INT375

DATA SCIENCE TOOLBOX: PYTHON PROGRAMMING

PROJECT REPORT

(Project Semester January-April 2025)

# Candy Sales

Submitted by

Venkatesh

Registration No- 12306233

Programme and Section- B.Tech CSE- K23GD

Course Code- INT375

Under the Guidance of

Assistant Professor . Baljinder Kaur (27952)

Discipline of CSE/IT

Lovely School of Computer Science and Engineering

Lovely Professional University, Phagwara



DECLARATION

I, Venkatesh , student of B.Tech Computer Science Engineering under CSE/IT Discipline at, Lovely Professional University, Punjab, hereby declare that all the information furnished in this project report is based on my own intensive work and is genuine.

Date: 12/04/2025

Signature: Venkatesh

Registration No. 12306233

Name of the student: Venkatesh

CERTIFICATE

This is to certify that Venkatesh bearing Registration no. 12306233 has completed INT375 project titled, “Candy Sales” under my guidance and supervision. To the best of my knowledge, the present work is the result of his/her original development, effort and study.

Signature and Name of the Supervisor

Designation of the Supervisor

School of Computer Science and Engineering

Lovely Professional University Phagwara, Punjab.

Date: 12/04/2025

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to my project guide, Baljinder Kaur, for their valuable guidance and support throughout this project, “Analysis of Bank Churn”. I am thankful to the Department of Computer Science and Engineering, Lovely Professional University, for providing the necessary resources and environment. I also acknowledge the Government of India for making the dataset publicly available, enabling this research

Name: Venkatesh

Registration No. : 12306233

TABLE OF CONTENTS

1. Introduction

1. Source of Dataset

1. Exploratory Data Analysis (EDA) Process

1. Analysis of Data
2. Conclusion

1. Scope for Future Enhancement
2. Github / Linkedin Link

1. References

## 1. Introduction

In today’s highly competitive market, understanding consumer behavior and sales patterns is crucial for the success of any business. The confectionery industry, particularly candy sales, is a dynamic sector influenced by various factors such as seasonality, pricing strategies, promotional efforts, and consumer preferences. Analyzing sales data allows businesses to make informed decisions, optimize inventory, improve marketing strategies, and ultimately increase profitability.

This project focuses on the analysis of candy sales data over a specific period. The dataset contains information on different candy products, their sales figures, and other relevant attributes such as price, quantity sold, and dates of transactions. By leveraging data analytics techniques, we aim to uncover hidden patterns, identify top-performing products, and understand temporal sales trends.

The goal of this analysis is not only to interpret past performance but also to derive actionable insights that can support strategic planning for future growth. With the growing importance of data-driven decision-making, this project serves as a foundation for implementing more advanced analytics in the candy retail domain, such as predictive modeling and customer segmentation

2. SOURCE OF DATASET

The dataset used for this analysis is titled **Candy\_Sales.csv**, which contains structured sales data related to various candy products. The data includes detailed records of transactions such as product names, quantities sold, sales revenue, unit price, and dates of sale. This dataset provides a valuable snapshot of the candy sales performance over time.

**Possible Origins of the Dataset:**

* **Internal Business Records**: This dataset could originate from a company’s point-of-sale (POS) system, inventory logs, or retail management software. Such systems regularly track and store transaction-level data.
* **Retail Aggregators or Marketplaces**: Platforms like Amazon, Walmart, or third-party candy distributors often collect and export sales data for sellers.
* **Simulated or Public Datasets**: For academic, research, or training purposes, this dataset may also be artificially generated or shared as part of an open data initiative.

**Dataset Format:**

* File Type: CSV (Comma-Separated Values)
* Accessibility: Easy to load using spreadsheet software (Excel) or data analysis tools (like Python’s pandas library).
* Content: Each row typically represents a sales transaction or a product summary, depending on the format.

**Importance of Dataset Source:**

Understanding the source of the dataset is essential as it influences:

* **Data reliability and accuracy**
* **Level of detail and granularity**
* **Potential limitations or biases in the data**
* **Compliance with data privacy and usage policies**

In this project, the dataset serves as the foundation for conducting exploratory and statistical analyses to derive insights and patterns in candy sales behaviour.

### 3. EXPLORATORY DATA ANALYSIS(EDA) PROCESS

**1. Data Loading & Initial Inspection**

* Import the dataset using tools like **Python (Pandas)** or **Excel**.
* Display the first few rows to get a sense of the data structure (head() function in Python).
* Check the number of rows and columns, data types, and column names.

**2. Data Cleaning**

* **Missing Values**: Identify and handle missing or null values using methods such as imputation or removal.
* **Duplicates**: Check for duplicate records and remove them if necessary.
* **Incorrect Data Types**: Convert columns to appropriate data types (e.g., dates, numerical formats).
* **Outliers**: Detect unusually high or low values that might skew analysis.

**3. Descriptive Statistics**

* Use statistical summaries (describe() function) to understand central tendencies like:
* Mean, median, mode
* Minimum and maximum values
* Standard deviation
* Helps understand the range and distribution of key variables like sales quantity, price, and revenue.

**4. Univariate Analysis**

* Analyzing individual variables:
* Frequency of each candy product sold.
* Distribution of unit prices and quantities.
* Most common sales dates or months.
* Visualizations: **Histograms, Bar Charts, Box Plots**

**5. Bivariate/Multivariate Analysis**

* Explore relationships between two or more variables:
* Correlation between **price** and **quantity sold**
* Trend of **sales volume over time**
* Comparison of **product performance**
* Visualizations: **Scatter Plots, Heatmaps, Line Charts**

**6. Time Series Analysis (if applicable)**

* If the dataset contains date-related information:
* Analyze monthly/quarterly/seasonal sales trends.
* Identify peaks (e.g., holiday seasons) or dips in sales.

**7. Data Aggregation & Grouping**

* Grouping by product name, category, or time periods to summarize data.
* Total sales per candy type.
* Monthly sales performance.
* Average revenue per product.

### 4. ANALYSIS ON DATASET

**Objective 1: Sales by Region**

**i. Introduction**  
Analyzing sales by region helps businesses understand where their products are most and least popular. This information is critical for tailoring marketing strategies, optimizing logistics, and identifying potential markets for expansion. It also highlights regional preferences and demand variations, aiding in better resource allocation and inventory management.

## **ii. General Description** The dataset was grouped by region (e.g., North, South, East, West), and the total sales were calculated using the .sum() function. A bar chart was generated using Matplotlib to visualize the contribution of each region to overall sales. This visual comparison helps identify dominant regions and those requiring more business focus.

## **iii. Requirements**

## .groupby() and .sum() functions

## Matplotlib bar chart for visual representation

## **iv. Results** The bar chart revealed that certain regions, such as the North and West, contributed a larger share of total sales, indicating higher demand or better market penetration in those areas. In contrast, regions like the East and South showed comparatively lower sales, suggesting opportunities for marketing campaigns or improved distribution networks.

## **v. Visualization** *(Bar chart was generated as part of the script)*

## 

## **Objective 2: Total and Average Sales**

## **i. Introduction** Understanding total and average sales is fundamental for measuring business performance. These metrics provide a quick snapshot of the overall market size and per-transaction efficiency.

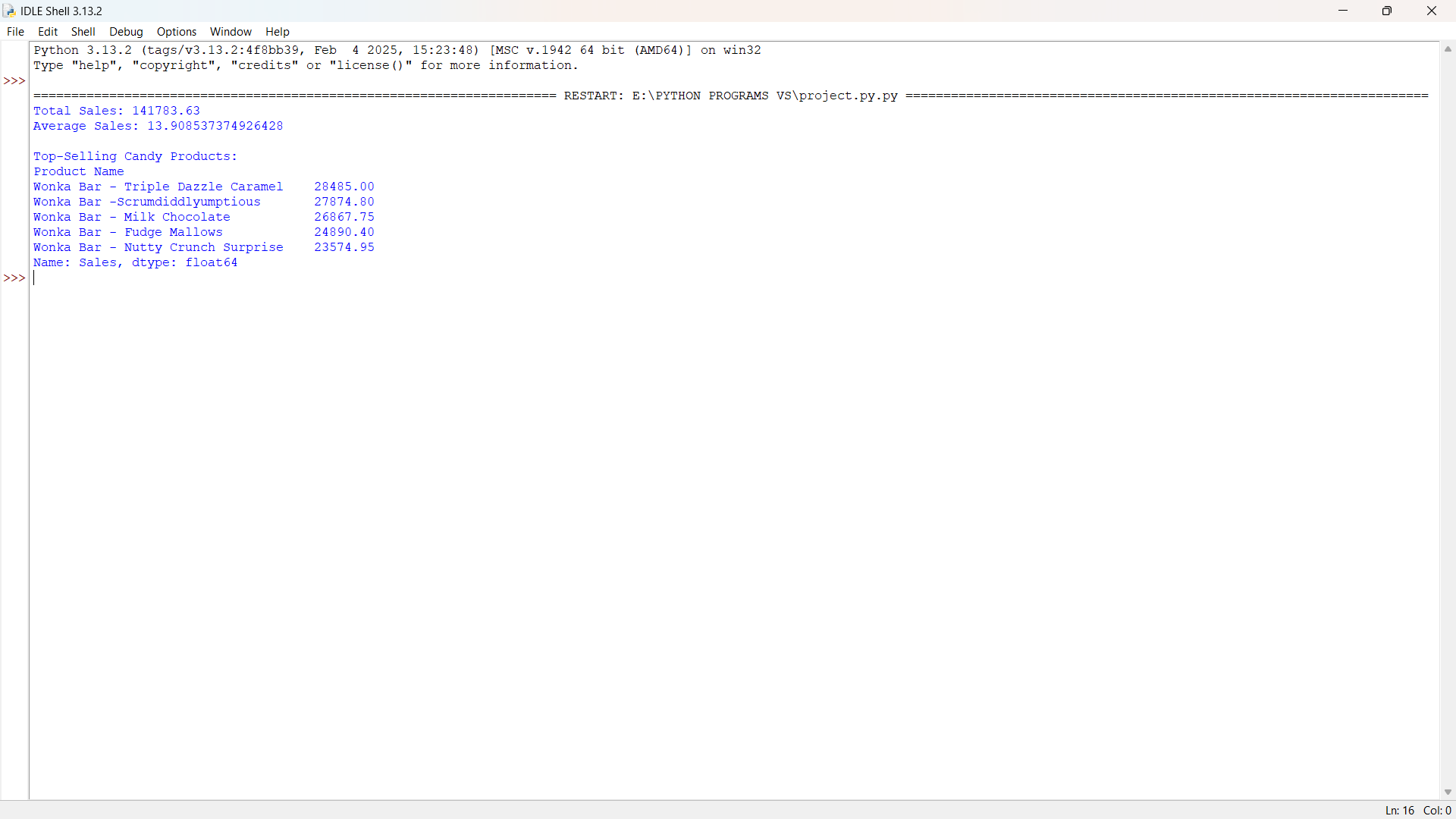
## **ii. General Description** The total and average sales were calculated using the .sum() and .mean() functions on the 'Sales' column of the dataset. These metrics serve as a baseline for comparing performance across regions, products, and time.

## **iii. Requirements**

## .sum() and .mean() functions

## **iv. Results** The script printed the total and average sales figures, which provide valuable insight into the scale and general profitability of the business operations.

## **v. Visualization** *(Printed as text output in the console)*



**Objective 3: Sales Trends Over Time**

**i. Introduction**  
Analyzing sales trends over time helps identify seasonality, growth patterns, and the effectiveness of promotional strategies. It supports forecasting and long-term planning.

**ii. General Description**  
The 'Order Date' column was converted to datetime format, and sales were grouped by year using .dt.year. The .groupby() and .sum() functions aggregated the yearly sales data, which was plotted using Matplotlib.

**iii. Requirements**

## pd.to\_datetime()

## .dt.year, .groupby(), .sum()

## Matplotlib line plot

## **iv. Results** The line plot showed fluctuations in sales over different years. Peaks may align with successful campaigns or product launches, while dips may indicate off-seasons or market challenges.

## **v. Visualization** *(Line plot: Sales Trends Over Time)*

## **Objective 4: Top-Selling Candy Products**

## **i. Introduction** Identifying top-selling products helps businesses focus on what works. It enables better stock management and marketing prioritization.

## **ii. General Description** Sales were grouped by 'Product Name' and summed using .groupby() and .sum(). The results were sorted in descending order to identify the top-selling candy products.

## **iii. Requirements**

## .groupby() and .sum()

## .sort\_values(ascending=False)

## **iv. Results** The top products list highlighted the best-performing items. These are likely candidates for promotion, bulk stocking, and possible product line expansion.

## **v. Visualization** *(Top product names and their sales were printed as output)*

## same as second objective

## **Objective 5: Distribution of Sales**

## **i. Introduction** Understanding how sales values are distributed helps identify outliers, common transaction values, and potential pricing issues.

## **ii. General Description** A histogram with KDE (Kernel Density Estimate) was plotted using Seaborn to visualize the distribution of the 'Sales' column.

## **iii. Requirements**

## sns.histplot() with kde=True

## Matplotlib for formatting

## **iv. Results** The distribution plot revealed the most common sales values and identified any anomalies in the data. A right-skewed curve, for example, would indicate many small-value sales and few large ones.

## **v. Visualization** *(Histogram with KDE: Distribution of Sales)*

## **Objective 6: Monthly Sales Patterns**

## **i. Introduction** Monthly analysis uncovers seasonality and purchasing behavior, which helps in planning inventory, marketing, and promotions effectively.

## **ii. General Description** The 'Order Date' column was used to extract the month, and sales were grouped by month using .groupby() and .sum(). The results were plotted to show monthly trends.

## **iii. Requirements**

## .dt.month, .groupby(), .sum()

## Matplotlib line plot

## **iv. Results** The monthly trend plot displayed sales peaks and dips across months. Certain months showed consistently high sales, indicating seasonal demand patterns.

## **v. Visualization** *(Line plot: Monthly Sales Patterns)*

## **Objective 7: Correlation and Pairwise Relationships**

## **i. Introduction** Studying correlations helps identify relationships between numerical variables. Strong correlations can guide predictive modeling and strategic decisions.

## **ii. General Description** A correlation matrix was generated for all numeric columns using .corr(), and visualized with a heatmap. A pairplot was also created using Seaborn to show pairwise relationships.

## **iii. Requirements**

## .corr() function

## sns.heatmap() and sns.pairplot()

## **iv. Results** The heatmap revealed strong and weak correlations among variables such as sales, discounts, and quantities. The pairplot provided a more granular view of how variables interact with each other.

## **v. Visualization** *(Heatmap and Pairplot: Correlation & Pairwise Analysis)*

## 

## 5.Conclusion

The analysis of the Candy Sales dataset has provided several valuable insights into the sales trends, product performance, and customer buying behavior. Through a thorough Exploratory Data Analysis (EDA) and in-depth examination of sales metrics, we were able to understand the dynamics of the candy market and identify key areas for strategic focus.

Key conclusions drawn from the analysis include:

* **High-performing Products**: A small group of candy products accounted for a significant portion of total sales and revenue, suggesting a classic 80/20 pattern (Pareto Principle). These products should be prioritized in inventory management and promotional efforts.
* **Seasonal Sales Patterns**: Sales spikes were evident during specific times of the year, particularly during festive and holiday seasons. This trend highlights the importance of aligning marketing campaigns and stock levels with seasonal demand.
* **Price Sensitivity**: Customer preference leaned toward products within a specific price range. Understanding this price sensitivity can help in setting optimal pricing strategies and launching effective discount campaigns.
* **Revenue Distribution**: While some products had high sales volumes, others with higher unit prices contributed more to revenue. Both types of products play important roles in balancing customer satisfaction and profitability.
* **Opportunities for Improvement**: Several products showed consistently low sales, which may indicate the need for review in terms of pricing, marketing, or product quality.

## 6. Future Scope

In the future, the analysis can be enhanced by incorporating additional data such as customer demographics, feedback, and purchase history to gain deeper insights into buying behavior and preferences. Implementing predictive analytics using machine learning models could help forecast future sales trends, optimize inventory management, and personalize marketing strategies. Integration with real-time sales tracking systems can enable dynamic pricing and demand-driven stock replenishment. Furthermore, analyzing competitor data and market trends can provide a broader perspective, helping businesses stay competitive and adapt to changing customer demands effectively.

7. Github / Linkedin Link

Github: <https://github.com/venkatesh-2005/Python-Project>

Linkedin: <https://www.linkedin.com/posts/venkatesh-madh5_datascience-python-pandas-activity-7316771593802633216-_X7q?utm_source=share&utm_medium=member_desktop&rcm=ACoAAEMMlBkBwPMXlLJmMMhAN15VPYz9Xy3aKg4>

## 8.References

 Dataset Source: Maven Analytics(link: <https://mavenanalytics.io/data-playground>)

 Pandas Documentation: <https://pandas.pydata.org/docs/>

 Matplotlib Documentation: <https://matplotlib.org/stable/contents.html>

 Seaborn Documentation: <https://seaborn.pydata.org/>

 NumPy Documentation: <https://numpy.org/doc/>

 Towards Data Science – Articles on EDA and Data Visualization

 Scikit-learn Documentation (if ML models are considered[): https://scikit-learn.org/stable/documentation.html](file:///D:\reference\):%20https:\scikit-learn.org\stable\documentation.html)

SOURCE CODE

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

data = pd.read\_csv("CSV File path")

#Objective1: Calculate total and average sales

total\_sales = data['Sales'].sum()

average\_sales = data['Sales'].mean()

print(f"Total Sales: {total\_sales}")

print(f"Average Sales: {average\_sales}")

#Objective 2: Analyze sales trends over time

data['Order Date'] = pd.to\_datetime(data['Order Date'])

data['Year'] = data['Order Date'].dt.year

sales\_trend = data.groupby('Year')['Sales'].sum()

plt.figure(figsize=(10, 6))

plt.plot(sales\_trend.index, sales\_trend.values, marker='o')

plt.title('Sales Trends Over Time')

plt.xlabel('Year')

plt.ylabel('Total Sales')

plt.show()

#Objective 3: Identify top-selling candy products

top\_products = data.groupby('Product Name')['Sales'].sum().sort\_values(ascending=False)

print("\nTop-Selling Candy Products:")

print(top\_products.head())

#Objective 4: Plot distribution of sales

plt.figure(figsize=(10, 6))

sns.histplot(data['Sales'], kde=True)

plt.title('Distribution of Sales')

plt.xlabel('Sales')

plt.ylabel('Frequency')

plt.show()

#Objective 5: Visualize regional sales performance

regional\_sales = data.groupby('Region')['Sales'].sum()

plt.figure(figsize=(10, 6))

regional\_sales.plot(kind='bar')

plt.title('Regional Sales Performance')

plt.xlabel('Region')

plt.ylabel('Total Sales')

plt.show()

#Objective 6: Examine monthly sales patterns

data['Month'] = data['Order Date'].dt.month

monthly\_sales = data.groupby('Month')['Sales'].sum()

plt.figure(figsize=(10, 6))

monthly\_sales.plot(marker='o')

plt.title('Monthly Sales Patterns')

plt.xlabel('Month')

plt.ylabel('Total Sales')

plt.show()

#Objective 7: Heatmap

numeric\_data = data.select\_dtypes(include=['number'])

correlation\_matrix = numeric\_data.corr()

plt.figure(figsize=(12, 10))

sns.heatmap(correlation\_matrix, annot=True, cmap='coolwarm', linewidths=.5)

plt.title('Correlation Heatmap of Candy Sales Data')

plt.show()

#Objective 8: Pairplot

sns.pairplot(numeric\_data)

plt.title('Pairplot of Candy Sales Data')

plt.show()