**PUBLIC HEALTH AWARENESS CAMPAIGN ANALYSIS**

OBJECTIVE:

The objective of a public health awareness campaign analysis is to assess the effectiveness of a campaign in achieving its goals. This typically involves measuring changes in knowledge, attitudes, and behaviors of the target audience, as well as the reach and impact of the campaign.

Public health awareness campaigns are an important tool for promoting health and preventing disease. By analyzing the effectiveness of these campaigns, researchers can identify what works best and develop more effective interventions in the future.

DATA COLLECTION PROCESS:

We have collected the data from kaggle[. https://www.kaggle.com/datasets/osmi/mental-health-in-tech-survey](.%20https:/www.kaggle.com/datasets/osmi/mental-health-in-tech-survey)

DATA VISUALIZATION

**Seaborn:**

Seaborn code is Python code that is used to create statistical graphics. Seaborn is built on top of matplotlib, and it provides a high-level interface for drawing attractive and informative statistical graphics.

Here is an example of a simple seaborn code:

Python

import seaborn as sns

import pandas as pd

# Load the tips dataset

tips = sns.load\_dataset("tips")

# Create a scatter plot of total\_bill vs. tip

sns.relplot(x="total\_bill", y="tip", data=tips)

# Show the plot

plt.show()

**Matplotlib:**

Matplotlib code is Python code that is used to create visualizations, such as charts, graphs, and plots. Matplotlib is a powerful tool for data visualization, and it can be used

to create a wide variety of plots, from simple line plots to complex scientific visualizations.

Here is an example of a simple matplotlib code to create a line plot:

Python

import matplotlib.pyplot as plt

import numpy as np

x = np.linspace(0, 10, 100)

y = np.sin(x)

plt.plot(x, y)

plt.xlabel('x')

plt.ylabel('y')

plt.title('Line Plot')

plt.show()

**Linear Regression:**

To perform linear regression in Python, you can use the sklearn.linear\_model library. This library provides a class called LinearRegression that can be used to fit a linear model to your data and make predictions.

Here is a simple example of how to use linear regression to predict the price of a house based on its square footage:

Python

import numpy as np

from sklearn.linear\_model import LinearRegression

# Create the training data

X = np.array([1200, 1500, 1800, 2100])

y = np.array([200000, 250000, 300000, 350000])

# Create the linear regression model

reg = LinearRegression()

# Fit the model to the training data

reg.fit(X, y)

# Make a prediction for a house with 1900 square feet

prediction = reg.predict([1900])

# Print the prediction

print(prediction)

**Correlation:**

To calculate correlation in Python, you can use the numpy.corrcoef() function or the scipy.stats.pearsonr() function.

The numpy.corrcoef() function calculates the correlation coefficients between all pairs of variables in a NumPy array. The correlation coefficients are returned in a square matrix, where the diagonal contains the correlations between the variables and themselves.

The scipy.stats.pearsonr() function calculates the Pearson correlation coefficient between two variables. The Pearson correlation coefficient is a measure of the linear relationship between two variables.

Here is an example of how to use the numpy.corrcoef() function to calculate the correlation coefficients between all pairs of variables in a NumPy array:

Python

import numpy as np

# Create a NumPy array with three variables

X = np.array([[1, 2, 3], [4, 5, 6], [7, 8, 9]])

# Calculate the correlation coefficients between all pairs of variables

corr = np.corrcoef(X)

# Print the correlation matrix

print(corr)

**\*Prediction and Learning in Public Health Care Awareness Campaign Analysis\***

Public health care awareness campaigns are designed to increase knowledge and understanding of important health issues, promote healthy behaviors, and encourage people to seek preventive care. By tracking and analyzing data from these campaigns, public health officials can gain valuable insights into how to improve their effectiveness.

Prediction and learning algorithms can be used to analyze campaign data in a variety of ways. For example, these algorithms can be used to:

**\* \*Predict which individuals are most likely to respond to a particular campaign message or intervention.\***

This information can be used to target outreach efforts more effectively.

**\* \*Identify factors that are associated with campaign success or failure.\***

This information can be used to improve future campaigns.

**\* \*Track changes in knowledge, attitudes, and behaviors over time.\***

This information can be used to assess the impact of campaigns and make necessary adjustments.

Here are some specific examples of how prediction and learning algorithms can be used to analyze public health care awareness campaign data:

**\* \*Predicting the likelihood of vaccination:\*** A public health official could use a prediction algorithm to identify individuals who are most likely to hesitate about getting vaccinated against a particular disease. This information could then be used to target outreach efforts with tailored messages and interventions.

**\* \*Identifying factors associated with smoking cessation:\*** A researcher could use a learning algorithm to identify factors that are associated with success in smoking cessation programs. This information could then be used to develop more effective programs and interventions.

**\* \*Tracking changes in knowledge and attitudes about healthy eating:\*** A public health agency could use a learning algorithm to track changes in knowledge and attitudes about healthy eating among a population over time. This information could then be used to assess the impact of public health campaigns and make necessary adjustments.

Prediction and learning algorithms have the potential to revolutionize the way public health campaigns are designed, implemented, and evaluated. By using these algorithms to analyze campaign data, public health officials can gain valuable insights into how to improve the effectiveness of their campaigns and achieve better health outcomes for the populations they serve.

## Challenges and Limitations

While prediction and learning algorithms offer significant potential for improving the effectiveness of public health care awareness campaigns, there are also some challenges and limitations that need to be considered.

One challenge is that these algorithms can only be as good as the data they are trained on. If the data is incomplete or inaccurate, the algorithms will not be able to make accurate predictions or learn effectively.

Another challenge is that these algorithms can be complex and difficult to interpret. This can make it difficult for public health officials to understand how the algorithms are working and to trust the results.

Finally, it is important to note that these algorithms cannot replace the need for human expertise and judgment. Public health officials still need to use their knowledge and experience to design and implement campaigns that are tailored to the specific needs of their communities.

## Conclusion

Despite the challenges, prediction and learning algorithms have the potential to make a significant contribution to improving the effectiveness of public health care awareness campaigns. By using these algorithms to analyze campaign data, public health officials can gain valuable insights into how to improve the design, implementation, and evaluation of their campaigns. This can lead to better health outcomes for the populations they serve.

SNIPPET:

import csv

import pandas as pd

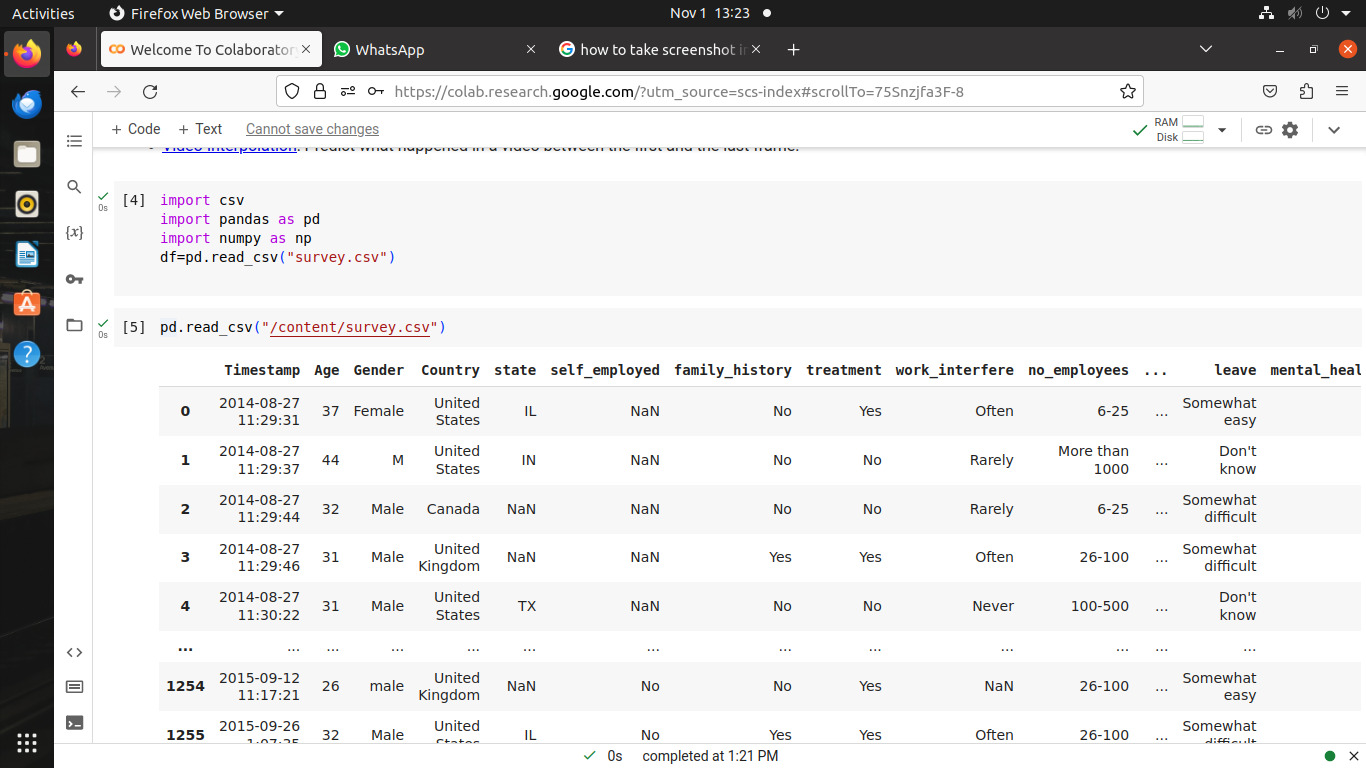
import numpy as np

df=pd.read\_csv("survey.csv")

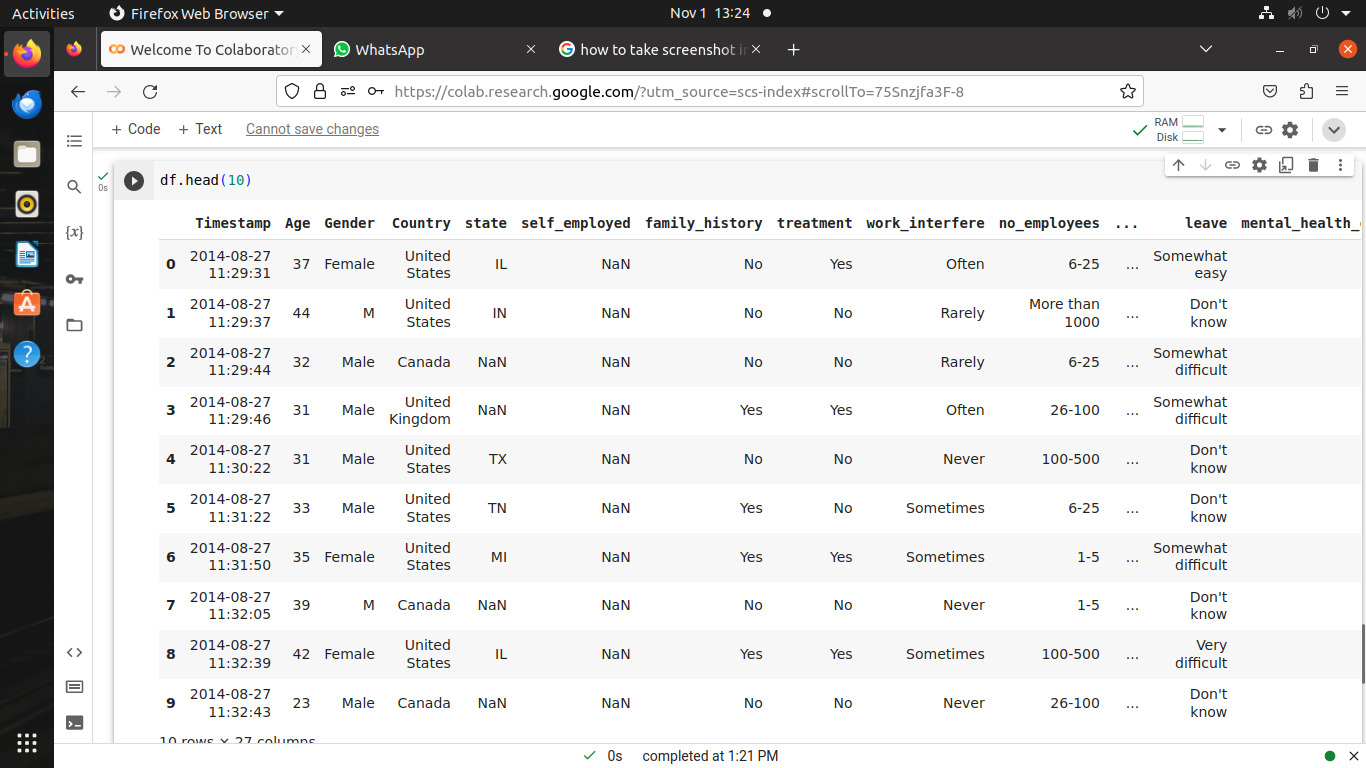
from sklearn.model\_selection import train\_test\_split

import matplotlib.pyplot as plt

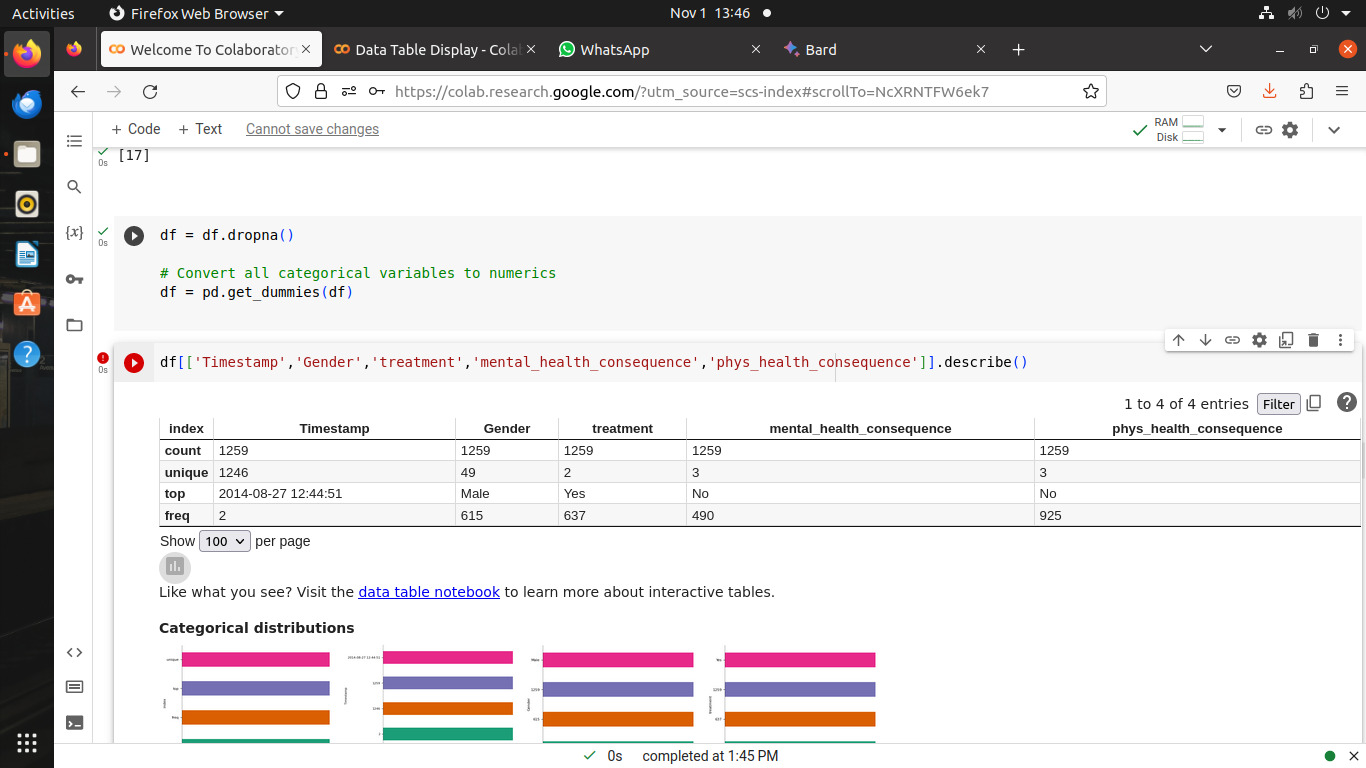
from sklearn.linear\_model import LinearRegression



pd.read\_csv("/content/survey.csv")



df[['Timestamp','Gender','treatment','mental\_health\_consequence','phys\_health\_consequence']].describe()

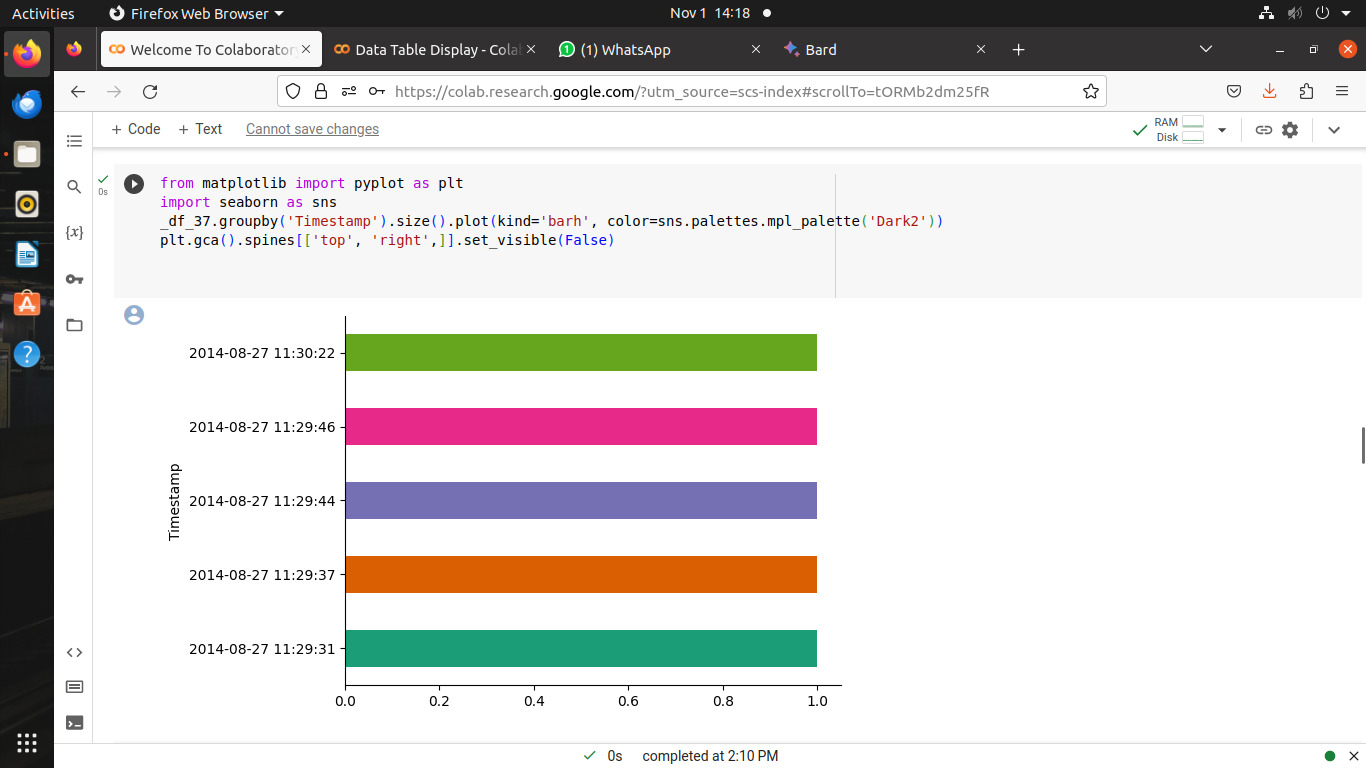


from matplotlib import pyplot as plt

import seaborn as sns

\_df\_37.groupby('Timestamp').size().plot(kind='barh', color=sns.palettes.mpl\_palette('Dark2'))

plt.gca().spines[['top', 'right',]].set\_visible(False)



from matplotlib import pyplot as plt

import seaborn as sns

def \_plot\_series(series, series\_name, series\_index=0):

from matplotlib import pyplot as plt

import seaborn as sns

palette = list(sns.palettes.mpl\_palette('Dark2'))

xs = series['Timestamp']

ys = series['Age']

plt.plot(xs, ys, label=series\_name, color=palette[series\_index % len(palette)])

fig, ax = plt.subplots(figsize=(10, 5.2), layout='constrained')

df\_sorted = \_df\_42.sort\_values('Timestamp', ascending=True)

for i, (series\_name, series) in enumerate(df\_sorted.groupby('Gender')):

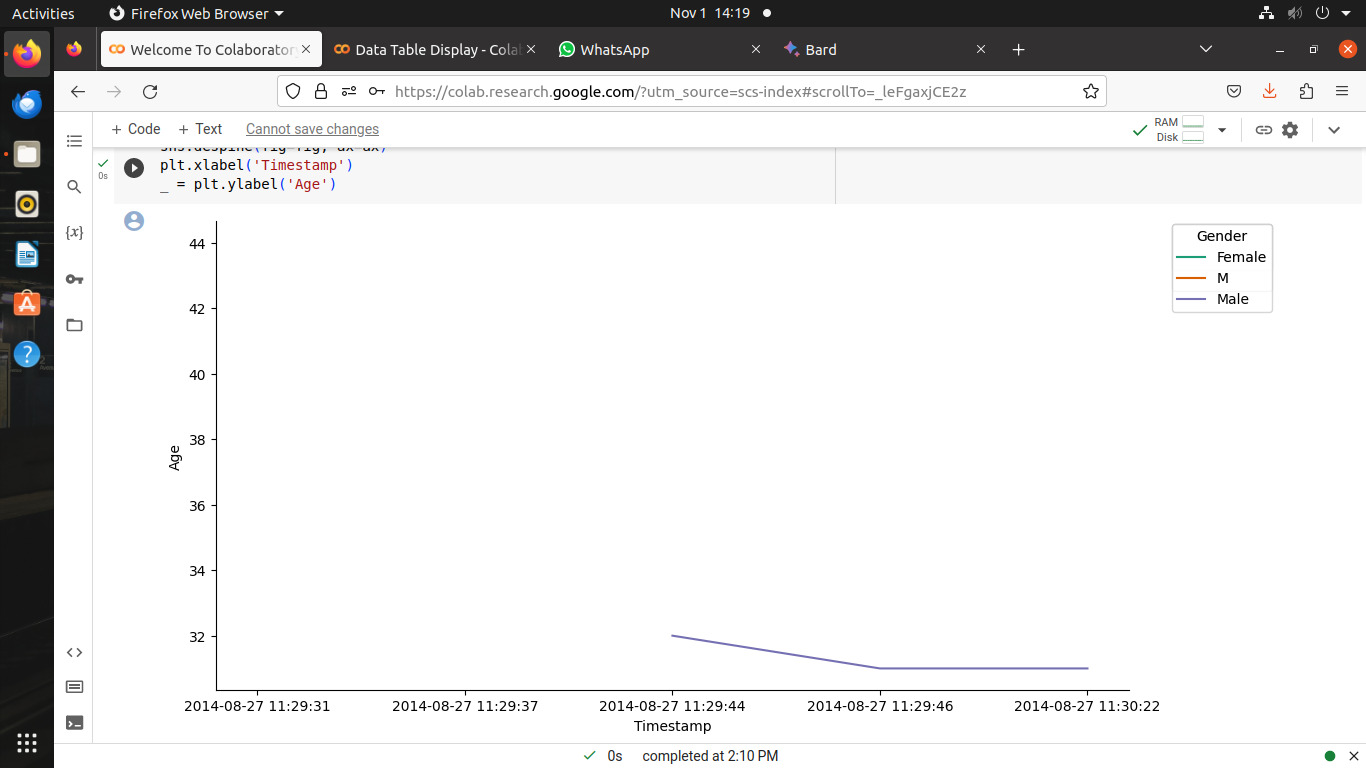
\_plot\_series(series, series\_name, i)

fig.legend(title='Gender', bbox\_to\_anchor=(1, 1), loc='upper left')

sns.despine(fig=fig, ax=ax)

plt.xlabel('Timestamp')

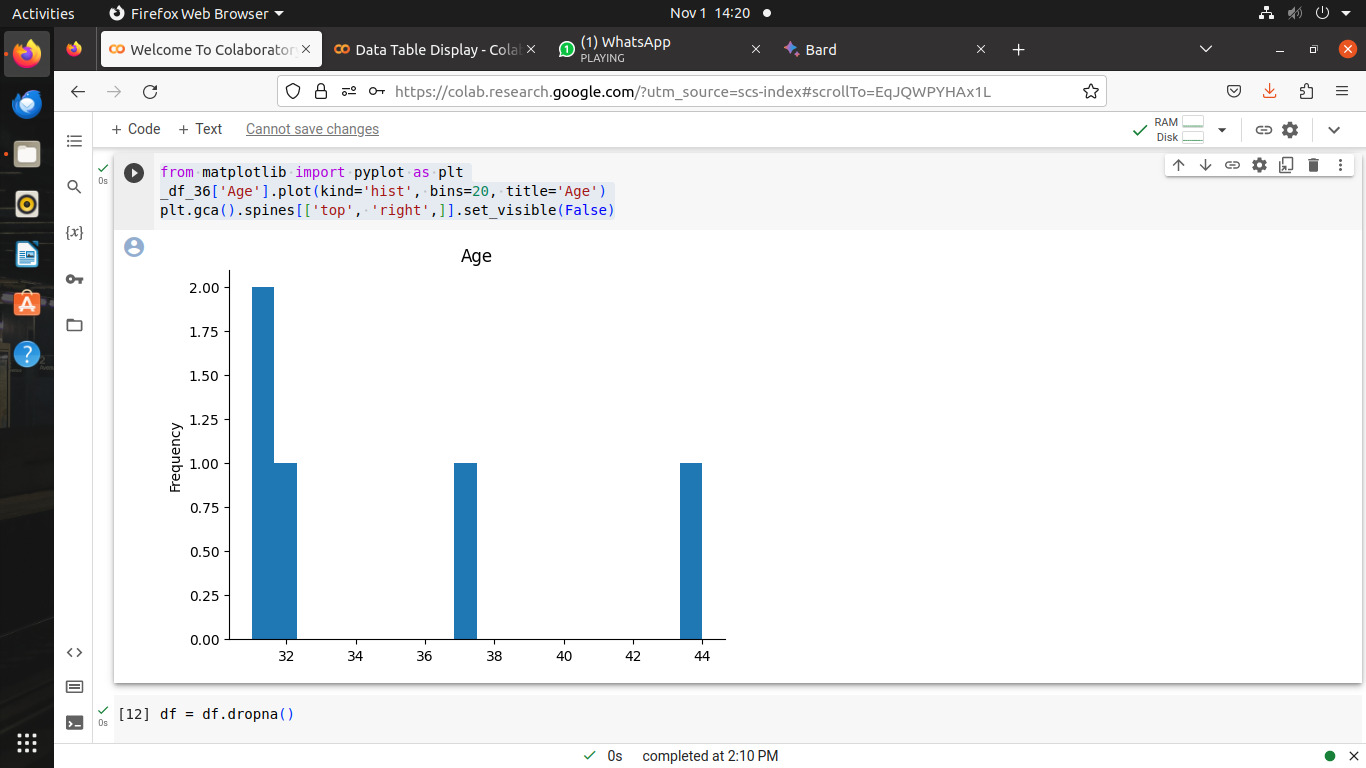
\_ = plt.ylabel('Age')



from matplotlib import pyplot as plt

\_df\_36['Age'].plot(kind='hist', bins=20, title='Age')

plt.gca().spines[['top', 'right',]].set\_visible(False)

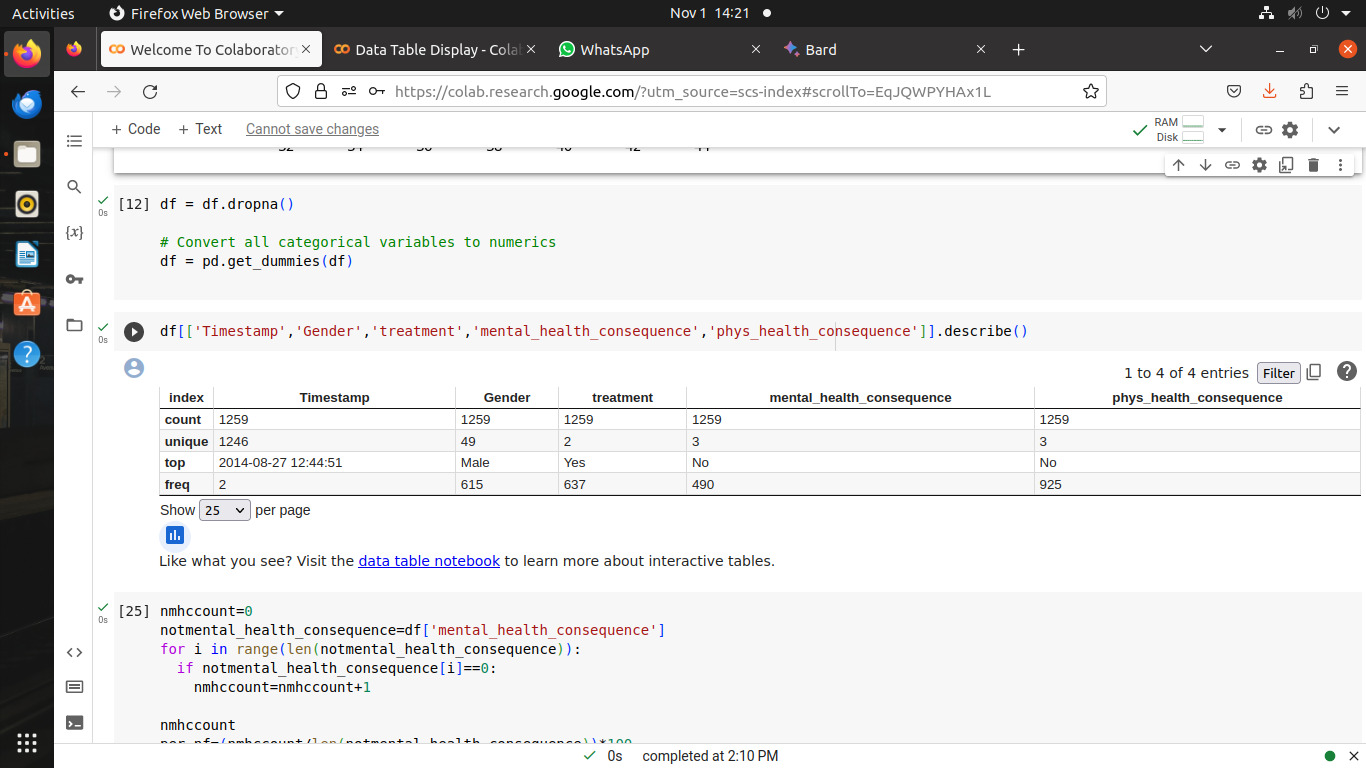


df = df.dropna()

# Convert all categorical variables to numerics

df = pd.get\_dummies(df)

df[['Timestamp','Gender','treatment','mental\_health\_consequence','phys\_health\_consequence']].describe()



nmhccount=0

notmental\_health\_consequence=df['mental\_health\_consequence']

for i in range(len(notmental\_health\_consequence)):

if notmental\_health\_consequence[i]==0:

nmhccount=nmhccount+1

nmhccount

per\_nf=(nmhccount/len(notmental\_health\_consequence))\*100

print('total percentage for mental Health concequence: ',per\_nf)

**OUTPUT:**

total percentage for mental Health concequence: 0.0

nphccount=0

notphys\_health\_consequence=df['phys\_health\_consequence']

for i in range(len(notphys\_health\_consequence)):

if notphys\_health\_consequence[i]==0:

nphccount=nmhccount+1

nphccount

per\_nf=(nphccount/len(notphys\_health\_consequence))\*100

print('total percentage for mental Health concequence: ',per\_nf)

**OUTPUT:**

total percentage for mental Health concequence: 0.0

from matplotlib import pyplot as plt

import seaborn as sns

import pandas as pd

plt.subplots(figsize=(8, 8))

df\_2dhist = pd.DataFrame({

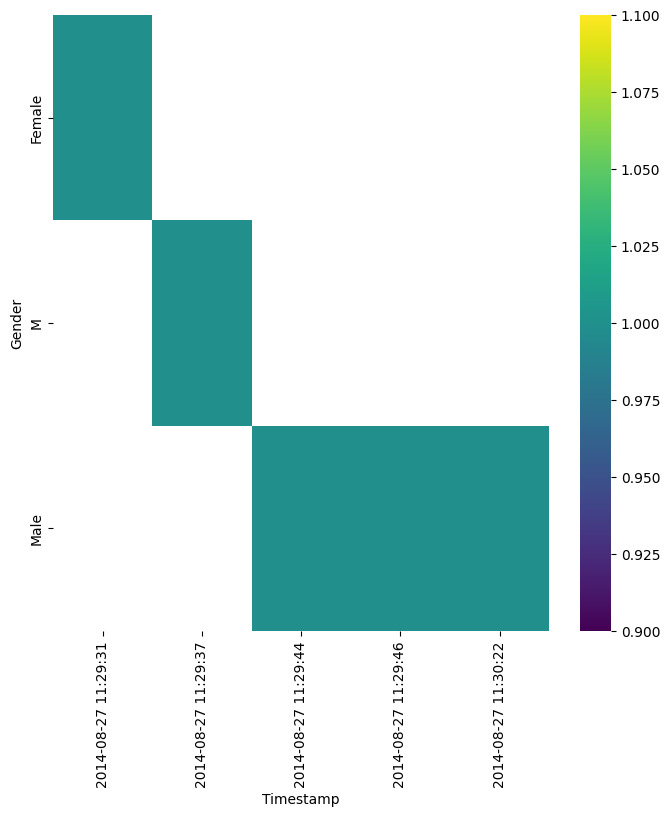
x\_label: grp['Gender'].value\_counts()

for x\_label, grp in \_df\_46.groupby('Timestamp')

})

sns.heatmap(df\_2dhist, cmap='viridis')

plt.xlabel('Timestamp')

\_ = plt.ylabel('Gender')

from matplotlib import pyplot as plt

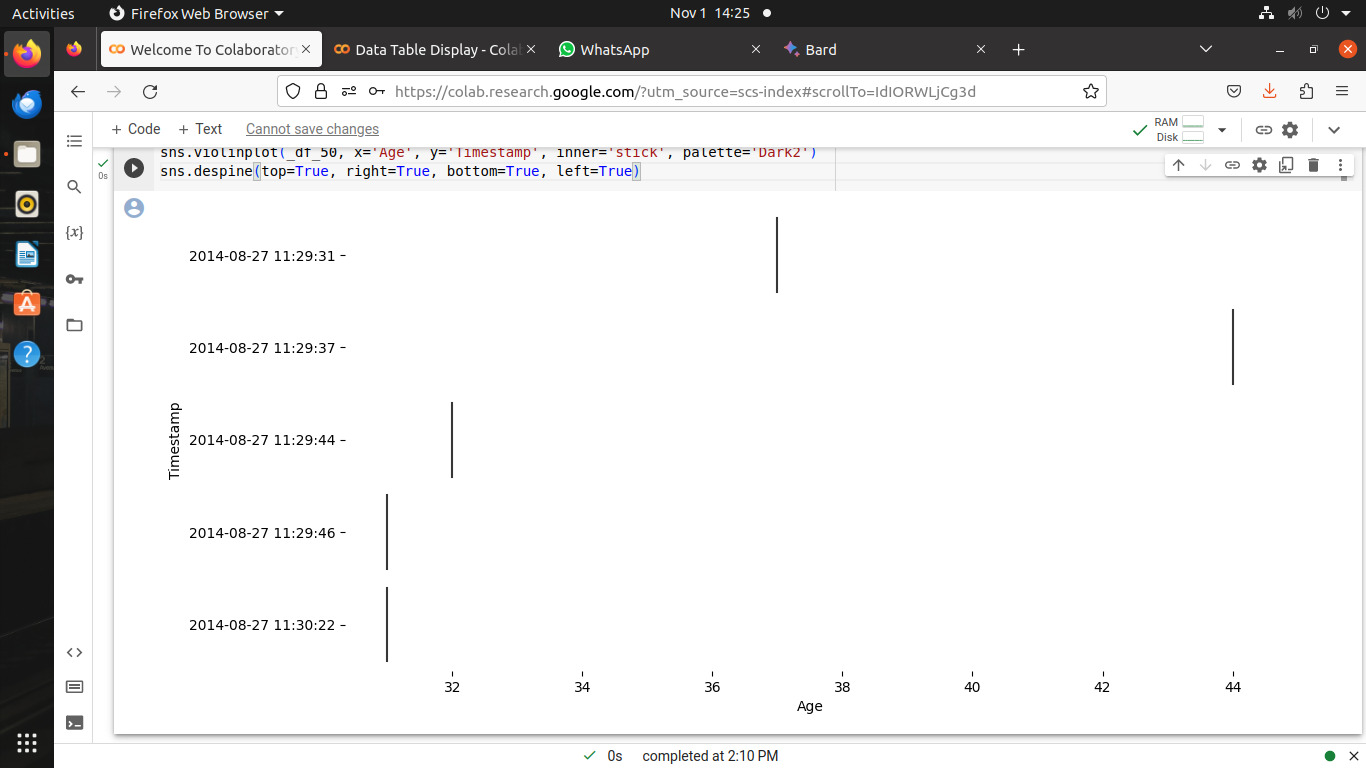
import seaborn as sns

figsize = (12, 1.2 \* len(\_df\_50['Timestamp'].unique()))

plt.figure(figsize=figsize)

sns.violinplot(\_df\_50, x='Age', y='Timestamp', inner='stick', palette='Dark2')

sns.despine(top=True, right=True, bottom=True, left=True)



**Key Take Away Points**

1. Country with the highest number of employees with mental health disorder is United States, with total number is 408 employees. It is more than 50% of respondents in United States.

2. There is a big gap between number of employees having mental health disorder in United States with other countries.

3. As a country with highest number of employees having mental health disorder, most of employers in United States are concerned enough on mental health condition of employees. It can be seen from the many companies that provide mental health benefits for employees.

4. Even so, not a few companies in United States are still not aware of the mental health condition of their employees. It can be seen from the fact that there are many employers that have not included mental health treatment as the company facilitiy and make it difficult employees to leave from office for mental health treatment purposes.

5. If we see to the attitude of the employees, we can see that most of employees, both fellow coworkers and supervisors, are equally open to employees who want to share their mental health condition. But some of the employees that choose to share about their mental health condition think it may give negative consequences to them and it may not get guarantee of their anonymity, although few cases are found that employees experienced negative consequences after they share about their mental health condition.

6. Although employers awarness of mental health is good, most employers prefer not to shortlist potential candidates with mental health disorder.

**Conclusion after Test:**

Based on the ROC-AUC Curve above, it shows that the Stacking Classifier model apart from having an AUC (Threshold) value = 0.7212 (with Accuration Score= 0.7233) stiil bigger than 0.5 (Standard AUC or threshold score) for the true positive rate is also higher than other models, apart from the Logistic Regressiion and Random Forest which has a slightly lower difference.

Stacking Classifier : Approaching the prediction totally random, not the opposite answer correct.

By using 3 models as a comparison in making predictions, namely Logistic Regression, Random Forest Classifier and Stacking Classifier, The Stacking Classifier is better in terms of making predictions because has better performance than any single model in the ensemble. The Stacking Classifier leverages the capabilities of multiple models that perform well at classification tasks

The Matrix Score above will change at runtime, although not too significant. this is due to the influence of the stochastic algorithm on each basic model.