## Artificial Intelligence and Machine Learning

Project Report

Semester-IV (Batch-2022)

Superstore Sales Data

A red and white sign

Description automatically generated with low confidence

**Supervised By: Submitted By:**

Mohd Talib Ujjwal, 2210992478  
 Utsav , 2210992484  
 Vaibhav Rana ,2210992488  
 Tushar Singla, 2210992470

**Department of Computer Science and Engineering**

Chitkara University Institute of Engineering & Technology, Chitkara University, Punjab

# CONTENT

* + - **Introduction**
    - **Problem Definition and Requirement**
    - **Tools Used**
    - **Data Summary**
    - **Feature Description**
    - **EDA**
    - **Data splitting**
    - **Feature Manipulation & Selection**
    - **Models**
    - **Result**
    - **Conclusions**

**1. INTRODUCTION**  
**1.1 Background**

The retail landscape has undergone significant transformations in recent years, driven by a multitude of economic, social, and environmental influences. Traditional methods of assessing sales performance often struggle to accurately depict the intricate dynamics of the market.

However, with the advent of AI and ML technologies, there's a newfound potential to revolutionize the way sales data is analyzed and evaluated. Superstore sales data, in particular, offers a rich trove of information that can be leveraged to predict trends, optimize pricing strategies, and enhance overall business performance.  
  
1.2 **Objectives**

The objectives of this project are to:

* Develop a predictive model for estimating profit based on user inputs.
* Test and validate the model using a variety of datasets.
* Demonstrate the effectiveness of the model through metrics and visualizations.

**1.3 Significance**

The significance of this project lies in its potential to provide more accurate sales predictions for superstores, benefiting both retailers and consumers alike. Improved accuracy can lead to better inventory management, pricing strategies, and overall business forecasting. This means retailers can plan their stock levels more effectively, set achievable goals, and make informed decisions about promotions and marketing campaigns.

**2. Problem Definition and Requirements**

**2.1 Problem Statement**

The Superstore Sales Data consists of various attributes such as order details, customer information, product categories, sales figures, and geographical information. The goal is to leverage this dataset to gain insights that will help improve business performance, optimize operations, and enhance customer satisfaction.

**2.2 Software/Hardware Requirements**

* Software: Python, Jupyter Notebooks, TensorFlow/PyTorch, Pandas, Matplotlib/Seaborn, Scikit-Learn
* Hardware: A machine with at least 8GB RAM and a multi-core processor
* Datasets: Publicly available fitness datasets or custom datasets created by the project team

**2.3 Proposed Design / Methodology**

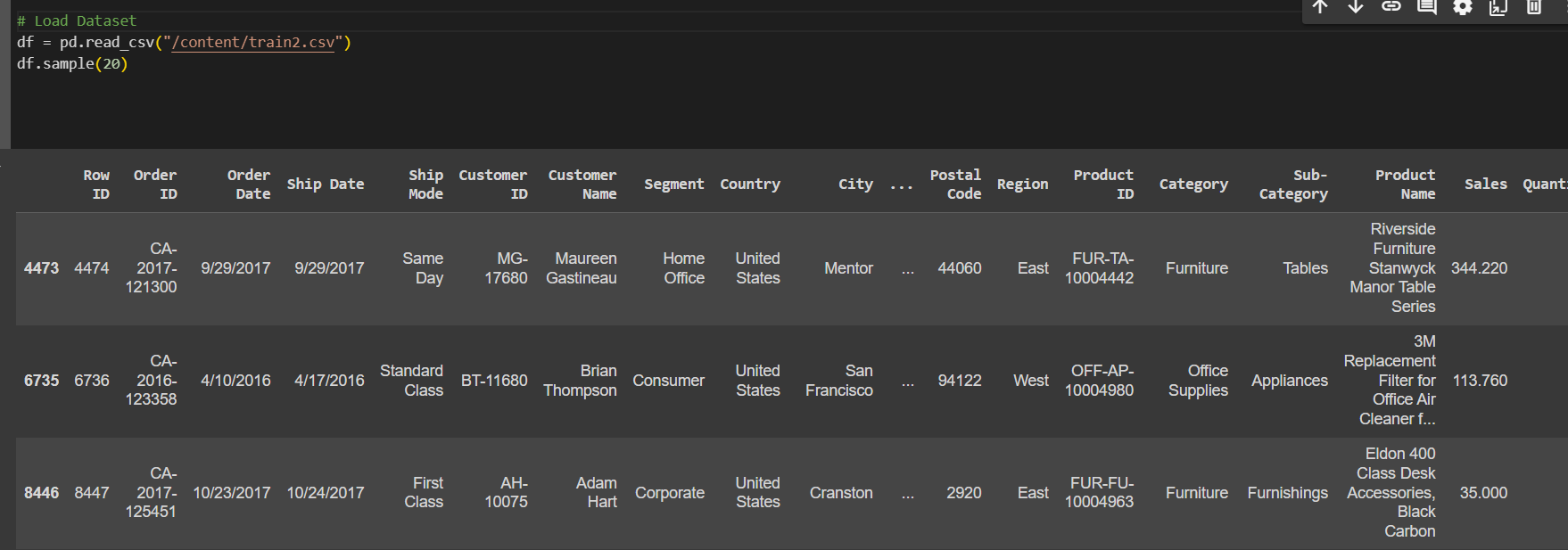
The proposed design for the House Price Prediction project involves the following key steps:

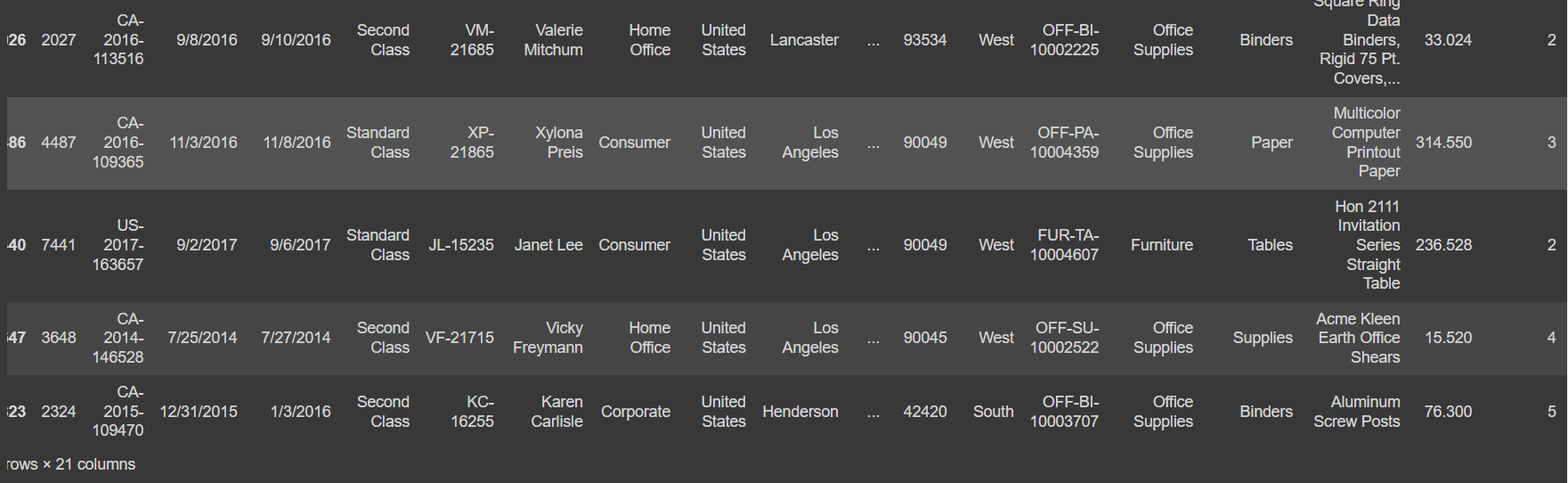
1. Data Collection and Preprocessing: Collect and clean data sets with relevant attributes such as House Price , Neighbourhood , Availability etc.
2. Feature Engineering: Identify and create meaningful features for training the ML model.
3. Model Selection: Choose suitable algorithms for prediction, such as linear regression, decision trees, or neural networks.
4. Model Training and Validation: Train the model on the dataset and validate its accuracy using appropriate metrics like RMSE, MAE, or R-squared.
5. Model Deployment: Outline how the model could be deployed in a real-world application.

**3. Tool Used**

* Google colaboratory is used as IDE.
* Pandas and NumPy are used for Data Manipulation & Pre-processing and Mathematical functions respectively.
* Exploratory data analysis is automated by data prep.
* For visualization of the plots, Matplotlib, Seaborn, Plotty are used.
* GitHub is used as version control system

**4. Data Summary**This is the Superstore sales data Prediction . In the below table it shows the top and bottom 5 rows respectively

****

**Fig 4.1**

**Fig 4.2**

**5. Features Description**

* 'Region': Geographic regions or areas where sales transactions take place.
* 'Category': Broad categories to which products belong (e.g., office supplies, technology, furniture).
* 'Sub-Category': Subcategories within product categories (e.g., chairs, tables, phones).
* Row ID: Sequential index for dataset organization and management.
* Order ID: Unique identifier for tracking individual sales orders.
* Order Date: Date when a sales order was placed by a customer.
* Ship Date: Date when ordered products are shipped to customers.
* Product ID: Unique identifier for individual products in the inventory.
* Postal Code: Numerical or alphanumeric code indicating geographic areas for postal delivery

# 6. Exploratory Data Analysis (EDA)

# Exploratory Data Analysis (EDA) is indeed a critical step in the data analysis process. It involves analyzing and visualizing data to understand its main characteristics, uncover patterns, identify anomalies, and formulate hypotheses. EDA typically includes tasks such as:

# Summary Statistics: Calculating and examining descriptive statistics such as mean, median, mode, standard deviation, and percentiles to understand the central tendency, dispersion, and shape of the data distribution.

# Data Visualization: Creating visualizations such as histograms, box plots, scatter plots, and heatmaps to explore the relationships between variables, detect outliers, and understand the data's structure.

# Missing Values Handling: Identifying missing values and determining appropriate strategies for handling them, such as imputation or removal.

# Feature Engineering: Creating new features or transforming existing ones to better represent the underlying patterns in the data and improve model performance.

# Correlation Analysis: Investigating the relationships between variables using correlation matrices or pairwise scatter plots to identify potential multicollinearity issues.

# Outlier Detection: Identifying data points that deviate significantly from the rest of the data and understanding whether they represent genuine anomalies or errors.

# Data Distribution Analysis: Examining the distribution of variables to assess whether they follow specific probability distributions and determine appropriate modeling techniques.

# Pattern Recognition: Searching for patterns, trends, or clusters in the data that may provide insights into the underlying processes or relationships.

# Data Cleaning: Preprocessing the data by removing duplicates, standardizing formats, and resolving inconsistencies to ensure its quality and reliability.

# 6.1 Charts

# EDA is done using charts also which helps to find relationship between various features or columns . We have many types of charts like violin plot, Scatter plot, Bar plot ,Pair plot, etc .Below are few Representation Of Graphs

# 6.1.1 Pair Plot

# A pair plot is a graphical tool used in Exploratory Data Analysis (EDA) to visualize the relationships between pairs of variables in a dataset. It typically consists of a grid of scatterplots, where each scatterplot represents the relationship between two variables, and the diagonal of the grid displays the distribution of each variable.

# Fig 6.1

# 6.1.2 Violin Plot

# A violin plot is a type of data visualization that combines the characteristics of a box plot with a kernel density plot. It provides insights into the distribution and density of the data across different levels or categories of a categorical variable.

# 

Fig 6.2

# 6.2 Feature Engineering & Data Pre-processing

# Handling Missing Values

**Fig 6.2.2.1**

# Data Encoding

# Encoding in the context of data refers to the process of converting data from one form to another, often to facilitate storage, processing, or transmission. There are many types of encoding used in data science and computing like Binary , one Hot, labelled encoding

# Fig 6.2.2.1

# Data Scaling

# Data scaling, also known as data normalization or feature scaling, is a preprocessing step used in machine learning and data analysis to standardize the range of independent variables or features of the dataset. The goal is to ensure that each feature contributes equally to the analysis and to prevent features with larger scales from dominating those with smaller scales.

# 6.2.2 Categorical Encoding

categorical encoding is a crucial preprocessing step when dealing with categorical variables such as "category," "sub-category”, “Region ". Here's how categorical encoding might be applied in this scenario:

**2.1 Label Encoding:**

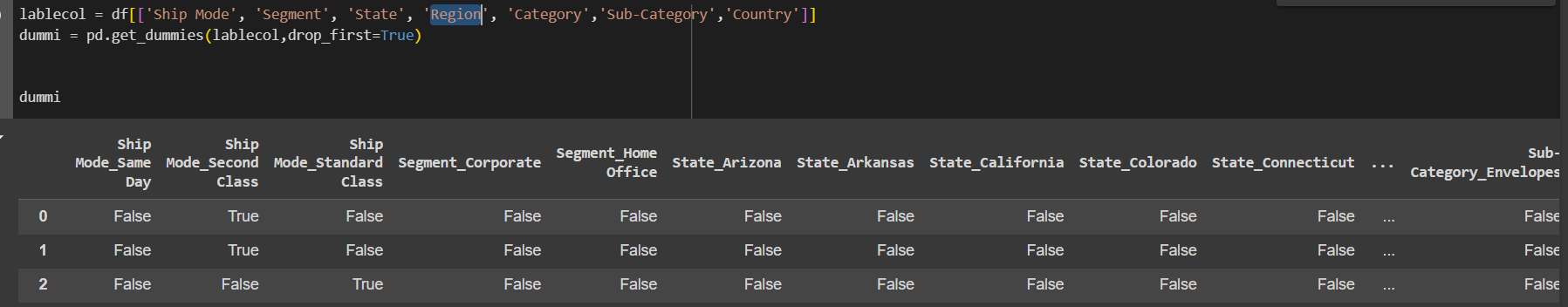
We have not used label encoding in our project as dependent column already contain numerical variables.

**2.2 One-Hot Encoding:**

One-hot encoding transforms each category within a sub-categorical variable into a binary feature. Each category becomes a separate binary feature (0 or 1), indicating its presence or absence in the observation.

**2.3 Dummy Encoding:**

Dummy encoding is similar to one-hot encoding but avoids the "dummy variable trap" by using one less binary feature than the total number of categories.



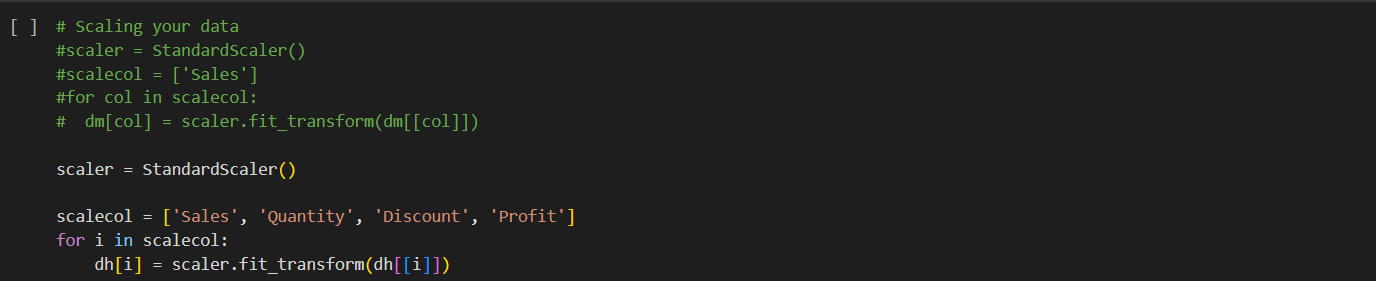
**Fig 6.2.2.1**

## 6.2.3 Data Scaling:

## Data scaling, also known as data normalization or feature scaling, is a preprocessing step used in machine learning and data analysis to standardize the range of independent variables or features of the dataset. The goal is to ensure that each feature contributes equally to the analysis and to prevent features with larger scales from dominating those with smaller scales.

## StandardScaler from scikit-learn library is used to scale the data. The StandardScaler scales each feature by subtracting the mean and dividing by the standard deviation, which results in a distribution with a mean of 0 and a standard deviation of 1

## It is less sensitive to outliers compared to other scaling methods like min-max scaling that's why we have used it here.



## Fig 6.2.3.1

## 7. Data Splitting:

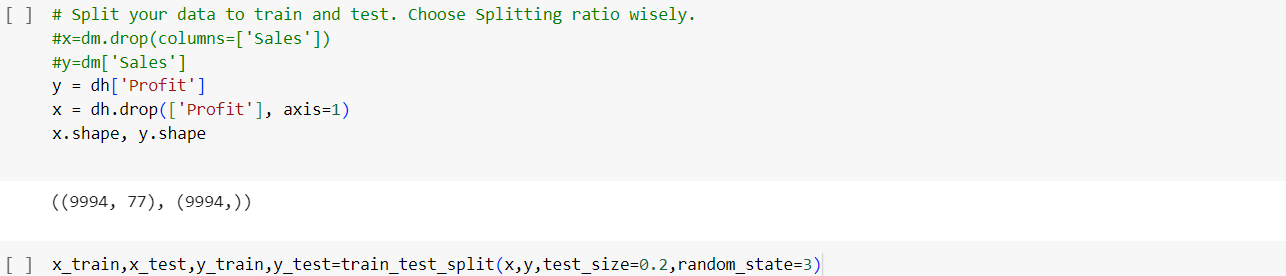
It involves dividing the available dataset into separate subsets for training, validation, and testing. Here's how data splitting can be implemented:

**Training Set:**

The training set is used to train the machine learning model. It typically comprises the majority of the dataset, allowing the model to learn patterns and relationships from the data. A common split is to allocate around 70-80% of the data to the training set.

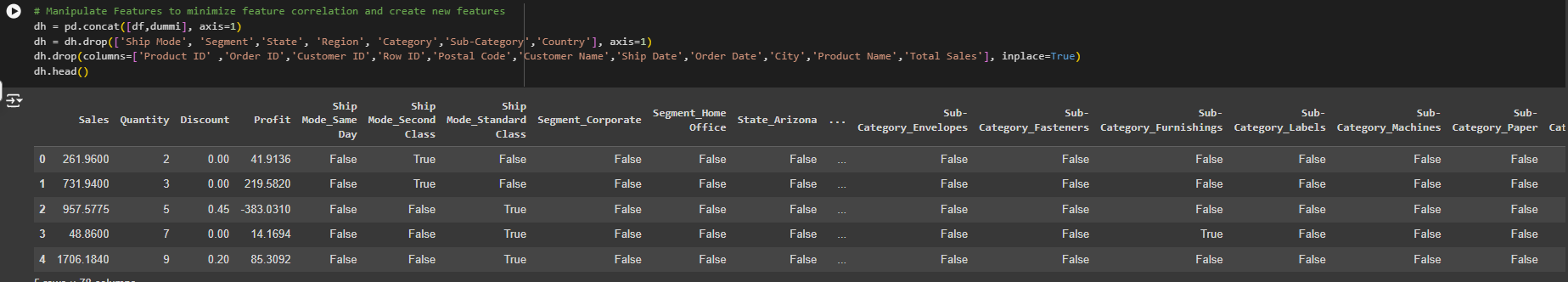
**Testing Set:**

The testing set is used to assess the final performance of the trained model. It represents unseen data that the model has not been exposed to during training or validation.

**Fig 7.1**

**8. Feature Manipulation & Selection**

Feature manipulation and selection are crucial steps in the process of preparing data for machine learning models. This involves creating new features or modifying existing ones to improve the performance of machine learning models. It can include transformations like scaling, binning, one-hot encoding, or creating interaction terms.



**Fig 8.1**

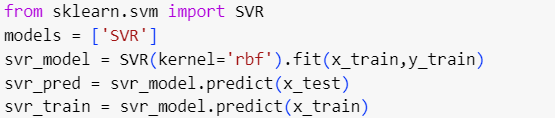
**9. Models**

**9.1 Model 1:**

**Support Vector Regressor (SVR):**

Support Vector Regression (SVR) is a powerful supervised learning algorithm utilized for regression tasks, particularly effective in scenarios where linear regression techniques might not be suitable. SVR operates by identifying the hyperplane within the feature space that best represents the relationship between the independent variables and the target variable, while simultaneously aiming to minimize the error between the actual and predicted values.

Unlike traditional regression models that focus solely on minimizing errors, SVR introduces the concept of a margin, which helps control overfitting by allowing a certain degree of deviation from the hyperplane. This margin essentially acts as a buffer zone, permitting some level of error in the predictions to prevent the model from fitting the noise in the data excessively.



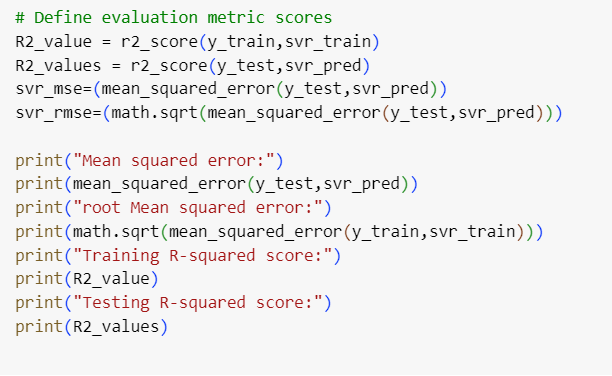


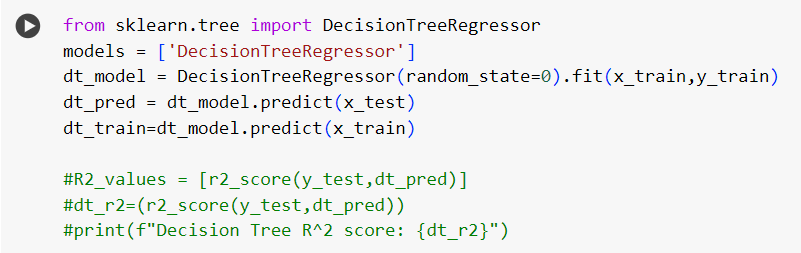
Fig 9.1

**9.2 Model 2:**

**Decision Tree Regression:**

It is a versatile supervised learning algorithm widely used for regression tasks, offering intuitive interpretability and robust performance across various domains.

At its core, the Decision Tree Regressor partitions the feature space into regions, creating a hierarchical structure of decision nodes that collectively form a tree-like model. Each node represents a specific feature and a corresponding threshold value, and the tree is built by recursively splitting the data based on these features to minimize the error in predicting the target variable.



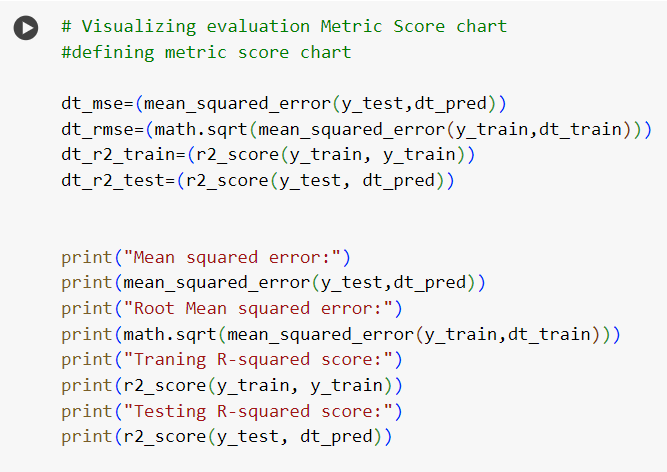
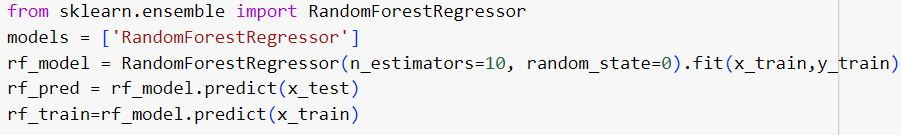


Fig 9.2.1

**9.3 Model 3:**

**Random Forest:**

Random Forest is a sophisticated machine learning algorithm that operates by constructing numerous decision trees during the training phase. Each decision tree is trained on a different subset of the original dataset, which is selected randomly with replacement. Additionally, at each node of every tree, only a random subset of features is considered for splitting. This randomness in both data and feature selection helps to decorrelate the trees and reduce overfitting, a common problem in machine learning where the model performs well on training data but poorly on unseen data.

Fig 9.3.1

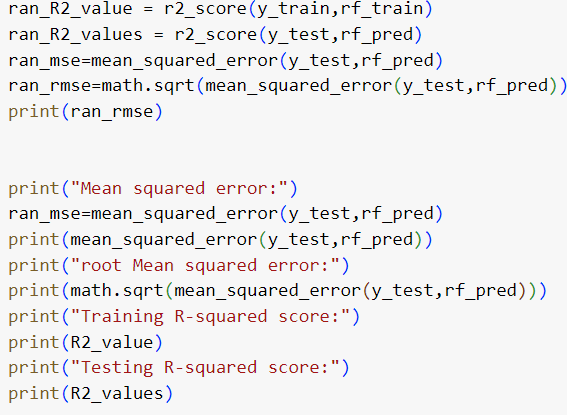


Fig 9.3.2

**10. Results**

Model Performance Metrics: Accuracy, RMSE, MAE, R-squared, etc.

Training and Validation Curves: Plots showing training loss/accuracy over time.

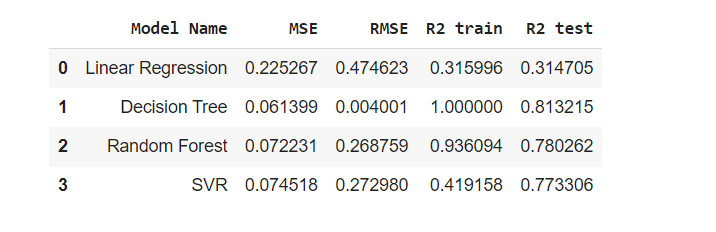
Example Predictions: Screenshot or tables showing example predictions from the model.

Fig 10.1

11. Conclusion:

**1. Summary of Findings:** Through our analysis of superstore sales data, several key insights have emerged. We've identified trends indicating fluctuations in profit margins across different product categories and seasons.

2. **Limitations:** Despite the valuable insights gained from our analysis, it's essential to acknowledge certain limitations. These may include incomplete or inaccurate data, assumptions made during the modeling process, and the complexity of capturing all variables influencing profitability within the scope of the project.

**3. Future Research Directions:** Incorporating Additional Data Sources: Integrating external data sources, such as demographic information or economic indicators, could provide deeper insights into consumer behavior and market trends.

Refining Models with Advanced Techniques: Exploring advanced machine learning algorithms and statistical techniques could improve the accuracy and predictive power of our profit prediction models.