MARKET BASKET INSIGHTS PROJECT

PROJECT INTRODUCTION

This project aims to analyze market basket data. In this notebook we will load and preprocess the dataset.

1. Loading and Preprocessing the Data

- Data Acquisition: Begin by acquiring the dataset relevant to your project. This might involve web scraping, accessing APIs, collecting sensor data, or using pre-existing datasets. Ensuring that the data is legally obtained and well-documented.
- Data Cleaning: Inspecting the data for missing values, duplicates, and outliers. Address these issues through data cleaning techniques, such as imputation, removal, or transformation.
- Data Transformation: Converting the data into a suitable format for analysis. This may include one-hot encoding, scaling, or normalizing numerical features. For unstructured data (e.g., text or images), preprocessing might involve tokenization or image resizing.

Python Code:

```
import pandas as pd
import numpy as np
from sklearn.preprocessing import StandardScaler
from sklearn.impute import SimpleImputer
data = pd.read_csv('your_dataset.csv')
missing_values = data.isnull().sum()
data = data.drop_duplicates()
imputer = SimpleImputer(strategy='mean')
data['column_with_missing_values'] =
imputer.fit_transform(data[['column_with_missing_values']])
data = pd.get_dummies(data, columns=['categorical_column'])scaler = StandardScaler()
data['numerical_feature'] = scaler.fit_transform(data[['numerical_feature']])
```

2. Perform Data Analysis

- Data Exploration: Conducting initial exploratory data analysis (EDA) to gain an understanding of the dataset. Use summary statistics, visualizations, and descriptive analytics to reveal patterns, trends, and relationships in the data.
- Feature Engineering: Creating new features or modify existing ones to enhance the predictive power of your model. Feature engineering may involve domain-specific knowledge or dimensionality reduction techniques.
- Model Development: Selecting an appropriate machine learning or deep learning model based on your project's goals. Train the model using the preprocessed data.
- Model Evaluation: Assessing the model's performance using relevant evaluation metrics. Depending on the project, this might include accuracy, precision, recall, F1-score, ROC-AUC, or mean squared error.

Python Code:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix
data = pd.read_csv('your_dataset.csv')
summary_stats = data.describe()
print("Summary Statistics:")
print(summary_stats)
plt.hist(data['numerical_feature'])
plt.title("Histogram of Numerical Feature")
plt.xlabel("Value")
plt.ylabel("Frequency")
plt.show()
data['new_feature'] = data['feature1'] + data['feature2']
X = data.drop('target', axis=1) # Assuming 'target' is your target variable
y = data['target']X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)
```

```
model = RandomForestClassifier()
model.fit(X_train, y_train)
y_pred = model.predict(X_test)
accuracy = accuracy_score(y_test, y_pred)
classification_rep = classification_report(y_test, y_pred)
confusion_mat = confusion_matrix(y_test, y_pred)
print("Model Evaluation Metrics:")
print(f"Accuracy: {accuracy}")
print("Classification Report:")
print(classification_rep)
print("Confusion Matrix:")
print(confusion_mat)
```

3. Document Your Analysis

- Project Overview: Begin the document with a brief introduction, explaining the project's context, goals, and datasets used.
- Data Preprocessing: Describing the data collection process, cleaning steps, and transformations applied. Include visualizations or summary statistics to illustrate the data's characteristics.
- Data Analysis: Presenting the results of your EDA and feature engineering efforts. Use clear and well-organized visualizations and tables to convey your findings.
- Model Development: Explaining the choice of model, its architecture, and the training process. Including information about hyper parameters and any tuning.
- Model Evaluation: Discussing the model's performance, including key evaluation metrics. Providing insights into the model's strengths and limitations.
- Conclusion: Summarizing the key takeaways, the success of the project, and potential areas for improvement.
- Appendices: Including code snippets, data dictionaries, and any additional information to support our analysis.

Python Code:

```
# Import necessary libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as pltfrom sklearn.model_selection import train_test_split
```

```
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score
print("Project Overview")
print("This project aims to predict house prices using a linear regression model.")
print("We will analyze a dataset of house features to make predictions.")
print("Dataset source: [Provide source link]")
print("\nData Preprocessing")
data = pd.read_csv('house_prices.csv')
data = data.dropna()
plt.hist(data['price'], bins=20)
plt.xlabel('Price')
plt.ylabel('Frequency')
plt.title('Distribution of House Prices')
plt.show()
print("\nData Analysis")
summary_stats = data.describe()
print(summary_stats)
print("\nModel Development")
# Split data into features and target variable
X = data.drop('price', axis=1)
y = data['price']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
model = LinearRegression()
model.fit(X_train, y_train)
print("\nModel Evaluation")
y_pred = model.predict(X_test)
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)
print(f"Mean Squared Error: {mse}")
print(f"R-squared (R2) Score: {r2}")
print("\nConclusion")
print("The linear regression model has been developed and evaluated.")
print("The Mean Squared Error and R-squared score indicate the model's performance.")
print("Further improvements can be made by exploring more complex models and feature engineering.")
```

CONCLUSION:

Market basket insights derived from the development process provide valuable information for businesses. They enable retailers to enhance the customer shopping experience, optimize inventory management, and design effective promotional strategies. By understanding item associations and customer preferences, businesses can tailor their offerings, improve cross-selling opportunities, and ultimately increase revenue. Continuous monitoring and adaptation to changing consumer behavior are crucial for sustained success in leveraging market basket insights.