**Step 3: Exploratory Data Analysis (EDA)**

Objective:

The primary goal of EDA in the SmartChef project is to uncover insights about recipe ratings, nutritional content, recipe categories, and how these factors might influence recipe recommendations.

Tools and Libraries:

You'll need Python libraries such as Pandas for data manipulation, Matplotlib and Seaborn for visualization. Ensure they are installed:

bashCopy code

pip install pandas matplotlib seaborn

3.1 Initial Data Overview

1. **General Structure**: Get a sense of the dataset's structure, column types, and first few entries.

pythonCopy code

data.info() print(data.head())

1. **Summary Statistics**: Review summary statistics for numerical columns, focusing on nutritional information and ratings.

pythonCopy code

print(data.describe())

3.2 Analyze Recipe Ratings

1. **Distribution of Ratings**: Understand the distribution of recipe ratings.

pythonCopy code

import matplotlib.pyplot as plt import seaborn as sns sns.histplot(data['rating'], bins=20, kde=True) plt.title('Distribution of Recipe Ratings') plt.xlabel('Rating') plt.ylabel('Frequency') plt.show()

1. **Rating vs. Nutritional Content**: Explore if there's a correlation between recipe ratings and their nutritional content (calories, fat, protein, sodium).

pythonCopy code

sns.pairplot(data[['rating', 'calories', 'fat', 'protein', 'sodium']], diag\_kind='kde') plt.show()

3.3 Explore Nutritional Information

1. **Nutrient Distribution**: Examine the distribution of key nutrients across recipes.

pythonCopy code

data[['calories', 'fat', 'protein', 'sodium']].hist(bins=15, figsize=(10, 7)) plt.suptitle('Nutritional Information Distribution') plt.show()

1. **Outliers in Nutritional Data**: Identify potential outliers that could skew analysis.

pythonCopy code

sns.boxplot(data=data[['calories', 'fat', 'protein', 'sodium']]) plt.title('Box Plot of Nutritional Information') plt.show()

3.4 Recipe Categories Analysis

Assuming your dataset contains binary columns for recipe categories or tags:

1. **Top Recipe Categories**: Identify and visualize the most common categories/tags.

pythonCopy code

categories\_sum = data.iloc[:, 6:].sum().sort\_values(ascending=False) # Adjust index as per your dataset top\_categories = categories\_sum.head(10) sns.barplot(x=top\_categories.values, y=top\_categories.index) plt.title('Top 10 Recipe Categories') plt.xlabel('Number of Recipes') plt.show()

1. **Category vs. Rating**: Explore how different categories/tags might influence recipe ratings.

pythonCopy code

# Example for a single category - repeat for others of interest sns.boxplot(x='vegetarian', y='rating', data=data) plt.title('Recipe Rating by Vegetarian Tag') plt.xlabel('Vegetarian') plt.ylabel('Rating') plt.show()

3.5 Correlation Analysis

1. **Nutritional Correlation**: Assess the correlation among nutritional metrics and ratings.

pythonCopy code

correlation\_matrix = data[['rating', 'calories', 'fat', 'protein', 'sodium']].corr() sns.heatmap(correlation\_matrix, annot=True, cmap='coolwarm') plt.title('Correlation Matrix') plt.show()

**Conclusion and Next Steps**

EDA provides valuable insights into your dataset, helping to inform the development of your recipe recommendation system. Through EDA, you've gained a better understanding of the distribution of ratings, the relationship between nutritional content and ratings, and the prevalence of different recipe categories.

Based on these findings, you can:

* **Refine Data Cleaning**: Adjust for outliers in nutritional data.
* **Feature Selection**: Identify the most relevant features for predicting recipe ratings or for categorizing recipes.
* **Model Development**: Proceed with informed choices in machine learning model development, focusing on features and relationships that appear most promising.

Top of Form