Name: Piyush Bhaskar

Enrolment Number:23116075

Branch: ECE

We use the file insurance.csv which is the **Medical Cost Personal** includes Age, BMI, children and charges etc.

First Read and Check the data and remove the duplicates

```
import pandas as pd
import numpy as np


# First of all we need to read the dataset to get the basic info
df = pd.read_csv("insurance.csv")

# So, this command shows the basic info about the data
print(df.info())
print(df.head())


# Check and remove duplicate records
df.drop_duplicates(inplace=True)
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1338 entries, 0 to 1337
Data columns (total 7 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   age       1338 non-null   int64
 1   sex       1338 non-null   object
 2   bmi       1338 non-null   float64
 3   children  1338 non-null   int64
 4   smoker    1338 non-null   object
 5   region    1338 non-null   object
 6   charges   1338 non-null   float64
dtypes: float64(2), int64(2), object(3)
memory usage: 73.3+ KB
None
   age     sex     bmi  children smoker     region      charges
0   19  female  27.900         0    yes  southwest  16884.92400
1   18    male  33.770         1     no  southeast   1725.55230
2   28    male  33.000         3     no  southeast   4449.46200
3   33    male  22.705         0     no  northwest  21984.47061
4   32    male  28.880         0     no  northwest   3866.85520
```

Now we need to fill the missing values using the mean of the features

```
print(df['bmi'].mean())
print(df['children'].mean())
print(df['charges'].mean())
```

```
30.66345175766642
1.0957367240089753
13279.12148665948
```

Now, we remove the Outliers from the dataset

```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

# Read the dataset
df = pd.read_csv("insurance.csv")

# Remove duplicates
df.drop_duplicates(inplace=True)

# Fill missing values using mean values
df['bmi'].fillna(df['bmi'].mean(), inplace=True)
df['children'].fillna(df['children'].mean(), inplace=True)
df['charges'].fillna(df['charges'].mean(), inplace=True)

# Removing outliers using IQR method
q1 = df[['age', 'bmi', 'children', 'charges']].quantile(0.25)
q3 = df[['age', 'bmi', 'children', 'charges']].quantile(0.75)
IQR = q3 - q1  # Calculate IQR

# Define Lower and Upper Bound
LB = q1 - 1.5 * IQR
UB = q3 + 1.5 * IQR

# Clip outliers instead of removing them
df[['age', 'bmi', 'children', 'charges']] = df[['age', 'bmi', 'children', 'charges']].clip(LB, UB, axis=1)

# Standardizing categorical columns
df[['sex', 'region']] = df[['sex', 'region']].apply(lambda x: x.str.strip().str.title())

# Display dataset info
print(df.info())
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 1192 entries, 0 to 1337
Data columns (total 7 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   age       1192 non-null   int64
 1   sex       1192 non-null   object
 2   bmi       1192 non-null   float64
 3   children  1192 non-null   int64
 4   smoker    1192 non-null   object
 5   region    1192 non-null   object
 6   charges   1192 non-null   float64
dtypes: float64(2), int64(2), object(3)
memory usage: 74.5+ KB
```

Univariate Analysis

Now, let's start the Univariate Analysis

First, we find the summary of the data features

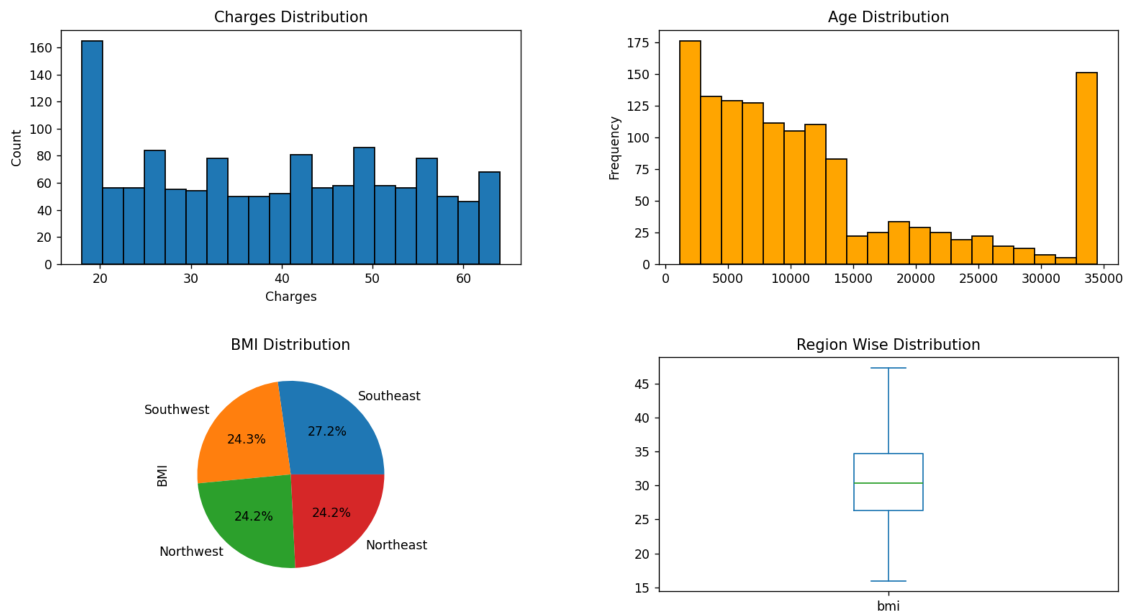
This include the Summary of the Features in the Dataset

```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

# Read the dataset
df = pd.read_csv("insurance.csv")

# Summary statistics for the main features of the data
print(df['age'].describe())
print(df['bmi'].describe())
print(df['charges'].describe())
```

```
count    1338.000000
mean       39.207025
std        14.049960
min        18.000000
25%        27.000000
50%        39.000000
75%        51.000000
max        64.000000
Name: age, dtype: float64
count    1338.000000
mean       30.663397
std         6.098187
min        15.960000
25%        26.296250
50%        30.400000
75%        34.693750
max        53.130000
Name: bmi, dtype: float64
count     1338.000000
mean     13270.422265
std      12110.011237
min       1121.873900
25%       4740.287150
50%       9382.033000
75%      16639.912515
max      63770.428010
Name: charges, dtype: float64
```

Plot of the Data:

```
# Plot data
fig, axes = plt.subplots(2, 2, figsize=[12,12])

# Age Distribution
df['age'].plot(kind='hist', bins=20, edgecolor='black', ax=axes[0,0])
axes[0,1].set_xlabel('Age')
axes[0,1].set_ylabel('Count')
axes[0,1].set_title('Age Distribution')

# Charges Distribution
df['charges'].plot(kind='hist', bins=20, edgecolor='black', color='orange', ax=axes[0,1])
axes[0,0].set_xlabel('Charges')
axes[0,0].set_ylabel('Count')
axes[0,0].set_title('Charges Distribution')

# Region Wise Distribution
df['region'].value_counts().plot(kind='pie', autopct='%0.1f%%', ax=axes[1,0])
axes[1,1].set_ylabel('')
axes[1,1].set_title('Region Wise Distribution')

# Box Plot After Outlier Removal
df['bmi'].plot(kind='box', ax=axes[1,1])
axes[1,0].set_ylabel('BMI')
axes[1,0].set_title('BMI Distribution')

# Adjust layout
plt.tight_layout(pad=3.0)
plt.subplots_adjust(hspace=0.4, wspace=0.3)

# Show the plots
plt.show()
```



Outcomes:

- Maximum number of people are from nearly less than 20 years.
- Maximum Number of people have charges less than 5000.
- Mostly people are from Southeast

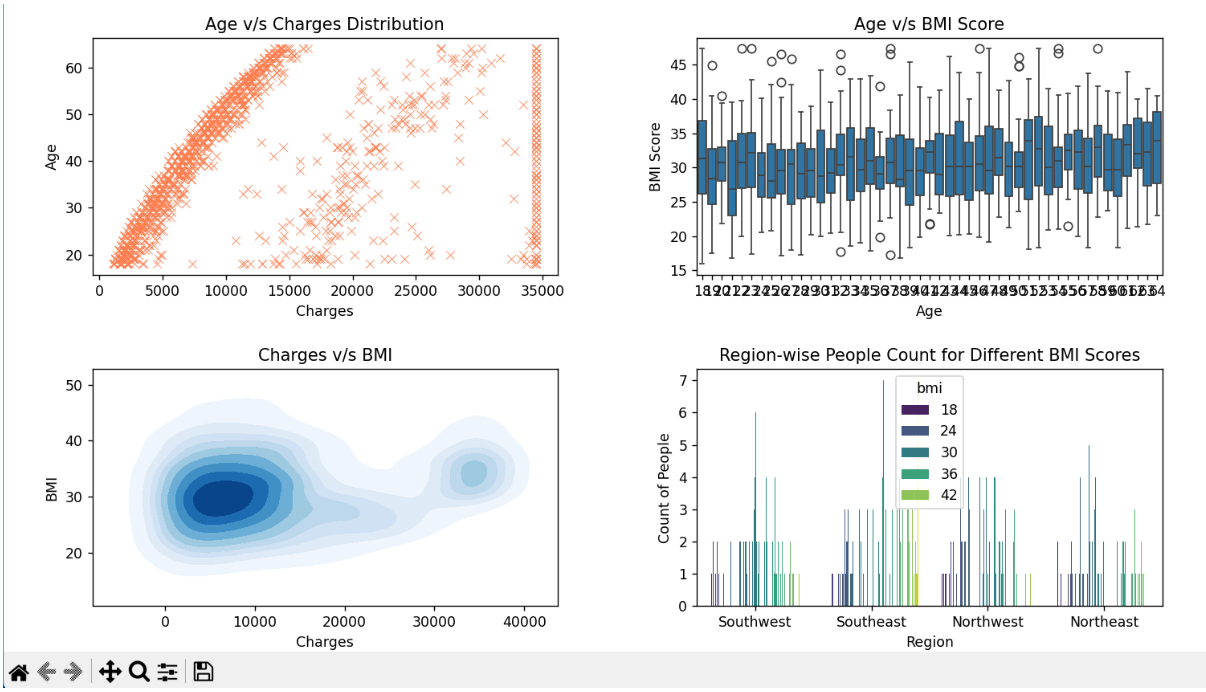
BIVARIATE ANALYSIS

```
import seaborn as sns
# Correlation Matrix for Age, Charges and BMI
cor_matrix=df[['age','charges','bmi']].corr()
print(cor_matrix)
```

```
              age   charges       bmi
age      1.000000  0.312423  0.111998
charges  0.312423  1.000000  0.161220
bmi      0.111998  0.161220  1.000000
```

```
fig1, axes= plt.subplots(2, 2, figsize=(12,12) )

# Scatar Plot between Charges and Age
sns.scatterplot(x=df['charges'],y=df['age'], marker= 'x', color='coral',ax=axes[0,0])
axes[0,0].set_xlabel("Charges")
axes[0,0].set_ylabel("Age")
axes[0,0].set_title("Age v/s Charges Distribution")

# Box Plot of performance score at different ages
sns.boxplot(x=df['age'],y=df['bmi'],ax=axes[0,1])
axes[0,1].set_xlabel("Age")
axes[0,1].set_ylabel("BMI Score")
axes[0,1].set_title("Age v/s BMI Score")

# Density graph of performance score at differnet salary
sns.kdeplot(x=df['charges'], y=df['bmi'], cmap="Blues", fill= True,ax=axes[1,0])
axes[1,0].set_xlabel("Charges")
axes[1,0].set_ylabel("BMI")
axes[1,0].set_title("Charges v/s BMI")

# Count of people in different department with different BMI
sns.countplot(x=df['region'], hue=df['bmi'], palette='viridis', ax=axes[1,1])
axes[1,1].set_xlabel("Region")
axes[1,1].set_ylabel("Count of People")
axes[1,1].set_title("Region-wise People Count for Different BMI Scores")

# Adjust layout
plt.tight_layout(pad=3.0)
plt.subplots_adjust(hspace=0.4, wspace=0.3)

# Plot the data
plt.show()
```

- **In this we plot the Scatter Plot for the Charges and Age**
- **Boxplot for Age and BMI**
- **Density Graph for the Charges and BMI**
- **Count Plot for Region wise Count**

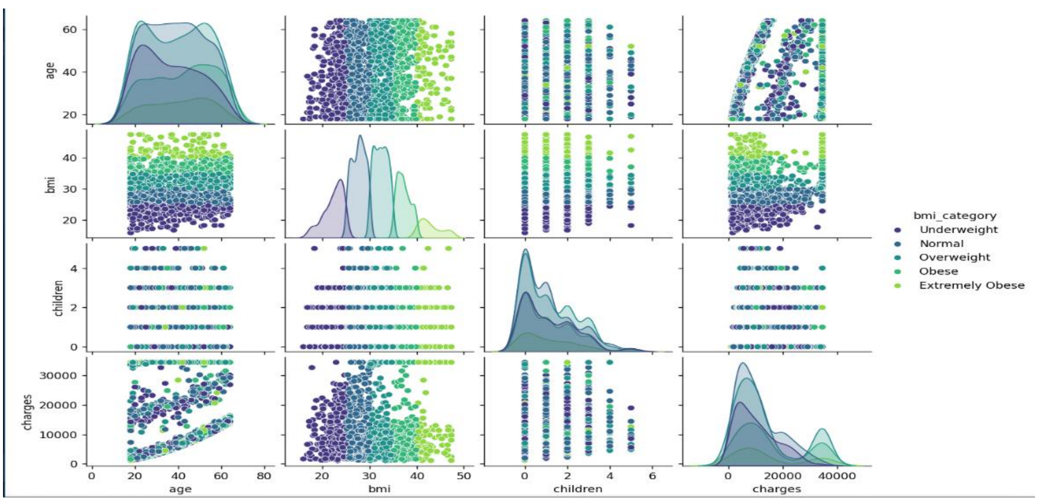


Outcomes:

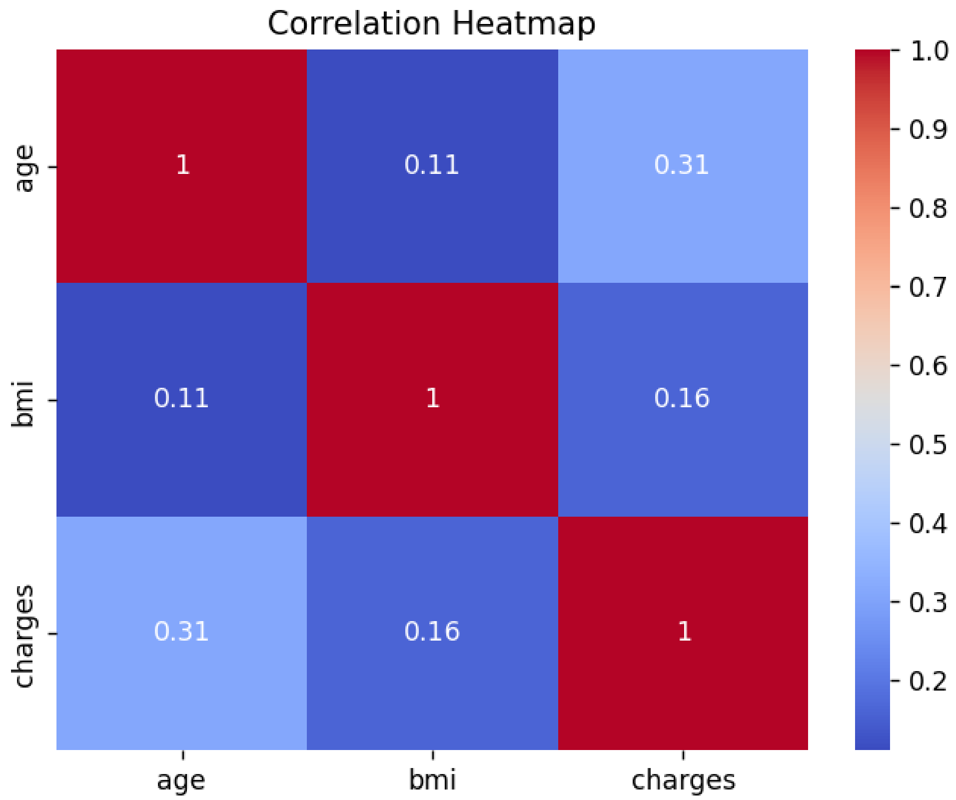
- **Older individuals tend to have higher medical expenses.**
- **BMI remains constant across ages but has outliers.**
- **Higher BMI is linked to higher medical charges.**
- **BMI distribution is similar across all regions.**

Multivariate Analysis.

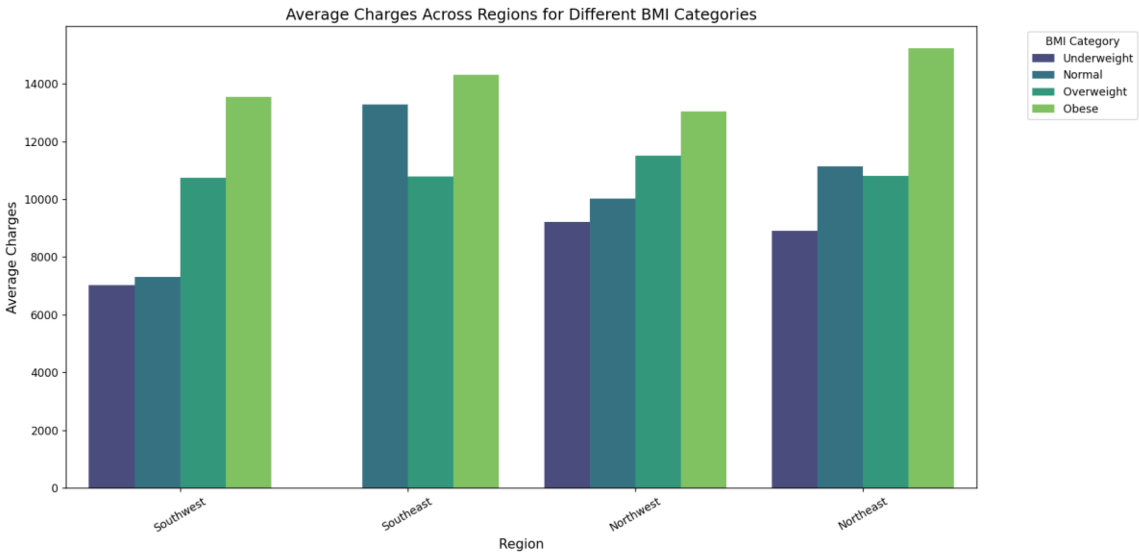
```
# Convert BMI into categorical bins
df['bmi_category'] = pd.cut(df['bmi'], bins=[15, 25, 30, 35, 40, 50], labels=['Underweight', 'Normal', 'Overweight', 'Obese', 'Extremely Obese'])

# Plot with categorical hue
sns.pairplot(df, hue='bmi_category', palette='viridis')

plt.show()
```



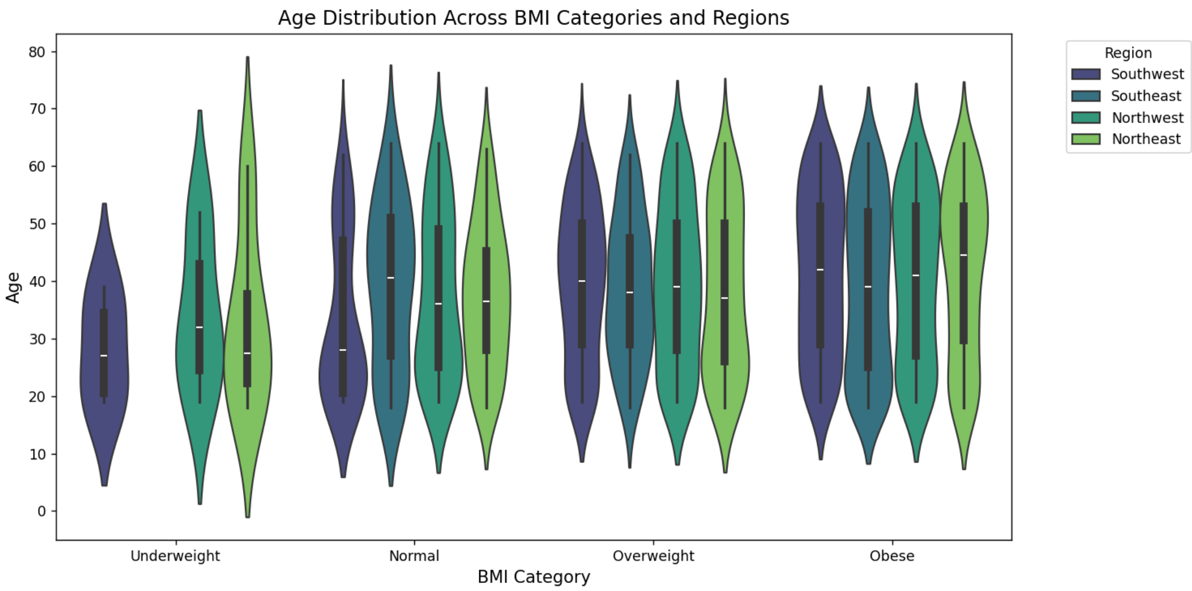
```
# Plot heatmap
sns.heatmap(df[['age','bmi','charges']].corr(), cmap='coolwarm', annot=True)

# Add title
plt.title("Correlation Heatmap")

# Show plot
plt.show()
```




```
# Define BMI categories
bins = [0, 18.5, 24.9, 29.9, 50]
labels = ["Underweight", "Normal", "Overweight", "Obese"]
df["bmi_category"] = pd.cut(df["bmi"], bins=bins, labels=labels)

# Create a barplot with a bigger figure size
plt.figure(figsize=(12, 6))  # Increase width and height for better fit
sns.barplot(x="region", y="charges", hue="bmi_category", data=df, palette="viridis", ci=None)

# Customize the plot
plt.title("Average Charges Across Regions for Different BMI Categories", fontsize=14)
plt.xlabel("Region", fontsize=12)
plt.ylabel("Average Charges", fontsize=12)
plt.xticks(rotation=30)  # Slight rotation for readability
plt.legend(title="BMI Category", bbox_to_anchor=(1.05, 1), loc='upper left')

# Adjust layout to fit everything properly
plt.tight_layout()

# Show plot
plt.show()
```




```python
# Ensure BMI is categorized if it's continuous
bins = [0, 18.5, 24.9, 29.9, 50]
labels = ["Underweight", "Normal", "Overweight", "Obese"]
df["bmi_category"] = pd.cut(df["bmi"], bins=bins, labels=labels)

# Increase figure size for clarity
plt.figure(figsize=(12, 6))

# Violin plot with categorized BMI
sns.violinplot(x="bmi_category", y="age", hue="region", data=df, palette="viridis", scale="width")

# Titles and Labels
plt.title("Age Distribution Across BMI Categories and Regions", fontsize=14)
plt.xlabel("BMI Category", fontsize=12)
plt.ylabel("Age", fontsize=12)

# Adjust Legend Position
plt.legend(title="Region", bbox_to_anchor=(1.05, 1), loc='upper left')

# Improve layout to fit all elements
plt.tight_layout()
plt.show()
```



Multivariate Analysis Summary

- **Pair Plot** – Visualized interactions between numerical variables based on performance scores.
- **Heatmap** – Displayed the correlation matrix, highlighting relationships between numerical variables.
- **Grouped Bar Chart** – Analysed salary distribution across departments and performance scores, showcasing pay variations across different categories.
- **Violin Plot** – Demonstrated significant variations in age distribution across different performance scores.

This analysis provided valuable insights into the relationships between key factors.

Final Conclusions

- **Univariate Analysis:**
    - Majority of people are below 20 years old.
    - Most people have medical charges below 5000.
    - Southeast has the highest number of people.
- **Bivariate Analysis:**
    - Older individuals have higher medical costs.
    - BMI remains constant across ages but has outliers.
    - Higher BMI leads to higher medical charges.
    - BMI distribution is similar across all regions.
- **Multivariate Analysis:**
    - **Pair Plot:** Showed relationships between numerical variables.
    - **Heatmap:** Highlighted correlations.
    - **Grouped Bar Chart:** Showed salary variations across categories.
    - **Violin Plot:** Demonstrated variations in age distribution.