Public Health Awareness Campaigns Data Visualization

**Title: Data Visualization for Public Health Awareness Campaigns Analysis**

**Project Description:**The Data Visualization project aims to provide insightful and actionable visual representations of data related to public health awareness campaigns. Visualizations play a crucial role in simplifying complex data, highlighting trends, and facilitating decision-making. This project focuses on transforming cleaned campaign data into meaningful charts, graphs, and visual narratives that help stakeholders gain a deeper understanding of public health awareness efforts.

**Project Objectives:**Data Selection: Identify and select the most relevant attributes and metrics from the cleaned campaign dataset for visualization.

**Visualization Design:** Plan and design an array of visualizations, including bar charts, line charts, heatmaps, and more, to represent campaign data effectively.

**Storytelling:** Create a compelling and data-driven narrative that explains key campaign insights to both technical and non-technical audiences.

**Interactive Dashboards:** Develop interactive dashboards or reports to allow users to explore campaign data, filter information, and gain real-time insights.

**Insight Extraction:** Extract actionable insights and trends from the visualizations that can guide future public health awareness campaigns.

**Feedback Incorporation:** Collaborate with stakeholders and subject matter experts to incorporate feedback and refine visualizations.

**Documentation:** Maintain clear documentation of visualization methodologies, tools, and data sources for reproducibility.

**Source:**

Cleaned data from phase 3

**Code:**

Import pandas as pd

Import numpy as np

Import matplotlib.pyplot as plt

From sklearn.model\_selection import train\_test\_split

From sklearn.linear\_model import LogisticRegression

From sklearn.metrics import accuracy\_score, classification\_report

From sklearn.preprocessing import LabelEncoder

# Load CSV data into a Pandas DataFrame

Data = pd.read\_csv(‘your\_data.csv’)

# Define the target variable and features

Target\_column = ‘treatment’

# Replace with specific target column

Features = [‘Age’, ‘Gender’, ‘Country’, ‘family\_history’, ‘work\_interfere’, ‘no\_employees’, ‘remote\_work’,

‘tech\_company’, ‘benefits’, ‘care\_options’, ‘wellness\_program’, ‘seek\_help’, ‘anonymity’, ‘leave’,

‘mental\_health\_consequence’, ‘phys\_health\_consequence’, ‘coworkers’, ‘supervisor’,

‘mental\_health\_interview’, ‘phys\_health\_interview’, ‘mental\_vs\_physical’, ‘obs\_consequence’]

# Filter the DataFrame to include only the selected columns

Data = data[[target\_column] + features]

# Handle categorical data by encoding it

Le = LabelEncoder()

Data[‘Gender’] = le.fit\_transform(data[‘Gender’])

Data[‘Country’] = le.fit\_transform(data[‘Country’])

# Split the data into training and testing sets

X = data.drop(columns=[target\_column])

Y = data[target\_column]

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

# Model building (example: Logistic Regression)

Model = LogisticRegression()

Model.fit(X\_train, y\_train)

# Make predictions

Y\_pred = model.predict(X\_test)

# Evaluation

Accuracy = accuracy\_score(y\_test, y\_pred)

Print(f”Accuracy: {accuracy}”)

Print(classification\_report(y\_test, y\_pred))

# Data visualization (example: a histogram of ‘Age’)

Plt.hist(data[‘Age’], bins=20, color=’blue’, alpha=0.7)

Plt.xlabel(‘Age’)

Plt.ylabel(‘Frequency’)

Plt.title(‘Age Distribution’)

Plt.show()