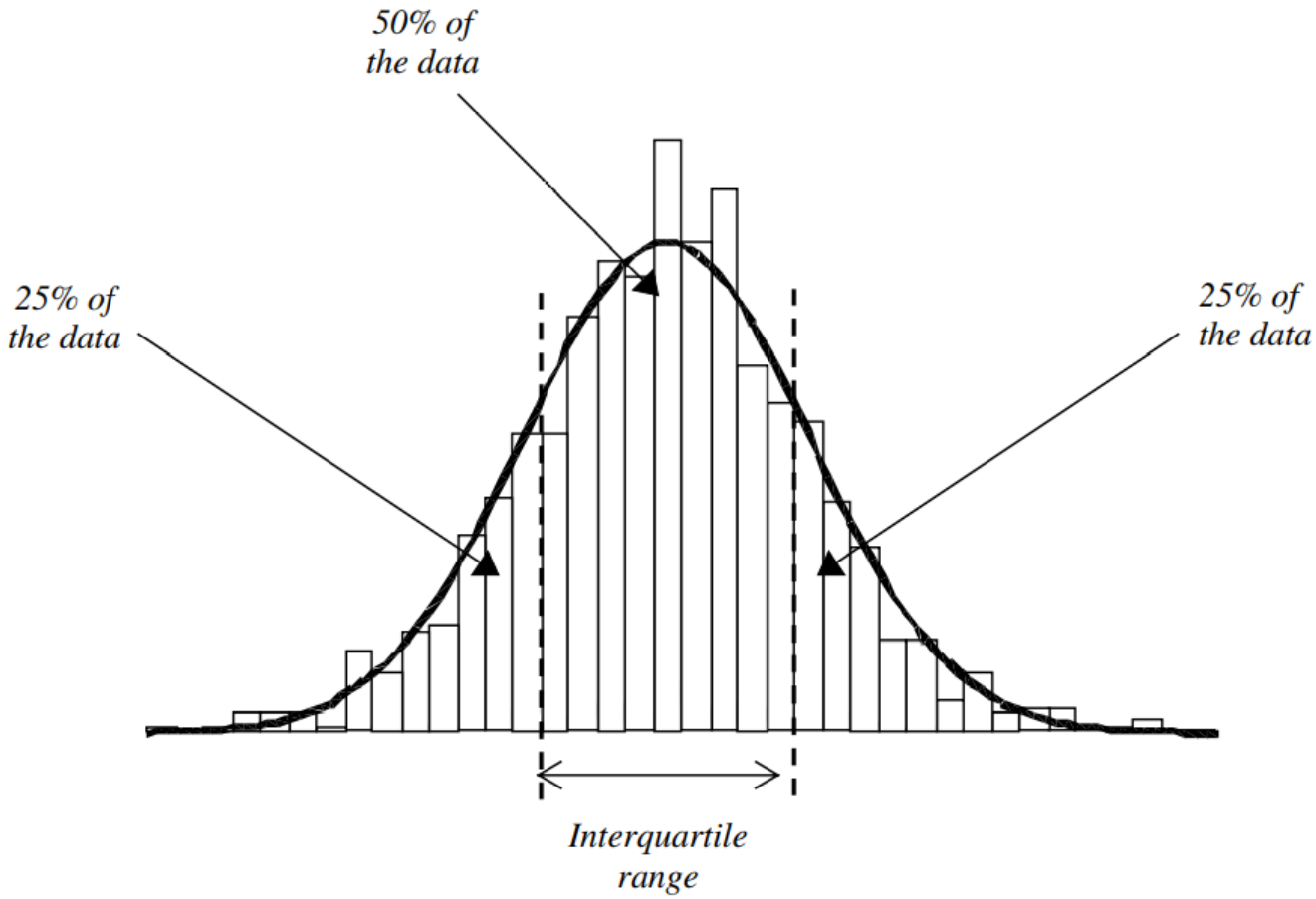
- These data points which are way too far from zero will be treated as the outliers.
- While calculating the Z-score we rescale and center the data and look for data points which are too far from zero.
- These data points which are way too far from zero will be treated as the outliers.

Finding Outliers | Using visualization tools

3. Box Plot

- It is a method for graphically representing groups of numerical data through their quartiles.
- Data should be sorted in ascending
- IQR=q3-q1




50% of
the data
25% of
the data
25% of
the data
Interquartile
range

Exercise: determine the outlier ranges using IQR

Location	PM_{10}	Order	
Edinburgh centre	0.4	1	
Glasgow centre	7.0	2	
Middlesbrough	12.2	3	
Nottingham centre	13.0	4	
Bristol centre	13.4	5	
Belfast centre	14.3	6	
Thurrock	**14.6**	**7**	**← first quartile is 14.6**
Cardiff centre	15.0	8	
Liverpool centre	15.2	9	
Birmingham east	15.5	10	
Hull centre	15.6	11	
Birmingham centre	15.7	12	
Wolverhampton centre	16.2	13	
Manchester Piccadilly	**16.6**	**14**	**← middle value is 16.6**
Leeds centre	18.0	15	
Leicester centre	18.5	16	
Port Talbot	20.1	17	
Swansea centre	23.0	18	
London Bexley	23.2	19	
Sutton roadside	24.8	20	
London Kensington	**24.9**	**21**	**← third quartile is 24.9**
London Brent	25.0	22	
London Bloomsbury	29.4	23	
Southampton centre	30.0	24	
London Hillingdon	30.7	25	
Camden kerbside	32.8	26	
Bury roadside	38.0	27	

Normalization

- Data normalization, also called feature scaling,
- is the process where the values for each numerical variable are scaled in order to range between a specified set of value
- Example: Converting the values to a common scale, such as 0 to 1 or -1 to 1
- Goal
  - To ensure large values in the dataset don’t influence the learning process and have a similar impact on the model’s learning process

Normalization

- It is used in machine learning:
  - To make model training less sensitive to the scale of feature and
  - allows our model to converge to better weights and, in turn, leads to a more accurate model.
- Normalization makes the features more consistent with each other, which allows the model to predict outputs more accurately.

Type of Normalization

Single feature scaling:

- Converts every value of a column into a number between 0 and 1.
- The new value is calculated as the current value divided by the max value of the column

- Min Max:
  - Converts every value of a column into a number between 0 and 1.
  - The new value is calculated as the *difference between the current value and the min value, divided by the range of the column value*

Z-score:

- Converts every value of a column into a number around 0.
- Typical values obtained by a z-score transformation range from -3 and 3.
- The new value is calculated as the difference between the current value and the average value, divided by the standard deviation.

Log Scaling:

- Log scaling involves the conversion of a column to the logarithmic scale

Clipping:

- It involves the capping of all values below or above a certain value. Clipping is useful when a column contains some outliers.

Encoding categorical features

- Many times the data we use may not have the features values in a continuous form,
- But instead the forms of categories with text labels,
  - For example, a person could have features ["male", "female"].
- To get this data processed by the machine learning model, it is necessary for converting these categorical features into a machine-understandable form.

- There are two functions available through which we can encode our categorical features:

OrdinalEncoder:

  - This estimator transforms each categorical feature to one new feature of integers (0 to n_catergories-1)

OneHotEncode:

  - This encoder function transforms each categorical feature with n_categories possible values into n_categories binary features, with one of them 1, and all others 0