Exploratory Data Analysis

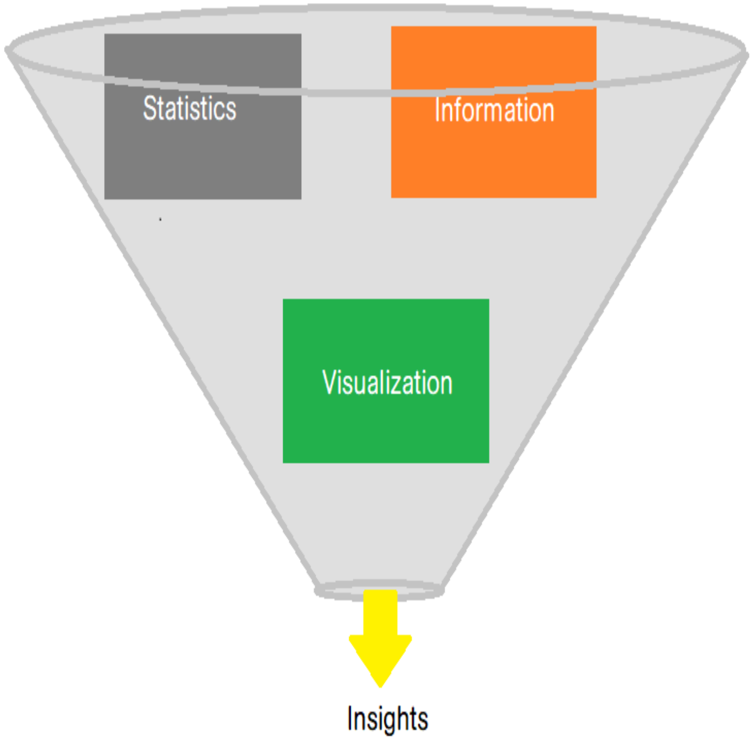
(EDA)

Agenda

Introduction to EDA
Describe Data (Descriptive Analytics)
Data Pre-processing
Data Visualization
Data Preparation

Introduction to EDA

EDA is an approach to analyze data using both non-visual and visual techniques
Generation of insights is a "Creative" process , however there is a structured approach which is followed
Involves thorough analysis of data to understand the current business situation
EDA objective is to extract "Gold" from the "data mine" based on domain understanding



Describe Data

Know the Problem Statement or the Business Objective

Load and view the given data

Check the relevance of the data against the objective or goal to be achieved

- Scope of the data
- Time relevance of the data
- Quantum of data
- Features of the data

Understand each feature in the data with help of Data Dictionary

Know the central tendency and data distribution of each feature

Data Pre-processing

Practical data set generally has lot of “noise” and/or “undesired” data points which might impact the outcome, hence pre-processing is an important step

As these “noise” elements are so well amalgamated with the complete data set, cleansing process is more governed by the data scientist ability

These noise elements are in the form of

- Bad values
- Anomalies (Not valid or not adhering to business rules)
- Missing values
- Not Useful Data

How to detect ‘Bad Values’?

Numeric Fields:

- Check if datatype of every numeric feature/column is valid
  - ‘Salary Amount’ field is expected to be numeric with data type as float
  - But if the data type appears as ‘Object’ there is bad data which has to be cleaned
- Check range of values
  - ‘Age’ field with a minimum value of 0 and maximum as 60

Categorical Fields:

- Check categorical levels of each feature/column with “Object” datatype
- Level may have some special characters like “?” , “-”, “!” or invalid categories which does not represent the feature

How to detect ‘Anomalies’

Understanding the meaning and relevance of each feature and business knowledge plays an important role in identifying other anomalies in data

- In finance, business expects financial ratios to be within range
- For a loan data some features like,
  - Fixed Obligation to Income Ratio (‘FOIR’) is expected to be in a range of 0-1
  - Net Loan to Value Ratio (“Net_LTV”) from 0-100 etc.

How to detect ‘Not Useful Data’

- Duplicate records or rows
  - If retained, may result in misleading algorithmic evaluations, hence recommended to be removed
  - Same data appearing for all features in multiple records
- A feature or column that has a single value in all the records
  - Zero-variance predictors as their value remains same across all the records
- Feature or column with more than 25-30% missing values

Python Example

Pre-processing

Data Visualization

Visualization is a technique for creating diagrams, images or animations to communicate a message

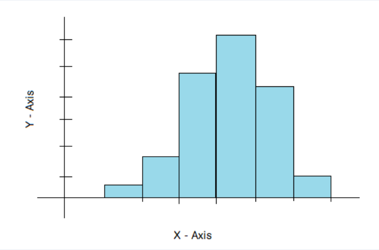
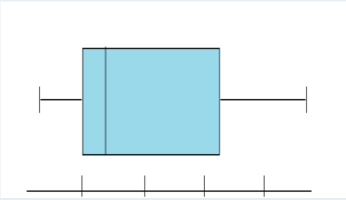
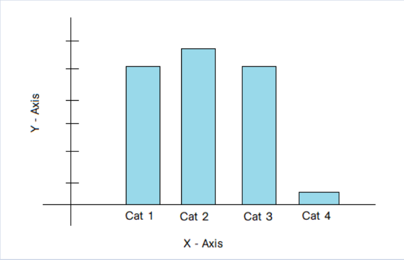
Usage of charts or graphs to visualize huge amounts of complex data is easier than poring over spreadsheets or reports

Data Analysis using Visualization includes:

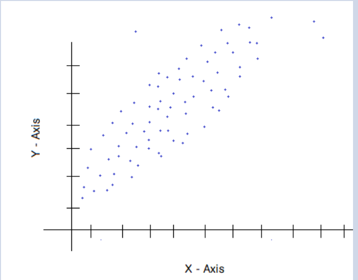
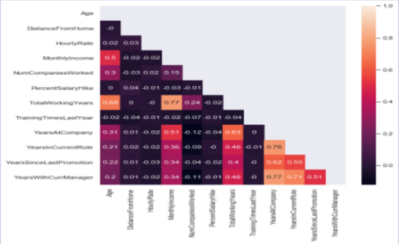
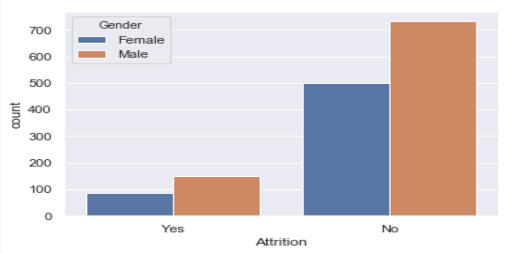
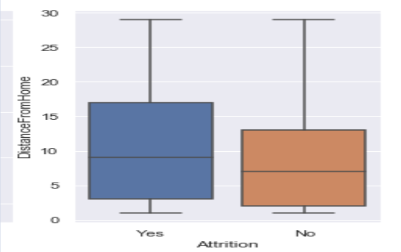
- Univariate Analysis
- Bivariate Analysis
- Multivariate Analysis

Key for this analysis is generating insights/inferences aligned with the business problem

Univariate Analysis

Data Summary			Data Visualization (Univariate Analysis)
Data Types	**Central Tendency**	**Distribution**	**Graphs**
Numeric Variables	Mean Median Mode 5 Number Summary	Standard Deviation Range IQR	Histogram  Boxplot 
Categorical Variables	Mode	Frequency of the levels	Countplot 

Bivariate Analysis

Data Types	Data Visualization (Bivariate Analysis)
Numeric & Numeric Variable OR All Numeric Variables	Pairplot/ Scatterplot  Correlation Plot/ Heatmap 
Categorical & Categorical Variable	Countplot with hue(diff colour) 
Categorical & Numeric Variable	Boxplot (x as Categorical & y as Numeric) 

Python Example

Visualization

Data Preparation

Scaling
Transformation
Outliers Detection & Treatment
Data Encoding

Scaling

Why do we need to do it?

- Data set has features with different "weights"
- In "Distance" based algorithms it is recommended to transform the features so that all features are in same "scale"

Most commonly used scaling techniques are

- Z-Score
  - $Z = (X - \mu) / \sigma$
  - Scaled data will have mean tending to 0 and standard deviation tending to 1
  - Used in weight based techniques (PCA, Neural Network etc.)
- Min-Max
  - $(X - X_{min})/(X_{max} - X_{min})$
  - Scaled data will range between 0 and 1
  - Used in distance based techniques (Clustering, KNN etc.)

Transformation

Why do we need to do it?

- When a variable is on larger scale, we can transform it to a lower scale using Log Transformation
- To deal with Skewness

Most commonly used transformation techniques are

- For Positively Skewed features Log, Exponential, and Square Root Transformations are used
- For Negatively Skewed features Log, Cube Root, and Square Transformations are used

If data is transformed, results are obtained in terms of transformed data

Hence, care should be taken to reverse the same to conclude the results

Outliers

Outliers are data points that have a value significantly different than the rest of the values in the feature

It might be a valid data point or may have been caused due to error

- If we consider height of student of class 7, most of them may be in a range of 4.8 Feet to 5.4 Feet. However, there maybe 1 or 2 students who are around 4 Feet or around 6 feet
- During data entry extra zeros have been added to an amount field making it different from others
- Most of the data provided for Fraud detection will have very few records where fraud has occurred. There are high chances that these records get identified as outliers

Hence, it is important to analyze the outliers before deciding on treatment

Outliers

Outlier treatment is not mandatory

- There are algorithms in machine learning that are not very sensitive to outliers
- We can choose the relevant algorithms to work on the data

When essential, outlier treatments are done with following considerations:

- Treatment of outliers should not change the meaning of the data to a great extent which in turn reflects current business situation
- Business or domain knowledge to be taken into account to decide on the treatment

Basic techniques to detect outliers

- Z Score
- Boxplot

Outlier Detection

ZScore

First scale the variables by applying ZScore

All records with score greater than 3 and less than -3 are considered as outliers

For a feature, if we assume a normal distribution, 99.7% of the data points are within ± 3 σ value, anything beyond it is outlier which are very few data points

Boxplot

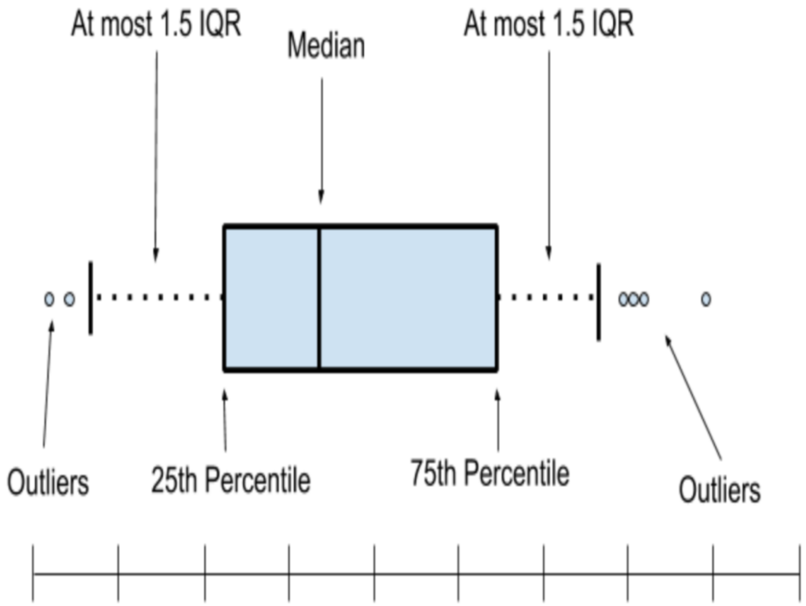
Any data point more than Q3+1.5*IQR or less than Q1-1.5*IQR is taken as an outlier

50% of data points are within ± 0.5 IQR of the median

In a normal distribution 68% are with ± 1σ

So IQR (50%) is slightly less than ± 1σ (68%)

In order to correspond ± 3 σ range, ±1.5IQR (i.e. 3* ± 0.5 IQR) is taken as range to identify outliers



Data Encoding

"Object" and/or "Categorical" type of variables which have a values as "Label" like Male/Female are not allowed in the models, hence the same needs to be "encoded" in numeric format

There are primarily two types of encoding:

- One Hot Encoding
  - Each category is converted to a column having only boolean values
  - Recommended if the there are less number of categorical levels within the field (less than 25)
- Label Encoding
  - When there are too many levels/categories in a variable in a dataset
  - If the labels are "Ordinal" like "Satisfaction Score"

Python Example

Data Transformation

Case Study

Credit Card Default