atomcamp

Project: Exploratory Data Analysis (EDA) on Food Service Data

Introduction

The objective of this Project is to analyze a food service dataset to gain insights into operational efficiency and food waste management. The dataset consists of variables such as the number of meals served, kitchen staff, environmental conditions (temperature and humidity), and food waste. Our goal is to explore this data, clean it, visualize key patterns, and derive actionable insights to optimize operations.

Link to dataset: Food Data

Dataset Overview

The dataset contains the following columns:

1. **ID**: A unique identifier for each record.
2. **date**: The date of the observation.
3. **meals_served**: The number of meals served on that day.
4. **kitchen_staff**: The number of kitchen staff working on that day.
5. **temperature_C**: The temperature (in Celsius) on the recorded day.
6. **humidity_percent**: The humidity percentage on the recorded day.
7. **day_of_week**: The day of the week as a numeric value (0 = Sunday, 1 = Monday, etc.).
8. **special_event**: A binary variable indicating whether a special event occurred (1 = event, 0 = no event).

9. **past_waste_kg**: The amount of food waste in kilograms from previous days.

10. **staff_experience**: The experience level of the kitchen staff (e.g., "Beginner", "Intermediate").

11. **waste_category**: The category of food waste (e.g., "dairy", "meat").

Exploratory Data Analysis (EDA) Process

The EDA process involves several steps to clean and analyze the data:

---

1. Data Cleaning

Before any meaningful analysis can take place, we need to clean the data. The following tasks will be performed:

1.1. Check for Missing Values

We will inspect the dataset for missing values in each column. Missing data is common and needs to be either imputed (filled in) or removed depending on the importance of the column and the number of missing values.

1.2. Check for Duplicate Rows

We'll check for any duplicate entries in the dataset. Duplicate rows can skew the analysis and should be removed.

1.3. Handle Categorical Data

For categorical columns, we will check for missing values and decide on how to handle them, such as by using the most frequent category or removing rows with missing values.

1.4. Correct Data Types

Ensure the date column is in the correct date format. This will help with time-based analysis (e.g., seasonal patterns).

---

atomcamp

2. Exploratory Data Analysis (EDA)

2.1. Summary Statistics

Start by computing the summary statistics for the numerical columns:

- **Mean**, **median**, **standard deviation**, **minimum**, and **maximum** values.
- This will help identify the central tendency and spread of the data.

2.2. Visualizing Distributions

We'll use several types of visualizations to explore the data:

- **Histograms**: To understand the distribution of numerical features like meals_served, temperature_C, humidity_percent, and past_waste_kg.
- **Boxplots**: To detect outliers and understand the spread of the data.
- **Bar plots**: To visualize categorical variables like staff_experience and waste_category.

2.3. Correlation Analysis

By calculating the **correlation matrix** between the numerical variables, we can identify potential relationships between features. For example:

- Is there a correlation between the number of meals served and the amount of food waste?
- Does temperature or humidity influence food waste?

---

3. Hypothesis Testing

3.1. Impact of Kitchen Staff on Food Waste

We will test whether the number of kitchen staff affects the amount of food waste. One hypothesis could be:

- **Null hypothesis (H0)**: There is no relationship between the number of kitchen staff and food waste.

- **Alternative hypothesis (H1)**: The number of kitchen staff significantly affects food waste.

We will use a **t-test** or **ANOVA** to compare the mean food waste across different groups of kitchen staff (e.g., low, medium, high staff levels).

3.2. Special Events and Food Waste

We will test whether food waste increases during special events:

- **Null hypothesis (H0)**: There is no difference in food waste between special event days and non-special event days.

- **Alternative hypothesis (H1)**: Food waste is higher on special event days.

We will perform a **t-test** comparing the average food waste on days with and without special events.

---

4. Key Insights and Recommendations

Based on the data exploration and analysis, we will derive insights and provide recommendations, including:

- **Staffing Optimization**: If there is a significant relationship between staff numbers and food waste, suggest optimal staffing levels to minimize waste.

- **Environmental Factors**: If temperature or humidity affects food waste, recommend strategies to adjust food preparation based on weather conditions.

- **Event Management**: If special events lead to higher food waste, suggest strategies to better manage food during these times (e.g., pre-planning portion sizes or reducing food waste through donation).

---

atomcamp

5. Data Visualization and Reporting

The final part of the Project involves creating visualizations that make the insights easy to understand. These could include:

- **Histograms and Box Plots** to visualize the distributions of meals_served, temperature_C, humidity_percent, and past_waste_kg.
- **Correlation Heatmap** to show relationships between numeric variables.
- **Bar Plots** comparing food waste across waste_category and staff_experience.

Deliverables

Each student/team is required to the following in the **github submission link:**

1. Project Report (PDF or Word format)

Your report should be clear, well-organized, and include the following sections:

1. **Introduction**
    - Brief overview of the objective of the project
    - Summary of the dataset and key variables
2. **Data Cleaning**
    - Description of steps taken to handle missing values, duplicates, and data types
    - How categorical data was treated
3. **Exploratory Data Analysis (EDA)**
    - Summary statistics
    - Visualizations (e.g., histograms, box plots, bar charts)
    - Key patterns or trends observed

4. **Correlation Analysis**
    - Heatmap or table showing relationships between numeric variables
    - Interpretation of key correlations
5. **Hypothesis Testing**
    - Clearly stated hypotheses
    - Statistical tests used (e.g., t-test, ANOVA)
    - Results and conclusions drawn
6. **Key Insights and Recommendations**
    - Operational or strategic insights based on your analysis
    - Recommendations for food waste reduction, staffing, or event management
7. **Conclusion**
    - Brief summary of findings
    - Limitations and suggestions for further analysis
8. **Appendix (if needed)**
    - Additional charts, code snippets, or supporting data

2. Jupyter Notebook (or Python script)

Containing all code used for data cleaning, analysis, and visualization. It should be clean, commented, and easy to follow.

Homework Submission Requirements

atomcamp

1. **GitHub link:** The repository should be organized in a clear and logical way to allow others to easily navigate through the files and understand the workflow. Here is an ideal structure for the repository

```
Food-Service-EDA/
|
├── data/
|   └── Food_data.csv          # The original dataset
|
├── notebooks/
|   └── Food_Service_EDA.ipynb # Jupyter notebook for the EDA process
|
├── reports/
|   └── Final_Report.pdf       # A well-written PDF report summarizing the findings
|
├── .gitignore                 # Git ignore file (to exclude unnecessary files)
|
├── README.md                  # Project documentation and instructions
|
└── requirements.txt           # Python dependencies for the project
```

atomcamp

Relevant Python LeetCode-style Questions for EDA Practice

1. Find the Day with Maximum Waste

Problem:
You are given a list of daily food waste amounts. Write a function to return the index (or date) with the maximum waste.

```
def max_waste_day(waste_list: List[int]) -> int:

    # Example: waste_list = [10, 15, 8, 22, 5]

    # Output: 3 (index of maximum waste value 22)

    pass
```

2. Group Days by Staff Experience

Problem:
Given a list of records with each record containing a day and staff experience level (Beginner, Intermediate, Expert), group and count how many days fall into each category.

```
def group_by_experience(records: List[Tuple[str, str]]) -> Dict[str, int]:

    # Example: [("2023-06-01", "Beginner"), ("2023-06-02", "Intermediate")]

    # Output: {'Beginner': 1, 'Intermediate': 1}

    pass
```

3. Temperature Outlier Detection

Problem:
Given a list of daily temperatures, identify all days where the temperature is more than 2 standard deviations away from the mean.

```
def detect_outliers(temps: List[float]) -> List[int]:

    # Return indices of outlier temperatures

    pass
```

4. Correlation Approximation

Problem:
Given two lists meals_served and food_waste, write a function to compute the **Pearson correlation coefficient** between them.

```
def correlation(x: List[float], y: List[float]) -> float:

    # Use the formula for Pearson correlation

    pass
```

5. Average Waste on Event vs. Non-Event Days

Problem:
Given two lists — one with food waste values, and another with binary flags for special events — compute the average food waste on event days and non-event days.

```
def compare_event_waste(waste: List[float], events: List[int]) -> Tuple[float, float]:

    # Output: (avg_event_day_waste, avg_non_event_day_waste)

    pass
```

6. Categorical Encoding

Problem:
Given a list of staff experience levels (Beginner, Intermediate, Expert), encode them as integers (e.g., Beginner = 0, Intermediate = 1, Expert = 2).

```
def encode_experience(levels: List[str]) -> List[int]:

    pass
```