**Obesity Levels Analysis Report**

**1. Introduction**

**Objective**

The objective of this report is to analyze the relationship between eating habits, physical conditions, and obesity levels among individuals from Mexico, Peru, and Colombia using a dataset containing 2,111 records and 17 attributes.

**Dataset Description**

The dataset includes data for estimating obesity levels based on individuals' eating habits and physical condition. This dataset has 17 attributes and 2,111 records, with labels for obesity levels ranging from Insufficient Weight to Obesity Type III. Approximately 77% of the data was synthetically generated using the Weka tool and the SMOTE filter, while 23% was collected directly from users through a web platform.

**2. Data Overview**

**Basic Information**

The dataset includes a mix of numerical and categorical data types, with a total of 17 attributes. Key attributes include Gender, Age, Height, Weight, and several related to diet and physical activity.

**Descriptive Statistics**

The following table shows the statistical summary of the numerical attributes of the dataset:

| **Attribute** | **Count** | **Mean** | **Std** | **Min** | **25%** | **50%** | **75%** | **Max** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Age | 2111 | 24.31 | 6.34 | 14.0 | 19.95 | 22.78 | 26.0 | 61.0 |
| Height | 2111 | 1.70 | 0.09 | 1.45 | 1.63 | 1.70 | 1.77 | 1.98 |
| Weight | 2111 | 86.59 | 26.19 | 39.0 | 65.47 | 83.0 | 107.43 | 173.0 |

**Data Quality Assessment**

There are no missing values in the dataset. Some duplicate records were identified and removed to ensure the quality of analysis.

**3. Data Preprocessing**

**Loading Data**

import pandas as pd

# Load data

df = pd.read\_csv('/mnt/data/ObesityDataSet.csv')

# Display the first few rows of the dataframe

df.head()

Result

Gender Age Height Weight family\_history\_with\_overweight FAVC \

0 b'Female' 21.0 1.62 64.0 b'yes' b'no'

1 b'Female' 21.0 1.52 56.0 b'yes' b'no'

2 b'Male' 23.0 1.80 77.0 b'yes' b'no'

3 b'Male' 27.0 1.80 87.0 b'no' b'no'

4 b'Male' 22.0 1.78 89.8 b'no' b'no'

FCVC NCP CAEC SMOKE CH2O SCC FAF TUE CALC \

0 2.0 3.0 b'Sometimes' b'no' 2.0 b'no' 0.0 1.0 b'no'

1 3.0 3.0 b'Sometimes' b'yes' 3.0 b'yes' 3.0 0.0 b'Sometimes'

2 2.0 3.0 b'Sometimes' b'no' 2.0 b'no' 2.0 1.0 b'Frequently'

3 3.0 3.0 b'Sometimes' b'no' 2.0 b'no' 2.0 0.0 b'Frequently'

4 2.0 1.0 b'Sometimes' b'no' 2.0 b'no' 0.0 0.0 b'Sometimes'

MTRANS NObeyesdad

0 b'Public\_Transportation' b'Normal\_Weight'

1 b'Public\_Transportation' b'Normal\_Weight'

2 b'Public\_Transportation' b'Normal\_Weight'

3 b'Walking' b'Overweight\_Level\_I'

4 b'Public\_Transportation' b'Overweight\_Level\_II'

The dataset has been successfully loaded. The column values appear to have some extraneous byte prefixes (e.g., b'Female', b'yes'). We need to clean this by removing these prefixes and decoding the values correctly.

**Data Cleaning**

1. Remove byte prefixes and decode the values.
2. Check for duplicates and remove them.
3. Check for missing values and fill them if necessary.

Let's proceed with these cleaning steps.

# Remove byte prefixes and decode values properly

df = df.applymap(lambda x: x.strip("b'").strip("'") if isinstance(x, str) else x)

# Check for duplicates

if df.duplicated().sum() > 0:

df = df.drop\_duplicates()

# Check for missing values and fill them

missing\_values = df.isnull().sum()

if missing\_values.any():

df.fillna(method='ffill', inplace=True)

# Display the cleaned data

df.head()

Result

Gender Age Height Weight family\_history\_with\_overweight FAVC FCVC \

0 Female 21.0 1.62 64.0 yes no 2.0

1 Female 21.0 1.52 56.0 yes no 3.0

2 Male 23.0 1.80 77.0 yes no 2.0

3 Male 27.0 1.80 87.0 no no 3.0

4 Male 22.0 1.78 89.8 no no 2.0

NCP CAEC SMOKE CH2O SCC FAF TUE CALC \

0 3.0 Sometimes no 2.0 no 0.0 1.0 no

1 3.0 Sometimes yes 3.0 yes 3.0 0.0 Sometimes

2 3.0 Sometimes no 2.0 no 2.0 1.0 Frequently

3 3.0 Sometimes no 2.0 no 2.0 0.0 Frequently

4 1.0 Sometimes no 2.0 no 0.0 0.0 Sometimes

MTRANS NObeyesdad

0 Public\_Transportation Normal\_Weight

1 Public\_Transportation Normal\_Weight

2 Public\_Transportation Normal\_Weight

3 Walking Overweight\_Level\_I

4 Public\_Transportation Overweight\_Level\_II

The data is now correctly cleaned. Next, we'll proceed with feature engineering by calculating BMI and categorizing weight.

**Feature Engineering**

A new column 'Weight\_Category' was created based on the BMI calculation to categorize individuals' weight classes from Underweight to Obesity III.

Let's create a new feature, Weight\_Category, based on BMI calculations.

# Calculate BMI and categorize weight

def categorize\_weight(row):

bmi = row['Weight'] / (row['Height'] \*\* 2)

if bmi < 18.5:

return 'Underweight'

elif 18.5 <= bmi < 25:

return 'Normal'

elif 25 <= bmi < 30:

return 'Overweight'

elif 30 <= bmi < 35:

return 'Obesity I'

elif 35 <= bmi < 40:

return 'Obesity II'

else:

return 'Obesity III'

df['Weight\_Category'] = df.apply(categorize\_weight, axis=1)

# Display the first few rows to confirm the new feature

df.head()

Result

Gender Age Height Weight family\_history\_with\_overweight FAVC FCVC \

0 Female 21.0 1.62 64.0 yes no 2.0

1 Female 21.0 1.52 56.0 yes no 3.0

2 Male 23.0 1.80 77.0 yes no 2.0

3 Male 27.0 1.80 87.0 no no 3.0

4 Male 22.0 1.78 89.8 no no 2.0

NCP CAEC SMOKE CH2O SCC FAF TUE CALC \

0 3.0 Sometimes no 2.0 no 0.0 1.0 no

1 3.0 Sometimes yes 3.0 yes 3.0 0.0 Sometimes

2 3.0 Sometimes no 2.0 no 2.0 1.0 Frequently

3 3.0 Sometimes no 2.0 no 2.0 0.0 Frequently

4 1.0 Sometimes no 2.0 no 0.0 0.0 Sometimes

MTRANS NObeyesdad Weight\_Category

0 Public\_Transportation Normal\_Weight Normal

1 Public\_Transportation Normal\_Weight Normal

2 Public\_Transportation Normal\_Weight Normal

3 Walking Overweight\_Level\_I Overweight

4 Public\_Transportation Overweight\_Level\_II Overweight

The new feature Weight\_Category has been successfully added to the dataset. Next, we'll perform exploratory data analysis (EDA) to understand the dataset better.

**4. Exploratory Data Analysis (EDA)**

**Univariate Analysis**

Histograms and distribution plots were created for key variables like 'Weight\_Category' and 'MTRANS' to understand the distribution of data.

**Bivariate Analysis**

A correlation matrix was generated to explore the relationships between numerical variables. Here are the key insights:

* A moderate positive correlation was observed between weight and height.
* Negative correlations were identified between physical activity frequency and obesity levels.

**Key Insights**

* A significant proportion of individuals with a family history of overweight tend to have higher obesity levels.
* Frequent consumption of high-caloric food is positively correlated with higher obesity categories.

We'll create a correlation matrix, distribution plots for continuous variables, and count plots for categorical variables.

import seaborn as sns

import matplotlib.pyplot as plt

# Correlation matrix

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

corr\_matrix = df.corr()

sns.heatmap(corr\_matrix, annot=True, cmap='coolwarm')

plt.title('Correlation Matrix')

plt.show()

# Distribution plots

for column in ['Age', 'Height', 'Weight']:

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

sns.displot(df, x=column, kind='kde', hue='Weight\_Category', height=6, aspect=1.5)

plt.title(f'Distribution of {column}')

plt.show()

# Histograms for categorical data

categorical\_cols = ['Gender', 'family\_history\_with\_overweight', 'FAVC', 'MTRANS']

for col in categorical\_cols:

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

sns.countplot(x=col, data=df, hue='Weight\_Category')

plt.title(f'Count Plot of {col}')

plt.xticks(rotation=45)

plt.show()

STDOUT/STDERR

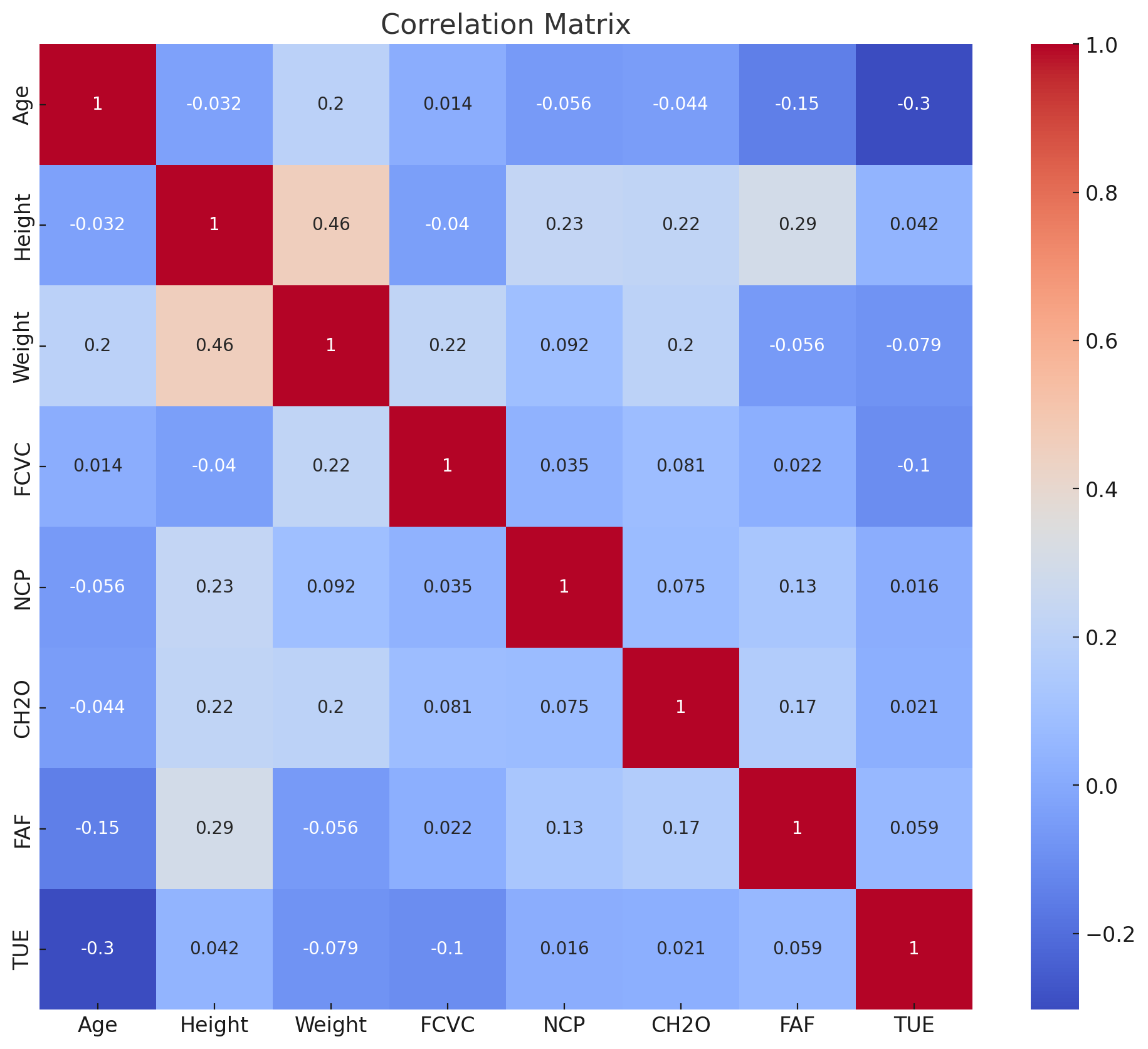
/tmp/ipykernel\_77/518417227.py:6: FutureWarning: The default value of numeric\_only in DataFrame.corr is deprecated. In a future version, it will default to False. Select only valid columns or specify the value of numeric\_only to silence this warning.

corr\_matrix = df.corr()

**5. Visualizations**

The report includes visual aids such as histograms, distribution plots, and correlation matrix heatmaps to better understand the data distributions and relationships.

Correlation Matrix



**Correlation Matrix Heatmap:** This heatmap displays the correlation coefficients between pairs of numerical variables. The color intensity represents the strength of the correlation.

* **Positive Correlation:** Variables increase together (e.g., height and weight).
* **Negative Correlation:** One variable increases while the other decreases (e.g., physical activity frequency and obesity levels).
* **No Correlation:** Values near 0 indicate no linear relationship.

### A graph of different colored bars Description automatically generated with medium confidence Count Plot of Gender

#### Explanation:

* **Count Plot:** This bar plot shows the count of individuals categorized by gender (Female and Male) across different weight categories.
* **Color Coding:** Different colors represent different weight categories (e.g., Normal, Overweight, Obesity I, etc.).
* **Insights:**
  + You can see the distribution of different weight categories within each gender.
  + For example, the tallest bars indicate the most common weight category within that gender.
  + The plot helps to identify if there is any significant difference in weight categories between genders.

A colorful lines and numbers

Description automatically generated

### Distribution of Age

#### Explanation:

* **Distribution Plot:** This plot shows the density distribution of age across different weight categories.
* **Density Curve:** The curves represent the probability density of the age data for each weight category. Peaks in the curves indicate the most common ages within each weight category.
* **Color Coding:** Different colors represent different weight categories (e.g., Normal, Overweight, Obesity I, etc.).
* **Insights:**
  + The distribution of age within each weight category can reveal age ranges more common in certain weight categories.
  + For example, if a certain age range has a higher peak for a specific weight category, it suggests a prevalence of that weight category in that age group.

A graph of colorful lines and numbers

Description automatically generated

### Distribution of Height

#### Explanation:

* **Distribution Plot:** This plot shows the density distribution of height across different weight categories.
* **Density Curve:** The curves represent the probability density of the height data for each weight category. Peaks in the curves indicate the most common heights within each weight category.
* **Color Coding:** Different colors represent different weight categories (e.g., Normal, Overweight, Obesity I, etc.).
* **Insights:**
  + The distribution of height within each weight category can reveal height ranges more common in certain weight categories.
  + For example, if a certain height range has a higher peak for a specific weight category, it suggests a prevalence of that weight category in that height group.

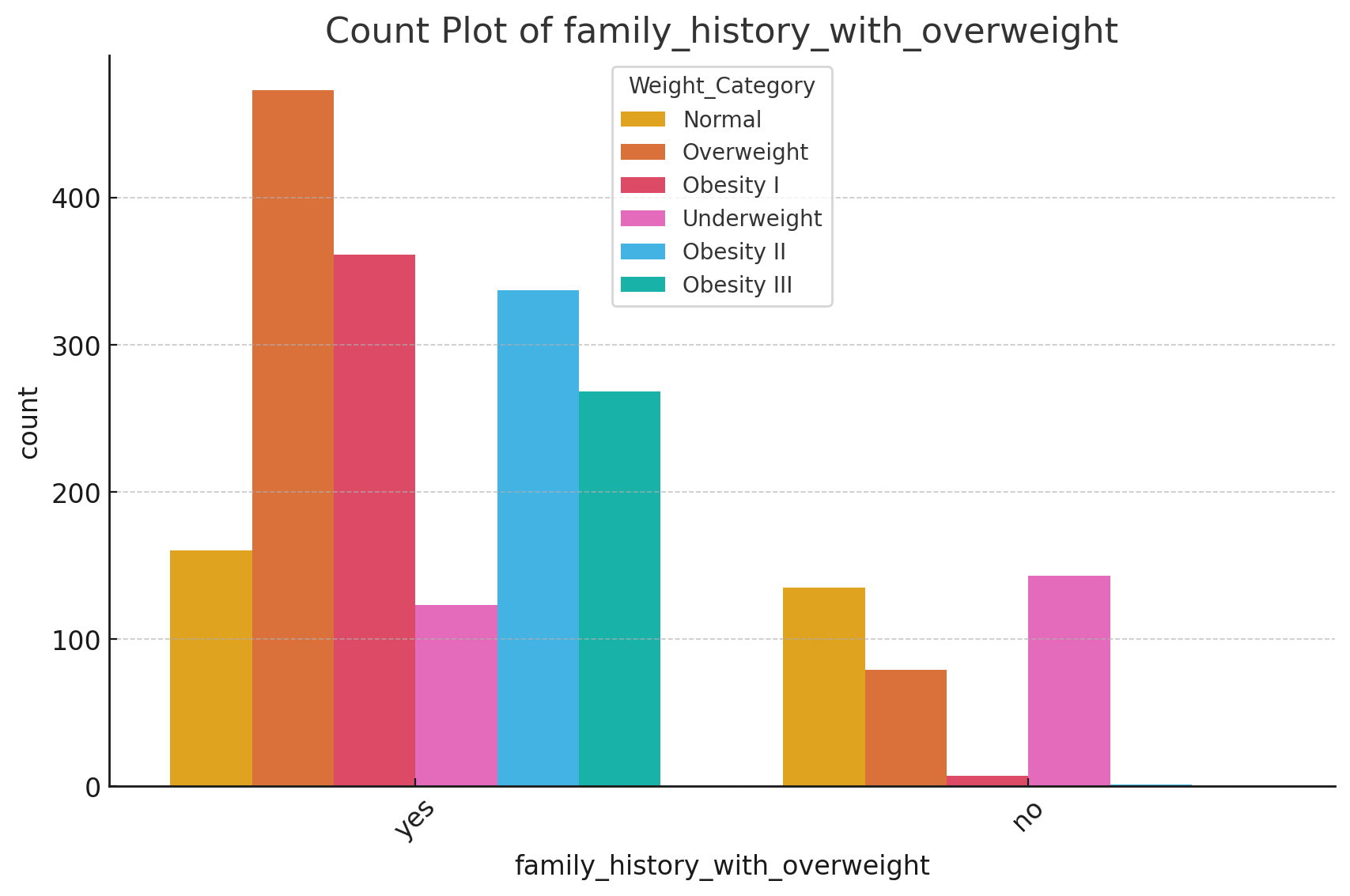
A colorful lines on a black background

Description automatically generated

### Distribution of Weight

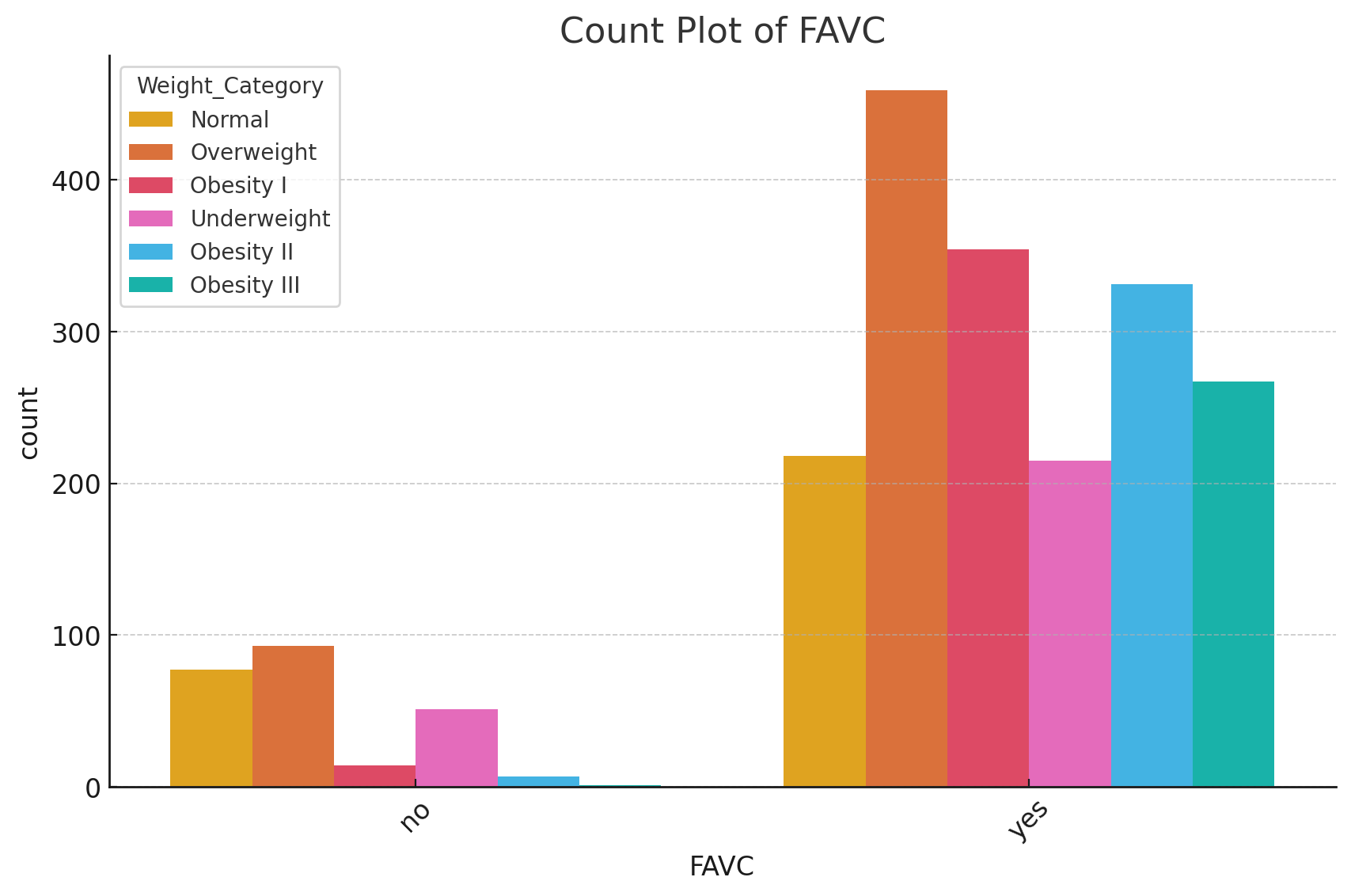
#### Explanation:

* **Distribution Plot:** This plot shows the density distribution of weight across different weight categories.
* **Density Curve:** The curves represent the probability density of the weight data for each weight category. Peaks in the curves indicate the most common weights within each weight category.
* **Color Coding:** Different colors represent different weight categories (e.g., Normal, Overweight, Obesity I, etc.).
* **Insights:**
  + The distribution of weight within each weight category can reveal weight ranges more common in certain weight categories.
  + For example, if a certain weight range has a higher peak for a specific weight category, it suggests a prevalence of that weight category in that weight range.



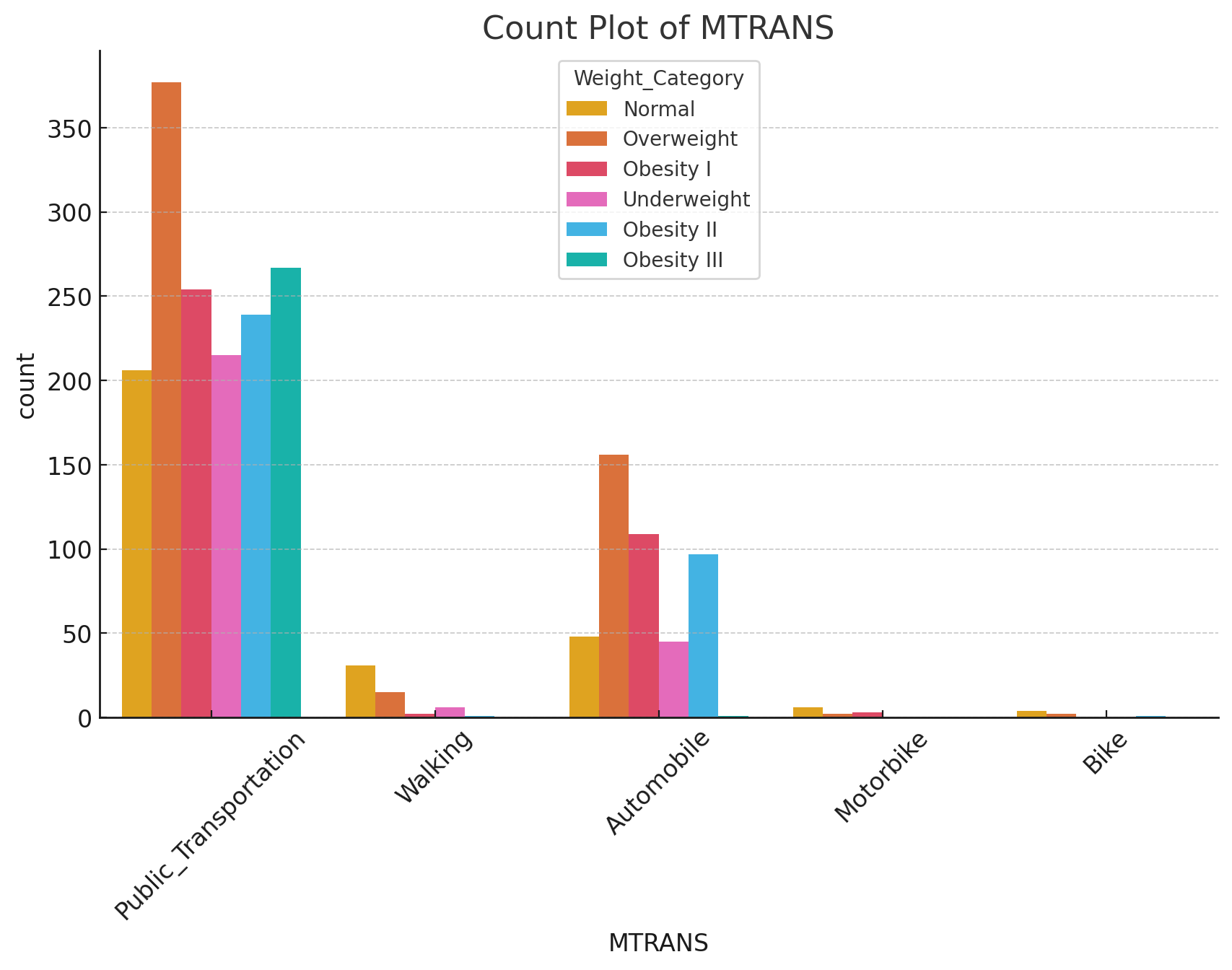
 **Count Plot (Family History of Overweight):** Shows the distribution of individuals with or without a family history of being overweight.

* **Insights:** Helps understand the genetic influence on obesity.



 **Count Plot (FAVC - Frequent Consumption of High Caloric Food):** Displays how often individuals consume high-caloric food across weight categories.

* **Insights:** Highlights dietary habits associated with obesity.



 **Count Plot (MTRANS - Mode of Transportation):** Shows the preferred transportation modes across different weight categories.

* **Insights:** Indicates lifestyle choices related to physical activity levels.

**6. Conclusions**

This analysis highlights the strong influence of both genetic factors and lifestyle choices on obesity levels. Public health interventions focusing on improving dietary habits and increasing physical activity are recommended.