IIT Guwahati

Advanced Certification Programme in Data Science Business Analytics

Week 17
Exploratory Data Analysis (EDA)

Topics Covered

- Exploratory Data Analysis (EDA) on Used Car Prices Dataset
- EDA Best Practices: Unveiling Data Insights
- Q & A

Async Recap

1. **Discover Patterns and Anomalies**
   Reveal trends and inconsistencies using summary statistics, correlation matrices, and visual tools like scatter plots and histograms.

2. **Clean and Prepare Data**
   Improve data quality by handling missing values, removing duplicates, and detecting outliers using IQR and Z-score methods.

3. **Understand Data Types**
   Recognise numerical, categorical, text, and time-series data to select appropriate analysis methods and visualisation formats.

4. **Apply Visualisation Techniques**
   Use charts such as boxplots, bar charts, and heatmaps to interpret distributions, relationships, and data skewness.

5. **Use Python Tools Effectively**
   Utilise Pandas, Seaborn, and SciPy to explore, clean, and visualise datasets efficiently for better analysis and modelling.

Exploratory Data Analysis (EDA) on Used Car Prices Dataset

Introduction to EDA

Data Insight Discovery

- Identifies patterns anomalies and relationships
- Guides preprocessing feature engineering model building
- Enables EDA using Pandas, NumPy, Seaborn, Matplotlib

Loading the Used Car Prices Dataset

Read And Preview The Dataset Using Pandas

```
import pandas as pd
# Load the dataset
df =
pd.read_csv("used_cars.csv")
# Display the first few rows
df.head()
```

- Import pandas
- Load dataset
- Preview top rows

Basic Dataset Information

Data Types and Missing Values

```
# Check dataset
information
df.info()
# Get the shape of the
dataset
df.shape
```

The dataset contains the following information:

- Number of rows and columns
- Data types of features
- Missing values

Summary Statistics Using describe()

Identifying Data Trends and Distributions

```
# Summary of numerical columns
df.describe()
# Summary of categorical columns
df.describe(include=['O'])
```

Numerical columns

- Mean, standard deviation, min, max, percentiles

Categorical columns

- Unique values, most frequent values

Handling Missing Values

Dealing With Incomplete Data

```
# Check for missing values
df.isnull().sum()
# Handling missing values
(Example: Fill Mileage with median)
df['Mileage'].fillna(df['Mileage'].medi
an(), inplace=True)
```

How to handle missing data?

- Identify missing data points
- Explore techniques for handling incomplete data

Analysing Categorical Variables

Understanding Category Distribution

```
# Frequency count of
categorical features
print(df['Fuel_Type'].value_c
ounts())
print(df['Transmission'].value
_counts())
```

- Calculate the distribution of categorical columns
- Identify dominant categories

Grouping Data using groupby()

Summarising by Category

```
# Average price by fuel type
df.groupby('Fuel_Type')['Pri
ce'].mean()
# Price range by
transmission type
df.groupby('Transmission')['
Price'].describe()
```

- Summarise data using groupby()
- Find average price per fuel type
- Find price range by transmission type
- Explore price differences between groups

Correlation Analysis

Finding Feature Relationships

```
# Correlation matrix
df.corr()
# Visualizing correlations
import seaborn as sns
import matplotlib.pyplot as plt
sns.heatmap(df.corr(), annot=True,
cmap='coolwarm')
plt.show()
```

- Identify relationships between numerical variables
- Visualise correlations with heatmaps

Visualising Data Distribution

Exploring the Spread of Numerical Data

```
import matplotlib.pyplot as plt
import seaborn as sns
# Histogram of price distribution
sns.histplot(df['Price'], bins=30,
kde=True)
plt.show()
```

How to visualise data distribution?

- Explore distribution of numerical variables
- Identify skew outliers

Visualising Price Variations by Fuel Type

Using Boxplots for Outlier Analysis

```
# Boxplot of price by fuel
type
sns.boxplot(x='Fuel_Type',
y='Price', data=df)
plt.show()
```

- Use boxplot tool for identifying outliers
- Illustrate the distribution of prices across different fuel types

Enhancing Data with Feature Engineering

Creating and Combining Features for Improved Analysis

Creating new features

```
# Creating a new feature - Car Age
df['Car_Age'] = 2025 - df['Year']
```

Combining features

```
# Creating a new feature - Price per KM
df['Price_per_KM'] = df['Price'] /
df['Mileage']
```

Enhancing Data with Feature Engineering

Binning and Encoding For Enhanced Performance

Binning numerical data

```
# Binning car age into categories
df['Car_Age_Group'] =
pd.cut(df['Car_Age'], bins=[0, 5, 10, 15,
20, 100], labels=['0-5', '5-10', '10-15', '15-
20', '20+'])
```

Encoding categorical features

```
# One-hot encoding categorical variables
df = pd.get_dummies(df,
columns=['Fuel_Type', 'Transmission'],
drop_first=True)
```

- Derive new features from existing data
- Convert non-numerical data into a machine-readable format
- Generate features that provide valuable insights for modelling

EDA Best Practices: Unveiling Data Insights

Establishing Clear Objectives

Guiding Your Exploratory Data Analysis

- **Set goals:**
  Establish clear and measurable objectives for your analysis
- **Define scope**:
  Determine the boundaries and focus areas of your EDA
- **Ask questions:**
  Formulate specific questions to guide your data exploration

Data Collection and Preparation

Essential Steps for Effective Analysis

- Gather relevant data from reliable sources
- Clean and transform data for consistency
- Address missing values and errors early
- Identify dependable sources that align with your research goals
- Correct errors, manage missing data and ensure data accuracy

Univariate Analysis

Explore Individual Variables To Understand Their Behavior

- **Descriptive Stats:** Calculate mean and median for central tendency
- **Distributions:** Use histograms and box plots to identify patterns
- **Individual Variables:** Assess range, outliers, and other characteristics

Bivariate Analysis

Exploring Relationships Between Two Variables

- **Scatter plots:** Detect patterns and clusters between variables
- **Correlation**: Measure strength and direction of relationships
- **Trend Identification:** Uncover dependencies between variable pairs

Visualisation Techniques

Present Your Data Clearly Using the Right Charts and Labels

- **Chart selection:** Match chart types to data and insights
- **Clear labels**: Use descriptive labels and titles
- **Effective communication:** Deliver insights clearly and concisely

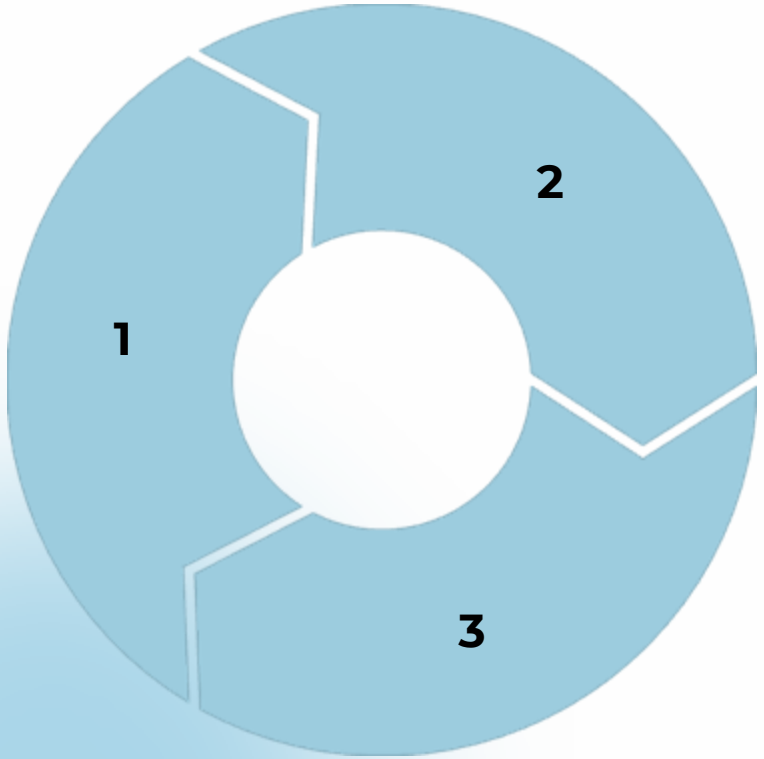
Managing Data Imperfections

Strategies for Missing Data and Outlier Handling

Handle outliers	Description
Missing value handling	Select a method for imputation, removal or other appropriate handling of missing values
Outlier management	Identify outliers and apply suitable techniques to manage or transform them
Documentation	Maintain a detailed log of all data cleaning and transformation steps taken

Translating Insights into Action

Summarising Findings and Planning Future Steps



1. Summarise key findings from EDA
2. Identify actionable insights for stakeholders
3. Outline further analysis or modeling steps

Q & A

Thank you