NAAN MUDHALVAN-IBM DATA ANALYTICS WITH COGNOS

PUBLIC HEALTH AWARENESS CAMPAIGN ANALYSIS.

Team Members:

NAME	ROLL.NO	E-Mail ID
M SHYAM KUMAR	2021115107	*shyamchess.murali@gmail.com*
S SIVA GANESH	2021115108	*ganeshsiva7425@gmail.com*
S R SUBASHREE	2021115110	*subashreesundararajan@gmail.com*
K SUMATHI	2021115112	*sumathikarthikeyan45@gmail.com*
S YUKESH	2021115331	*yukiiyukesh@gmail.com*

PROJECT DESCRIPTION :

The "Public Health Awareness Campaign Analysis" project aims to assess the effectiveness of various public health awareness campaigns in reaching their target audiences and increasing awareness on critical health issues. In an era marked by rapid information dissemination and evolving communication channels, understanding the impact of awareness campaigns is vital for informed decision-making and resource allocation.

PROVIDED KAGGLE DATASET:

https://www.kaggle.com/datasets/osmi/mental-health-in-tech-survey

ABSTRACT :

The "Public Health Awareness Campaign Analysis" project aims to assess the effectiveness of various public health awareness campaigns in reaching their target audiences and increasing awareness on critical health issues. Leveraging a comprehensive dataset comprising campaign objectives, media channels, demographics, and performance metrics, the study employs data analysis techniques, including data visualization,

statistical analysis, machine learning, and natural language processing. By identifying effective strategies, target demographics, and content patterns, this project seeks to inform future campaigns and contribute to the broader goal of promoting public health and well-being through data-driven decision-making, ultimately leading to improved health outcomes.This research endeavors to harness the power of data analysis to measure the comprehensive impact of public health campaigns, offering valuable insights to policymakers and public health organizations, thus advancing the efficiency and effectiveness of public health awareness efforts.

I. Project Definition

A. Overview

The project revolves around the analysis of data from public health awareness campaigns with the primary aim of gauging their effectiveness in reaching the target audience and elevating awareness levels. The ultimate goal is to derive insights that will not only assess the impact of the campaigns but will also serve as a foundation for shaping future strategies in this domain.

B. Objectives

1. Audience Reach: Measure the extent to which the campaigns are reaching the intended audience.

2. Awareness Levels: Evaluate the effectiveness of campaigns in increasing awareness regarding public health issues.

3. Campaign Impact: Assess the overall impact of the campaigns on the target audience.

C. Scope

The project encompasses defining clear analysis objectives, collecting comprehensive campaign data, designing insightful visualizations using IBM Cognos, and incorporating code for data analysis where deemed beneficial.

II. Design Thinking

A. Analysis Objectives

1. Audience Reach:

Define metrics and criteria to quantify the reach of public health awareness campaigns. This may include social media engagement, website visits, and geographical reach.

2. Awareness Levels:

Establish key performance indicators (KPIs) to measure changes in awareness levels, considering factors like survey responses, keyword analysis, and media coverage.

3. Campaign Impact:

Develop a framework for assessing the overall impact of campaigns, incorporating both quantitative and qualitative measures. This could involve sentiment analysis, behavior change metrics, and expert evaluations.

B. Data Collection

1. Sources:

Identify the primary sources of campaign data, encompassing social media analytics, website analytics, survey responses, and any relevant third-party data repositories.

2. Methods:

Define the methods for data collection, considering real-time monitoring, periodic surveys, and collaboration with external partners for enriched datasets.

C. Visualization Strategy

1. IBM Cognos:

Plan the utilization of IBM Cognos for creating visually compelling dashboards and reports. Consider the audience and tailor visualizations to effectively communicate key insights.

2. Dashboard Elements:

Outline the key elements of the dashboards, including interactive charts, trend analyses, and comparative visualizations, to present a holistic view of campaign performance.

D. Code Integration

1. Data Cleaning

Identify areas in the data preprocessing phase where code integration can enhance efficiency, ensuring that the data is clean and ready for analysis.

2. Transformation

Specify instances where code can facilitate complex data transformations, making it more amenable for visualization and interpretation.

3. Statistical Analysis

Determine the statistical analyses that can be performed using code to extract deeper insights, such as correlation studies or predictive modeling.

MACHINE LEARNING ALGORITHMS:

Natural Language Processing (NLP):

Sentiment Analysis: NLP can be used to analyze social media and news data to gauge public sentiment towards health campaigns.

Topic Modeling:

Discover the key topics and themes being discussed in public conversations, helping you tailor campaigns to address relevant issues.

Supervised Learning Algorithms:

Classification Models: Use classification algorithms such as logistic regression, decision trees, or random forests to predict the success of campaigns based on historical data.

Recommendation Systems:

Develop recommendation systems to suggest personalized health campaigns based on individual characteristics and preferences.

Time Series Analysis:

Forecasting Models: Time series analysis can be used to predict trends and patterns in public health data, allowing for better planning of awareness campaigns.

Geospatial Analysis:

Spatial Clustering: Use clustering algorithms to identify geographic areas with higher health-related needs, enabling targeted campaigns in those regions.

Geospatial Visualization:

Utilize geographic information systems (GIS) and data visualization to map the spread of diseases and campaign impact.

Anomaly Detection:

Identify unusual patterns or unexpected deviations in public health data, which could indicate the need for immediate action or a revised campaign strategy.

DEVELOPMENT – 1

STEP 1 : IMPORT LIBRARIES

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix
from sklearn.preprocessing import LabelEncoder
from sklearn.impute import SimpleImputer
```

In this step, the necessary libraries are imported to work with data, visualize it, and build a machine learning model. These libraries include pandas for data manipulation, matplotlib and seaborn for data visualization, and scikit-learn for machine learning.

STEP 2 : LOAD DATA

```
f = "survey.csv"
df = pd.read_csv(f)
```

This step loads a dataset from a CSV file named "survey.csv" into a

pandas DataFrame called 'df'.

STEP 3 : DISPLAY BASIC INFORMATION ABOUT THE DATASET

```
df.info()
```

This step provides an overview of the dataset's structure, including the number of rows, columns, data types, and information about missing values.

STEP 4 : HANDLE MISSING VALUES

```
df = df.drop(columns=['comments'])
imputer = SimpleImputer(strategy='median')
df['Age'] = imputer.fit_transform(df[['Age']])
```

This step first drops the 'comments' column as it's considered not useful for analysis. Then, it handles missing values in the 'Age' column by filling them with the median value of the 'Age' column.

STEP 5 : ENCODE BINARY COLUMNS

```
binary_cols = ['self_employed', 'family_history', 'treatment', 'remote_work', 'tech_company', 'benefits',
        'wellness_program', 'seek_help', 'anonymity', 'mental_health_interview', 'phys_health_interview',
        'mental_vs_physical', 'obs_consequence']

for col in binary_cols:
   df[col] = LabelEncoder().fit_transform(df[col])
```

This step encodes binary columns with '0' and '1' values using LabelEncoder. These columns typically contain 'Yes' or 'No' responses, and they are transformed into numerical values for analysis.

STEP 6 : DATA VISUALIZATION

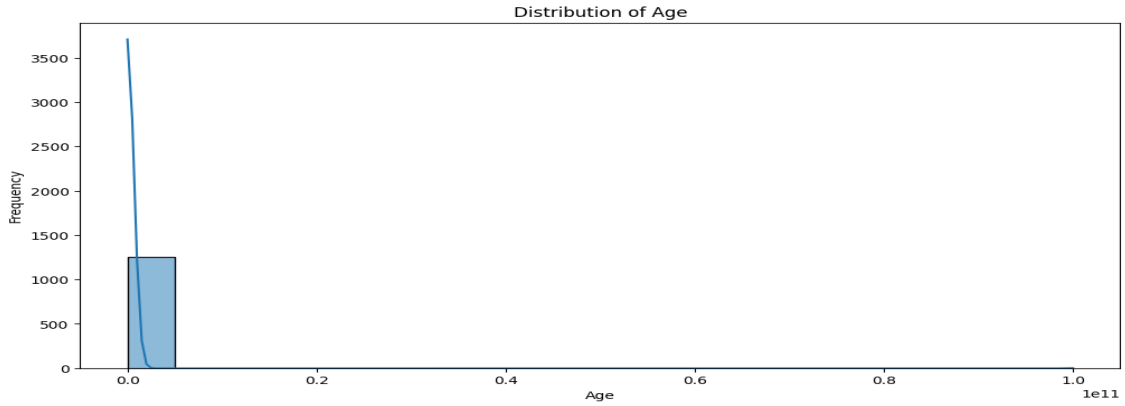
This section includes various data visualization steps using matplotlib and seaborn for understanding the dataset.

Distribution of Age

```
plt.figure(figsize=(12, 6))
sns.histplot(df['Age'], bins=20, kde=True)
plt.title('Distribution of Age')
```

```
plt.xlabel('Age')
plt.ylabel('Frequency')
plt.show()
```

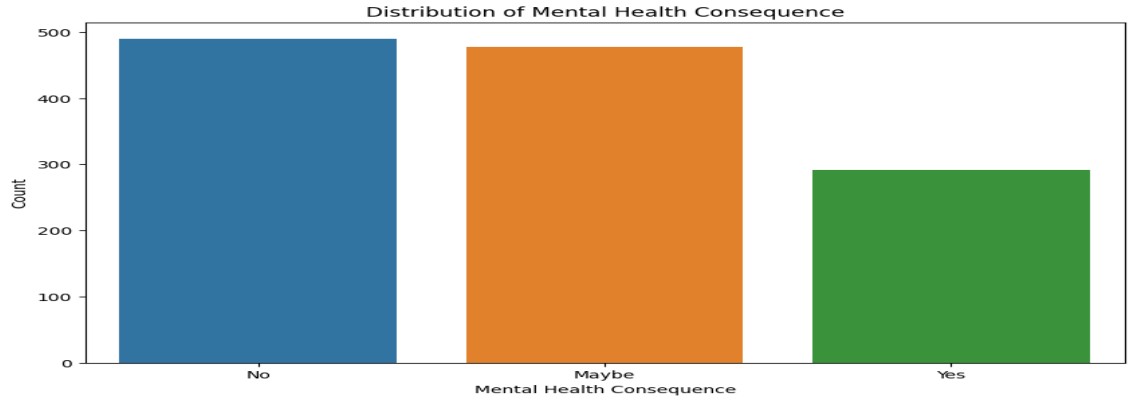
This step creates a histogram and kernel density plot to visualize the distribution of ages in the dataset.



Distribution of Mental Health Consequences

```
plt.figure(figsize=(10, 6))
sns.countplot(x='mental_health_consequence', data=df)
plt.title('Distribution of Mental Health Consequence')
plt.xlabel('Mental Health Consequence')
plt.ylabel('Count')
plt.show()
```

This step creates a count plot to show the distribution of the 'mental_health_consequence' column.

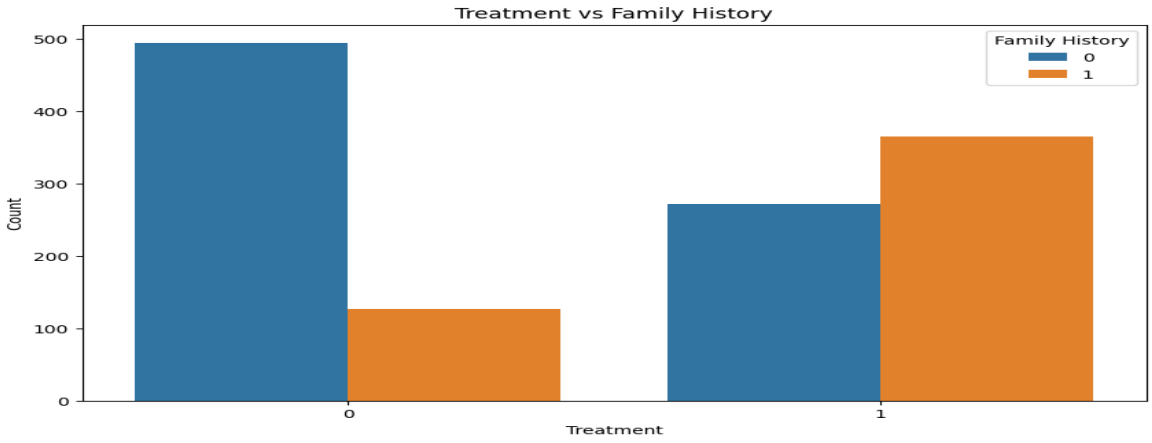


Relationship between Seeking Treatment and Family History

```
plt.figure(figsize=(10, 6))
sns.countplot(x='treatment', hue='family_history', data=df)
plt.title('Treatment vs Family History')
plt.xlabel('Treatment')
```

```
plt.ylabel('Count')
plt.legend(title='Family History', loc='upper right')
plt.show()
```

This step creates a count plot to visualize the relationship between seeking treatment and family history.



Distribution of Mental Health Consequence Based on Company Size

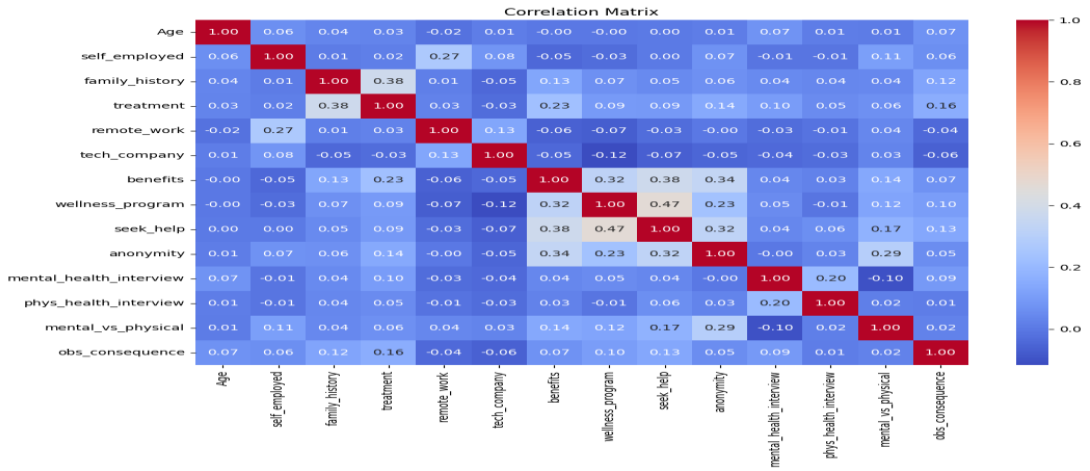
```
plt.figure(figsize=(12, 6))
sns.countplot(x='no_employees', hue='mental_health_consequence', data
df)
plt.title('Mental Health Consequence vs Company Size')
plt.xlabel('Number of Employees')
plt.ylabel('Count')
plt.legend(title='Mental Health Consequence', loc='upper right')
plt.show()
```

This step creates a count plot to show the distribution of mental health consequences based on company size.

Correlation Matrix

```
correlation_matrix = df.corr()
plt.figure(figsize=(12, 8))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', fmt=".2f")
plt.title('Correlation Matrix')
plt.show()
```

This step generates a heatmap to visualize the correlation between numerical variables in the dataset.



STEP 7 : DATA PREPROCESSING

```
# Drop unnecessary columns
df = df.drop(['Timestamp'], axis=1)  # Timestamp is non-numerical
# Split the data into features (X) and target variable (y)
X = df.drop(['treatment'], axis=1)
```

```
y = df['treatment']

# Encode categorical variables using one-hot encoding
X = pd.get_dummies(X, drop_first=True)

# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

In this step, the code drops the 'Timestamp' column, which is considered non-numerical.It splits the data into features (X) and the target variable (y) and encodes categorical variables using one-hot encoding.
The data is then split into training and testing sets.

STEP 8 : TRAIN A LOGISTIC REGRESSION MODEL

```
model = LogisticRegression(random_state=42)
model.fit(X_train, y_train)
```

This step trains a Logistic Regression model on the training data.

STEP 9 : MAKE PREDICTIONS AND EVALUATE THE MODEL

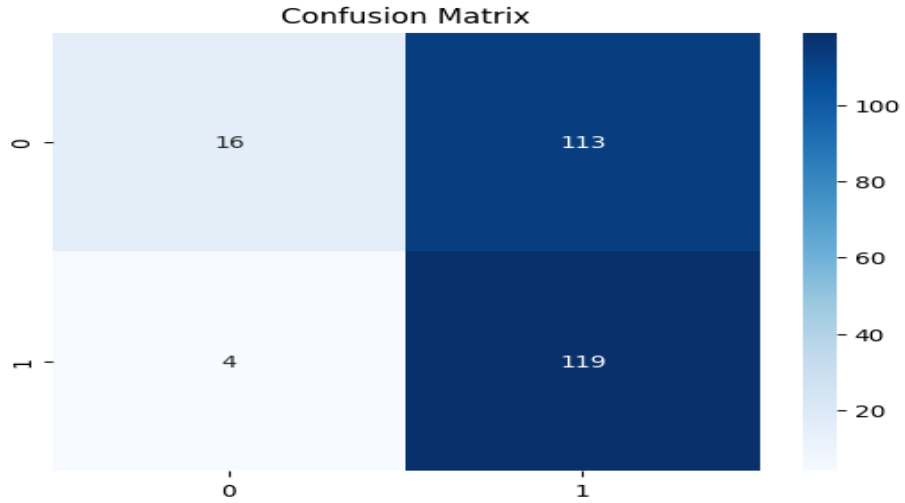
```
y_pred = model.predict(X_test)

# Evaluate the accuracy
accuracy = accuracy_score(y_test, y_pred)
print(f"Accuracy: {accuracy:.2f}")
```

This step makes predictions on the test set and calculates the accuracy of the model.

STEP 10 : DISPLAY CLASSIFICATION REPORT AND CONFUSION MATRIX

```
print("\nClassification Report:")
print(classification_report(y_test, y_pred))
print("\nConfusion Matrix:")
conf_matrix = confusion_matrix(y_test, y_pred)
sns.heatmap(conf_matrix, annot=True, fmt="d", cmap="Blues")
plt.title("Confusion Matrix")
plt.show()
```

This step displays a classification report and a confusion matrix to evaluate the performance of the logistic regression model. The classification report provides details on precision, recall, F1-score, and support for each class, while the confusion matrix visually represents the model's performance.



DEVELOPMENT – 2

STEP 1 : DATA SUMMARIZATION

```
desc_stats = df.describe()
print(desc_stats)
```

This step provides a condensed overview of numerical data in the project. It helps assess data quality, offers initial insights into data characteristics, supports decision-making, and facilitates clear communication of key findings..

STEP 2 : CREATE NEW COLUMN

```
df['Grouped_Gender'] = df['Gender'].apply(lambda x: 'Others' if x not in ['Male', 'Female', 'Non-binary'] else x)
```

This step creates a new column which can help in having more visually appealing and easier to understand charts and graphs..

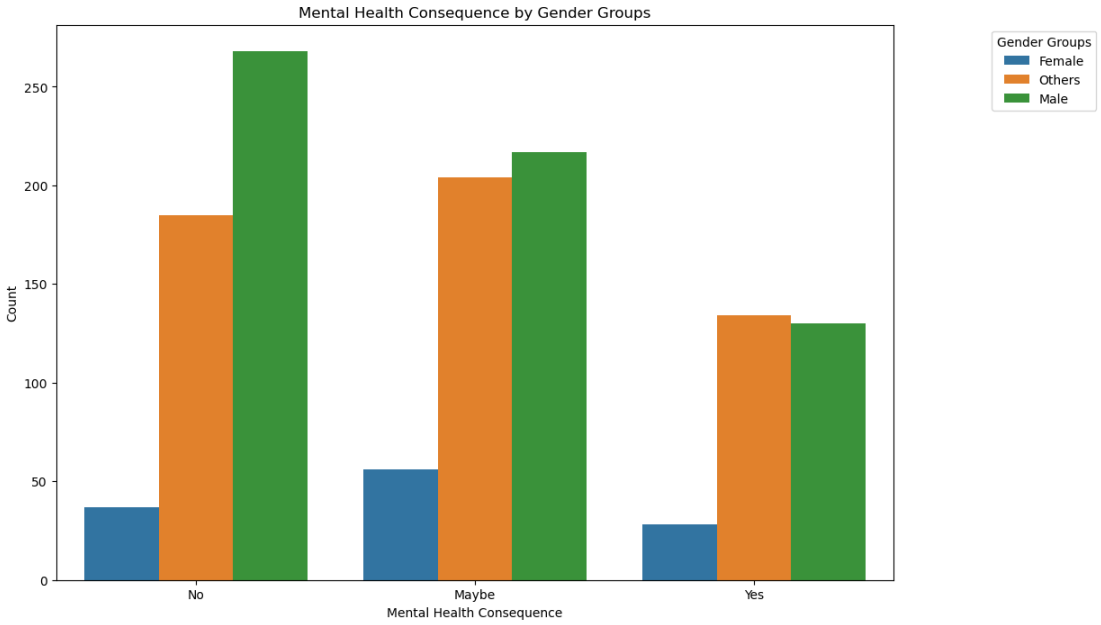
STEP 3 : DATA VISUALIZATION

This section includes various data visualization steps using matplotlib and seaborn for understanding the dataset.

Mental Health Consequence by Gender Groups

```
plt.figure(figsize=(12, 8))
sns.countplot(x='mental_health_consequence', hue='Grouped_Gender', data=df)
plt.title('Mental Health Consequence by Gender Groups')
plt.xlabel('Mental Health Consequence')
plt.ylabel('Count')
plt.legend(title='GenderGroups',loc='upperright', bbox_to_anchor=(1.25, 1))
plt.show()
```

This step creates a bar graph to visualize the distribution of Mental Health Consequence by Gender Groups.

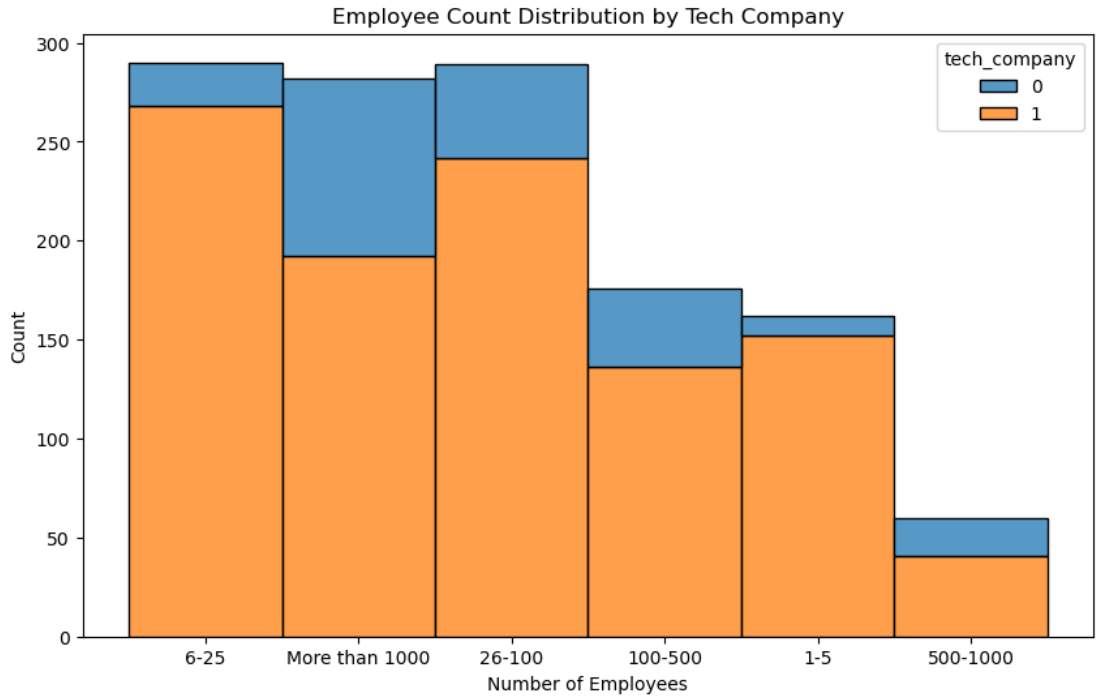


Employee Count Distribution by Tech Company

```
plt.figure(figsize=(10, 6))
sns.histplot(data=df, x='no_employees', hue='tech_company', multiple='stack')
plt.title('Employee Count Distribution by Tech Company')
```

```
plt.xlabel('Number of Employees')
plt.ylabel('Count')
plt.show()
```

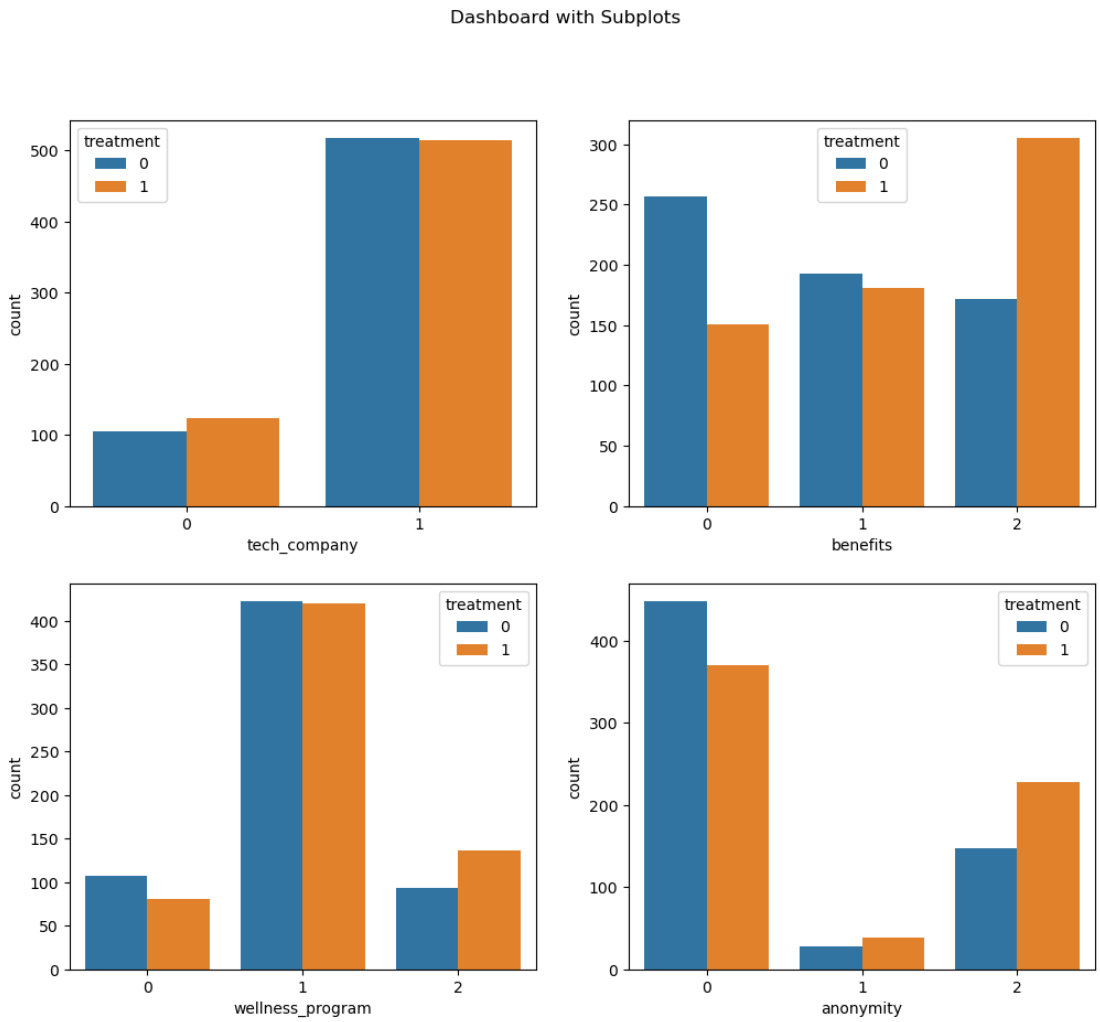
This step creates a histogram to show the Employee Count Distribution by Tech Company.



Dashboard with Subplots

```
fig, axs = plt.subplots(2, 2, figsize=(12, 10))
sns.countplot(x='tech_company', hue='treatment', data=df, ax=axs[0, 0])
sns.countplot(x='benefits', hue='treatment', data=df, ax=axs[0, 1])
sns.countplot(x='wellness_program', hue='treatment', data=df, ax=axs[1, 0])
sns.countplot(x='anonymity', hue='treatment', data=df, ax=axs[1, 1])
plt.suptitle('Dashboard with Subplots')
plt.show()
```

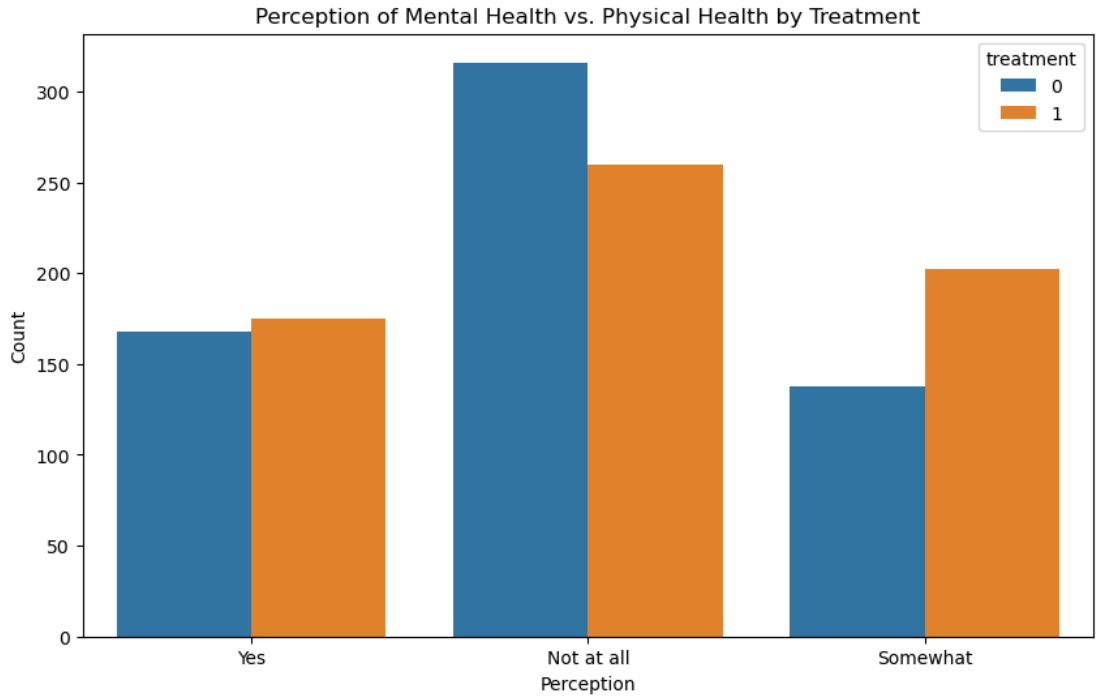
This step creates a Dashboard with Subplots of count plots for distribution of treatment based on tech company, benefits , wellness_program and anonymity.



Disparate Analysis between Mental Health and Perceived Physical Health

```
perception_labels = {0: 'Not at all', 1: 'Somewhat', 2: 'Yes'}
df['Mental_vs_Physical_Label']=df['mental_vs_physical']
.map(perception_labels)
plt.figure(figsize=(10, 6))
sns.countplot(x='Mental_vs_Physical_Label', data=df, hue='treatment')
plt.title('Perception of Mental Health vs. Physical Health by Treatment')
plt.xlabel('Perception')
plt.ylabel('Count')
plt.show()
```

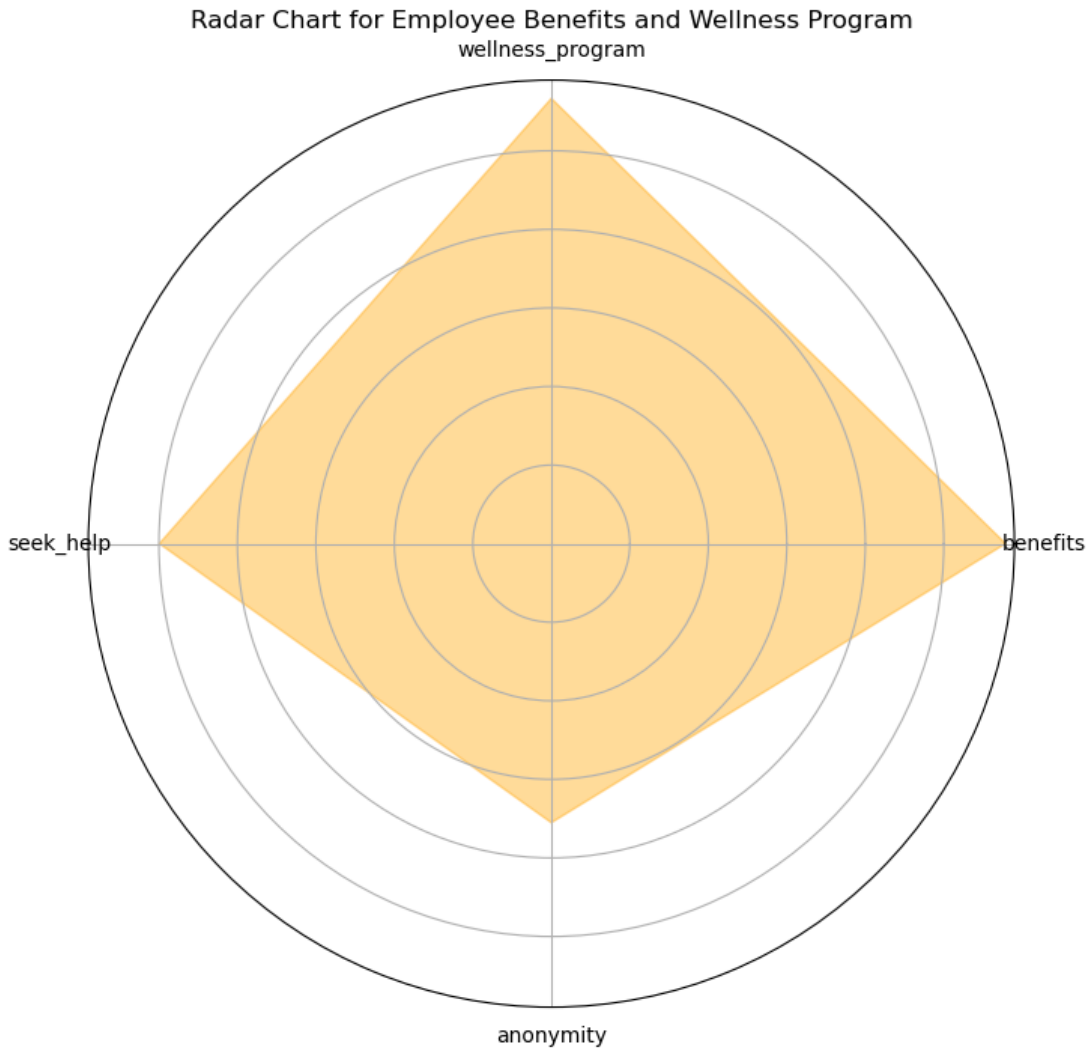
This step creates a bar graph to show the comparison between mental health and perceived physical health.



Employee Benefits and Wellness Program Evaluation by Dimension

```
benefits_cols = ['benefits', 'wellness_program', 'seek_help', 'anonymity']
benefits_values = df[benefits_cols].mean()
benefits_values = benefits_values / benefits_values.sum()
plt.figure(figsize=(8, 8))
angles = [n / len(benefits_cols) * 2 * np.pi for n in range(len(benefits_cols))]
ax = plt.subplot(111, polar=True)
ax.fill(angles, benefits_values, color='orange', alpha=0.4)
ax.set_yticklabels([])
ax.set_xticks(angles)
ax.set_xticklabels(benefits_cols)
plt.title('Radar Chart for Employee Benefits and Wellness Program')
plt.show()
```

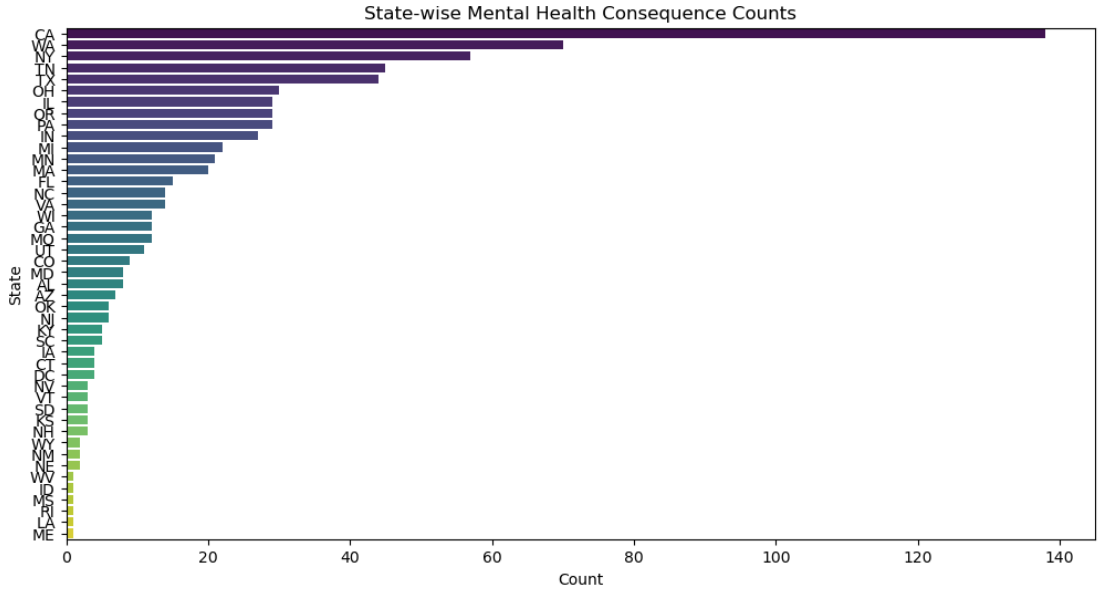
This step generates a radar chart to visualize the correlation between employee benefits and wellness program.



Distribution of State-wise Mental Health Consequence

```
state_counts = df['state'].value_counts()
plt.figure(figsize=(12, 6))
sns.barplot(x=state_counts, y=state_counts.index, palette='viridis')
plt.title('State-wise Mental Health Consequence Counts')
plt.xlabel('Count')
plt.ylabel('State')
plt.show()
```

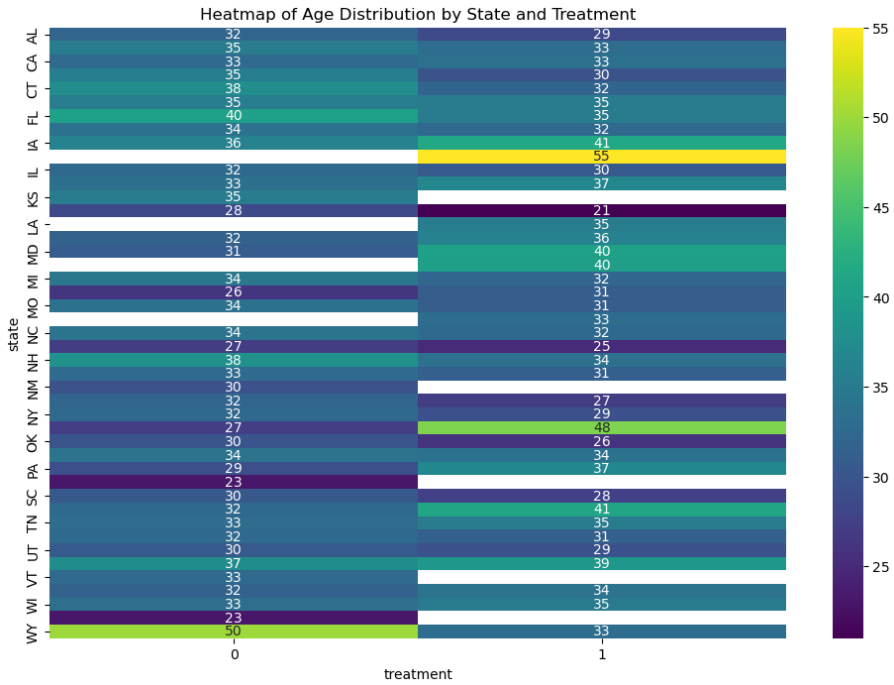
This step creates a Horizontal Bar Chart to visualize the State-wise Mental Health Consequence Counts.



Distribution of Age by State and Treatment

```
state_age_pivot = df.pivot_table(values='Age', index='state', columns='treatment', aggfunc='mean')
plt.figure(figsize=(12, 8))
sns.heatmap(state_age_pivot, annot=True, cmap='viridis')
plt.title('Heatmap of Age Distribution by State and Treatment')
plt.show()
```

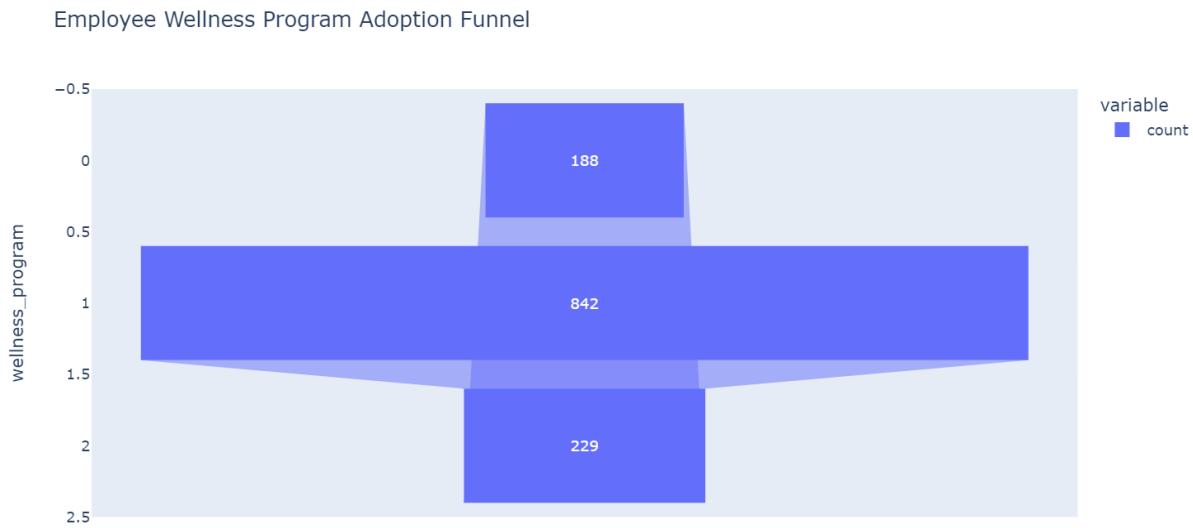
This step creates a heatmap to show the Age Distribution by State and Treatment.



Progress of Employee Wellness Program Enrollment

```
import plotly.express as px
wellness_program_counts = df['wellness_program'].value_counts()
fig = px.funnel(wellness_program_counts, title='Employee Wellness Program Adoption Funnel', labels={'index': 'Adoption Status', 'value': 'Count'})
fig.show()
```

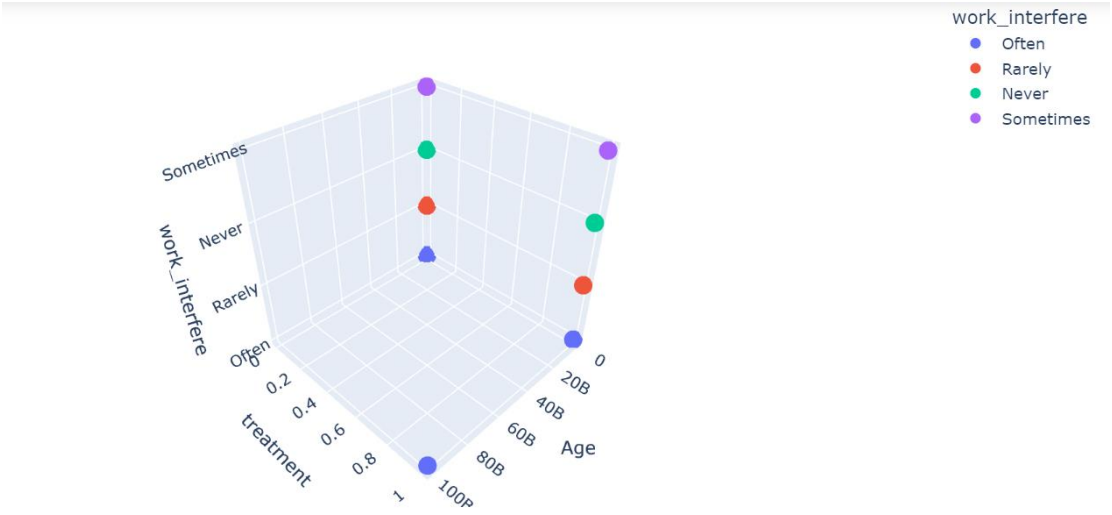
This step creates a Funnel Chart for Employee Wellness Program Adoption by importing plotly.express .



Distribution between Age, Treatment, and Work Interference using 3D

```
fig = px.scatter_3d(df, x='Age', y='treatment', z='work_interfere', color='work_interfere')
fig.update_layout(title='3D Scatter Plot')
fig.show()
```

This step creates a 3D scatter plot to show the distribution between Age, treatment and work interference.



STEP 7 : PERFORMING STATISTICAL TEST

```
from scipy.stats import ttest_ind
t_stat, p_value = ttest_ind(df[df['treatment'] == 1]['Age'], df [df ['treatment'] == 0]['Age'])
print(f"T-Statistic: {t_stat}\nP-value: {p_value}")
```

OUTPUT:
T-Statistic: 0.9881466556871645
P-value: 0.32327099499937273

In this step, we perform an independent two-sample t-test. It calculates the t-statistic and p-value to assess whether there is a statistically significant difference in the ages of two groups: the treatment group and the control group, as represented in a DataFrame (df).

CONCLUSION

This report presents a comprehensive analysis of a public health awareness campaign using data analysis techniques and machine learning. Here is a conclusion based on the information provided:

1.Project Overview:

The project's primary goal is to assess the effectiveness of public health awareness campaigns in reaching their target audience and increasing awareness on critical health issues. The project employs a wide range of data analysis techniques, including data visualization, statistical analysis, machine learning, and natural language processing.

2. Data Preprocessing:

The initial steps involve importing necessary libraries, loading the dataset, and performing data preprocessing. Data cleaning, handling missing values, and encoding binary columns ensure that the data is ready for analysis.

3. Data Visualization:

Data visualization is a crucial aspect of this project, helping to understand the dataset better. Several visualizations are presented, including the distribution of age, mental health consequences, the relationship between seeking treatment and family history, and more. These visualizations provide insights into various aspects of the data.

4. Machine Learning Model:

A logistic regression model is trained to predict whether individuals seek treatment for mental health issues based on various features. The accuracy of the model is calculated to evaluate its performance.

5. Data Summarization:

A summary of numerical data in the project is presented, offering a condensed overview of key statistics. This provides insights into data characteristics and quality.

6. New Feature Creation:

A new feature called 'Grouped_Gender' is introduced to make data visualization more visually appealing and easier to understand.

7. Further Data Visualization:

Additional data visualizations are performed, including assessing the distribution of mental health consequences by gender groups, evaluating employee count distribution by tech company, and creating a dashboard with subplots to explore various dimensions related to treatment and company policies.

8. Statistical Test:

A two-sample t-test is conducted to determine whether there is a statistically significant difference in the ages of individuals seeking treatment and those not seeking treatment. The results show that there is no significant difference in age between these two groups.

Overall, the project aims to provide valuable insights into the effectiveness of public health awareness campaigns. It leverages data analysis and machine learning to inform future campaign strategies, data-driven decision-making in public health, and ultimately improve health outcomes.