PUBLIC HEALTH AWARNESS AND CAMPAIGN ANALYIS

INTRODUCTION:

Public health is the science of protecting and improving the health of people and there communities.

According to the world health organisation " Health is a state of complete physical, mental and social well-being and not merely the absence of disease or infirmity" .

PRINCE MAHIDOL is a father of public health was born in bangkok on jan01 1981.

Awarness campaigns can address groups of people in a region affected by a particular climate threat, groups of stackholder, businesses or the public in general.

METHODS:

Machine learning techniques applied to public health surveillance data.

Exploratory data analysis of public health surveillance data

Public health surveillance system

Knowledge discovery and pattern recognition from public health surveillance data

Pre-processing public health surveillance data

OBJECTIVIES:

The purpose of a health campaign is to inform, remind and educate patients about their on going health care and make it easy for them to take steps towards their providers.

The four common aims of conducting campaigns are to :
1.Raise awareness 2.change attitudes 3.mobilize action 4. influence policy

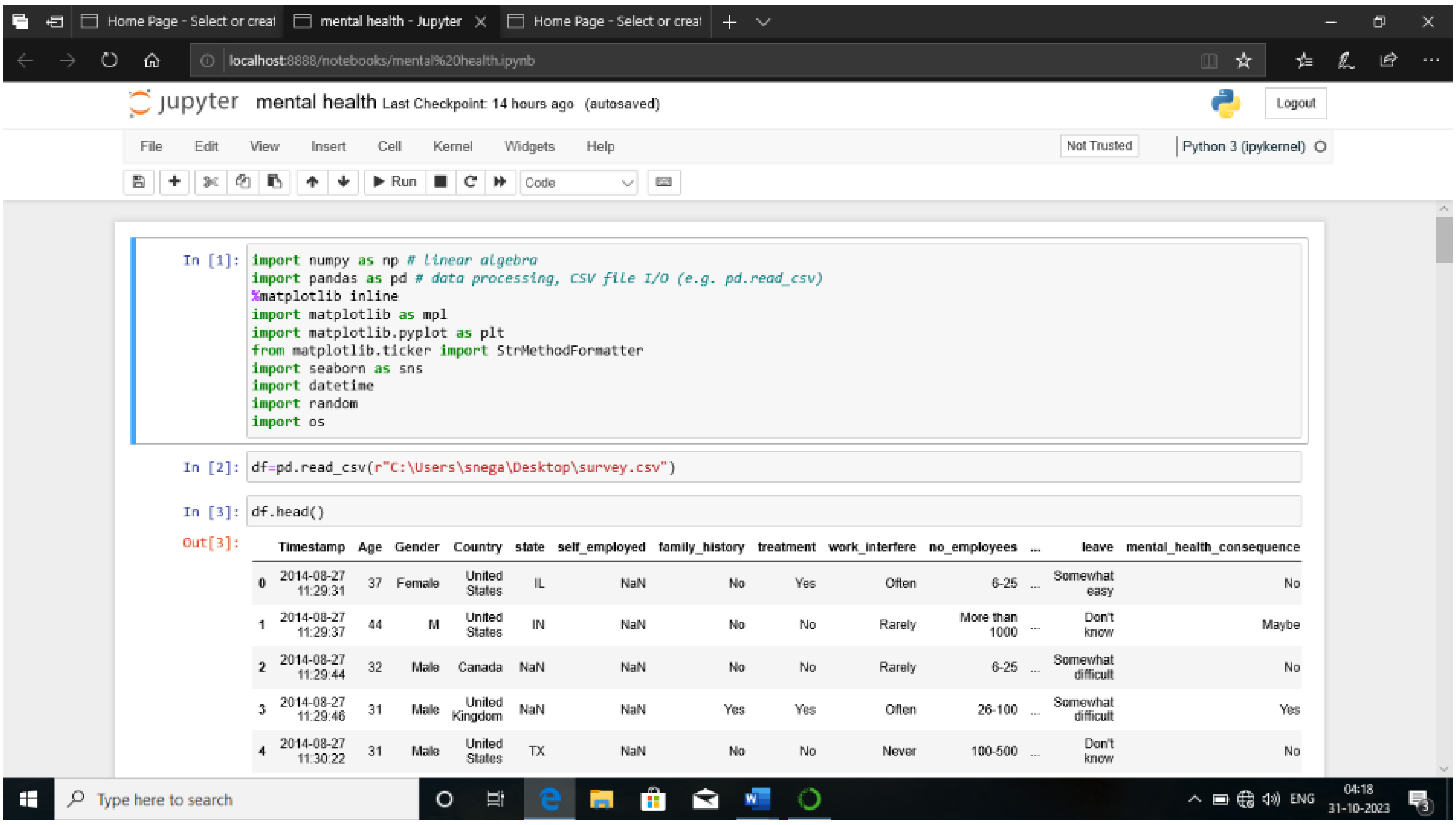
DATASET LINK:
https://www.kaggle.com/dataset/osmi/mental-health-in-tech-survey

MENTAL HEALTH DATA ANALYSIS:

- Reading the data sets

- Datasets in IBM DASHBOARD
- Data visualization
- Conclusion

1.Reading the data sets:

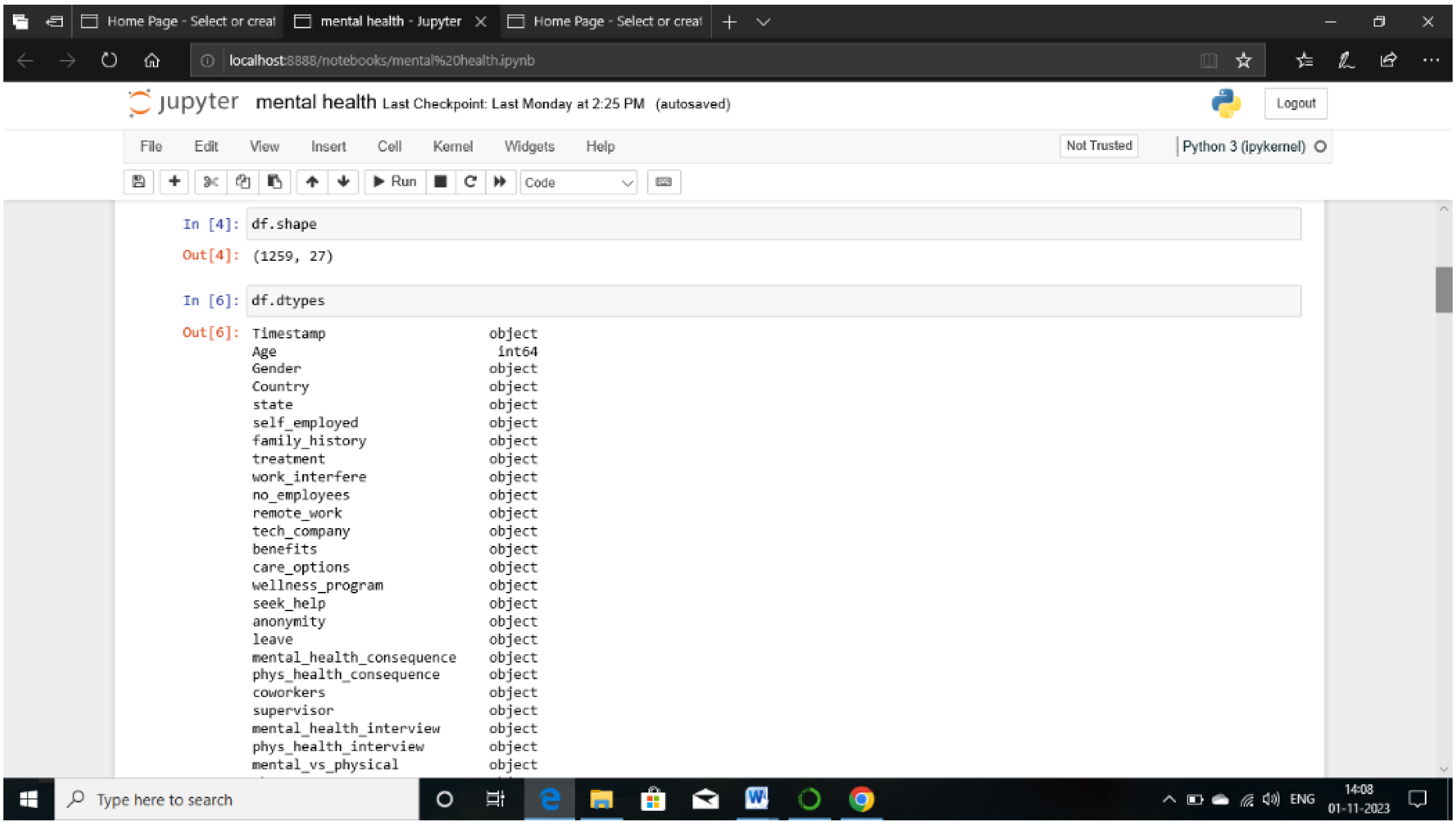


```
In [1]: import numpy as np # linear algebra
        import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
        %matplotlib inline
        import matplotlib as mpl
        import matplotlib.pyplot as plt
        from matplotlib.ticker import StrMethodFormatter
        import seaborn as sns
        import datetime
        import random
        import os

In [2]: df=pd.read_csv(r"C:\Users\snega\Desktop\survey.csv")

In [3]: df.head()
```

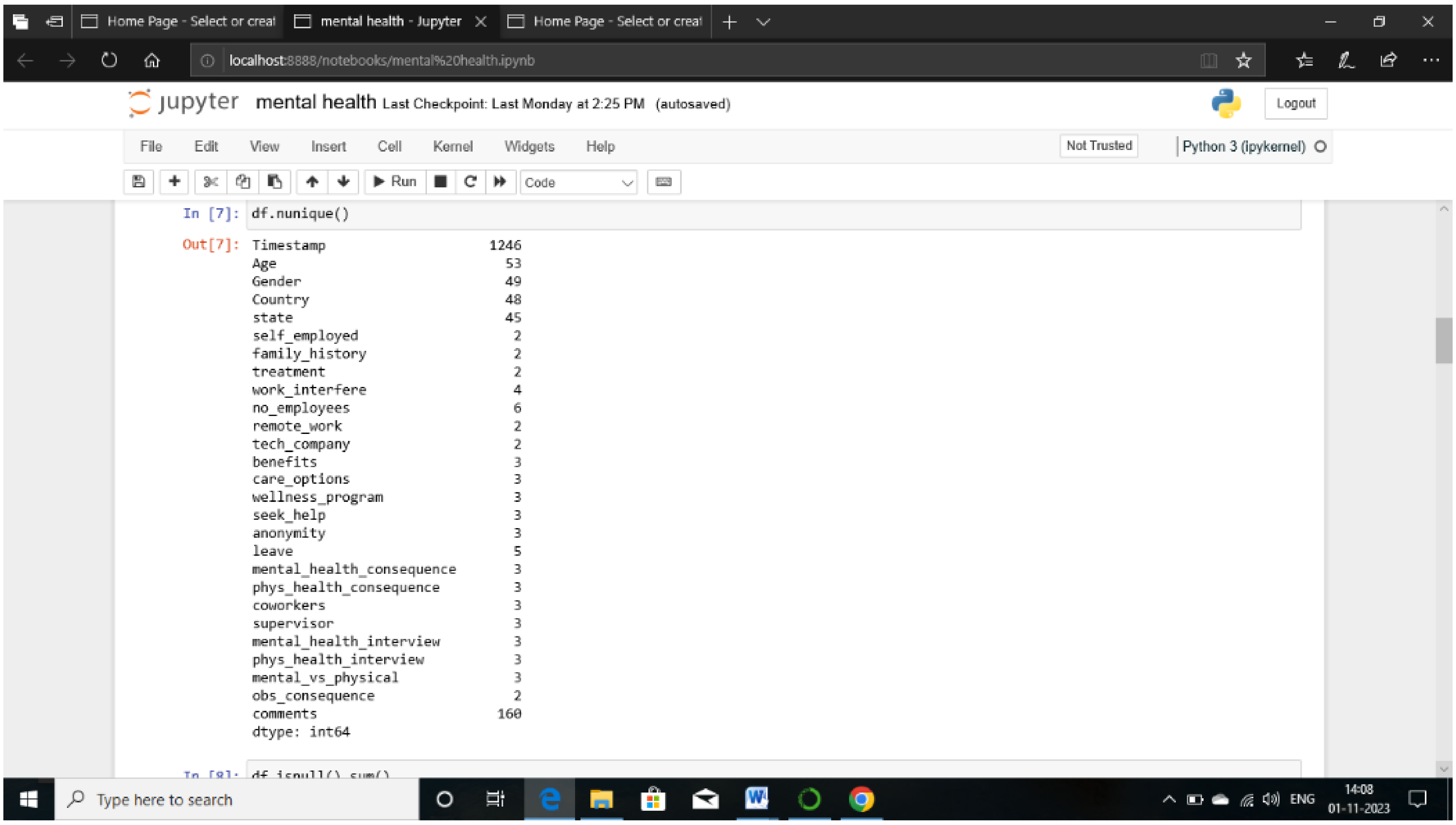
Out[3]:

	Timestamp	Age	Gender	Country	state	self_employed	family_history	treatment	work_interfere	no_employees	...	leave	mental_health_consequence
0	2014-08-27 11:29:31	37	Female	United States	IL	NaN	No	Yes	Often	6-25	...	Somewhat easy	No
1	2014-08-27 11:29:37	44	M	United States	IN	NaN	No	No	Rarely	More than 1000	...	Don't know	Maybe
2	2014-08-27 11:29:44	32	Male	Canada	NaN	NaN	No	No	Rarely	6-25	...	Somewhat difficult	No
3	2014-08-27 11:29:46	31	Male	United Kingdom	NaN	NaN	Yes	Yes	Often	26-100	...	Somewhat difficult	Yes
4	2014-08-27 11:30:22	31	Male	United States	TX	NaN	No	No	Never	100-500	...	Don't know	No

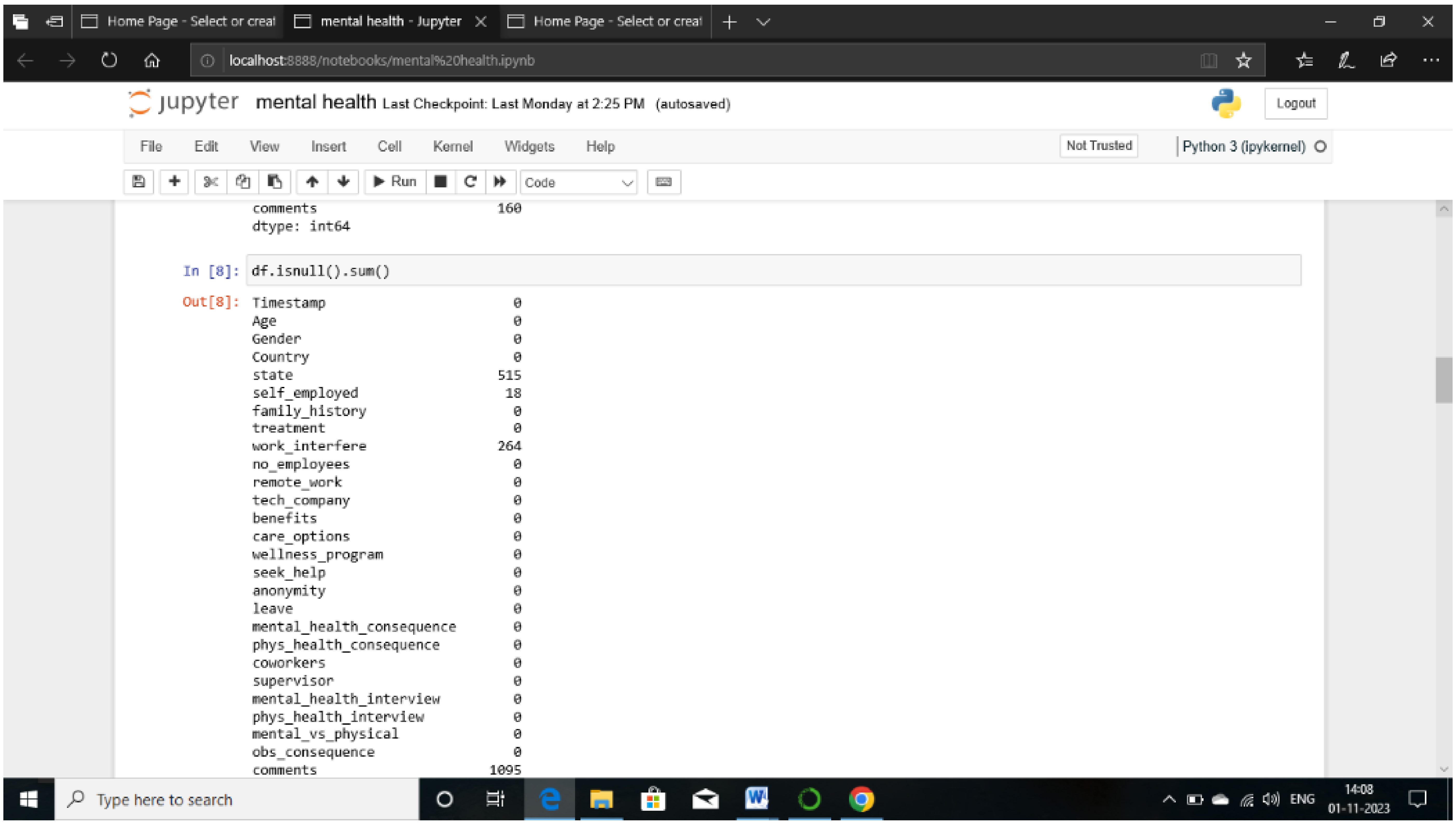
Home Page - Select or creat
mental health - Jupyter
Home Page - Select or creat
localhost:8888/notebooks/mental%20health.ipynb
jupyter mental health Last Checkpoint: 14 hours ago (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Not Trusted
Python 3 (ipykernel)
Run
Code
Type here to search
ENG
04:18
31-10-2023


Home Page - Select or creat
mental health - Jupyter
Home Page - Select or creat
localhost:8888/notebooks/mental%20health.ipynb
jupyter mental health Last Checkpoint: Last Monday at 2:25 PM (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3 (ipykernel)
Run
Code
Type here to search
ENG
14:08
01-11-2023

```
In [4]: df.shape

Out[4]: (1259, 27)

In [6]: df.dtypes

Out[6]: Timestamp                    object
        Age                           int64
        Gender                       object
        Country                      object
        state                        object
        self_employed                object
        family_history               object
        treatment                    object
        work_interfere               object
        no_employees                 object
        remote_work                  object
        tech_company                 object
        benefits                     object
        care_options                 object
        wellness_program             object
        seek_help                    object
        anonymity                    object
        leave                        object
        mental_health_consequence    object
        phys_health_consequence      object
        coworkers                    object
        supervisor                   object
        mental_health_interview      object
        phys_health_interview        object
        mental_vs_physical           object
```


Home Page - Select or creat
mental health - Jupyter
Home Page - Select or creat
localhost:8888/notebooks/mental%20health.ipynb
jupyter mental health Last Checkpoint: Last Monday at 2:25 PM (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Not Trusted
Python 3 (ipykernel)
Run
Code
Type here to search
ENG
14:08
01-11-2023

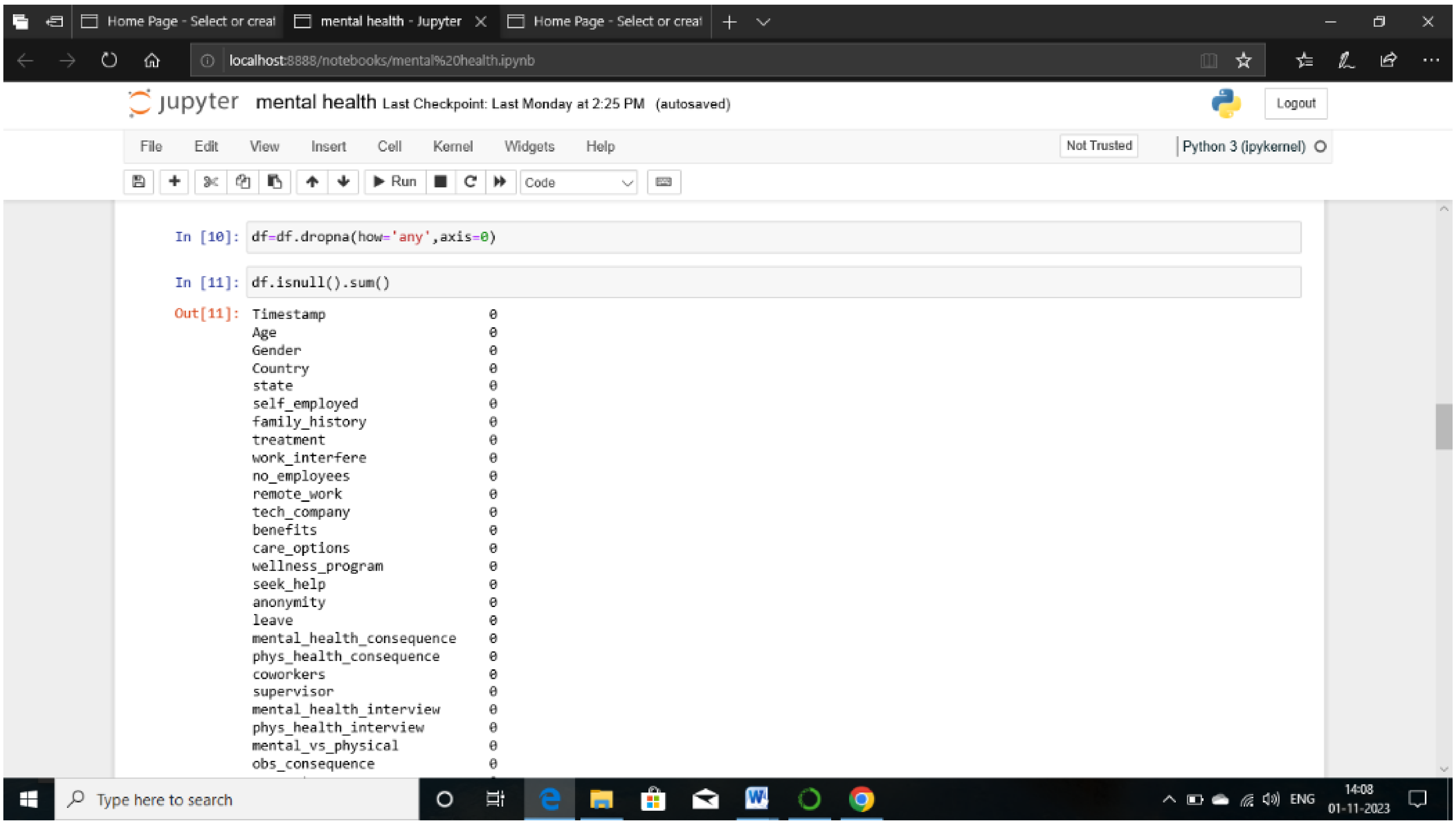
```
In [7]: df.nunique()

Out[7]: Timestamp                    1246
        Age                            53
        Gender                         49
        Country                        48
        state                          45
        self_employed                   2
        family_history                  2
        treatment                       2
        work_interfere                  4
        no_employees                    6
        remote_work                     2
        tech_company                    2
        benefits                        3
        care_options                    3
        wellness_program                3
        seek_help                       3
        anonymity                       3
        leave                           5
        mental_health_consequence       3
        phys_health_consequence         3
        coworkers                       3
        supervisor                      3
        mental_health_interview         3
        phys_health_interview           3
        mental_vs_physical              3
        obs_consequence                 2
        comments                      160
        dtype: int64

In [8]: df.isnull().sum()
```


Home Page - Select or creat
mental health - Jupyter
Home Page - Select or creat
localhost:8888/notebooks/mental%20health.ipynb
jupyter
mental health Last Checkpoint: Last Monday at 2:25 PM (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Not Trusted
Python 3 (ipykernel)
Run
Code
Type here to search
ENG
14:08
01-11-2023

```
comments                      160
dtype: int64
```

In [8]:
```
df.isnull().sum()
```

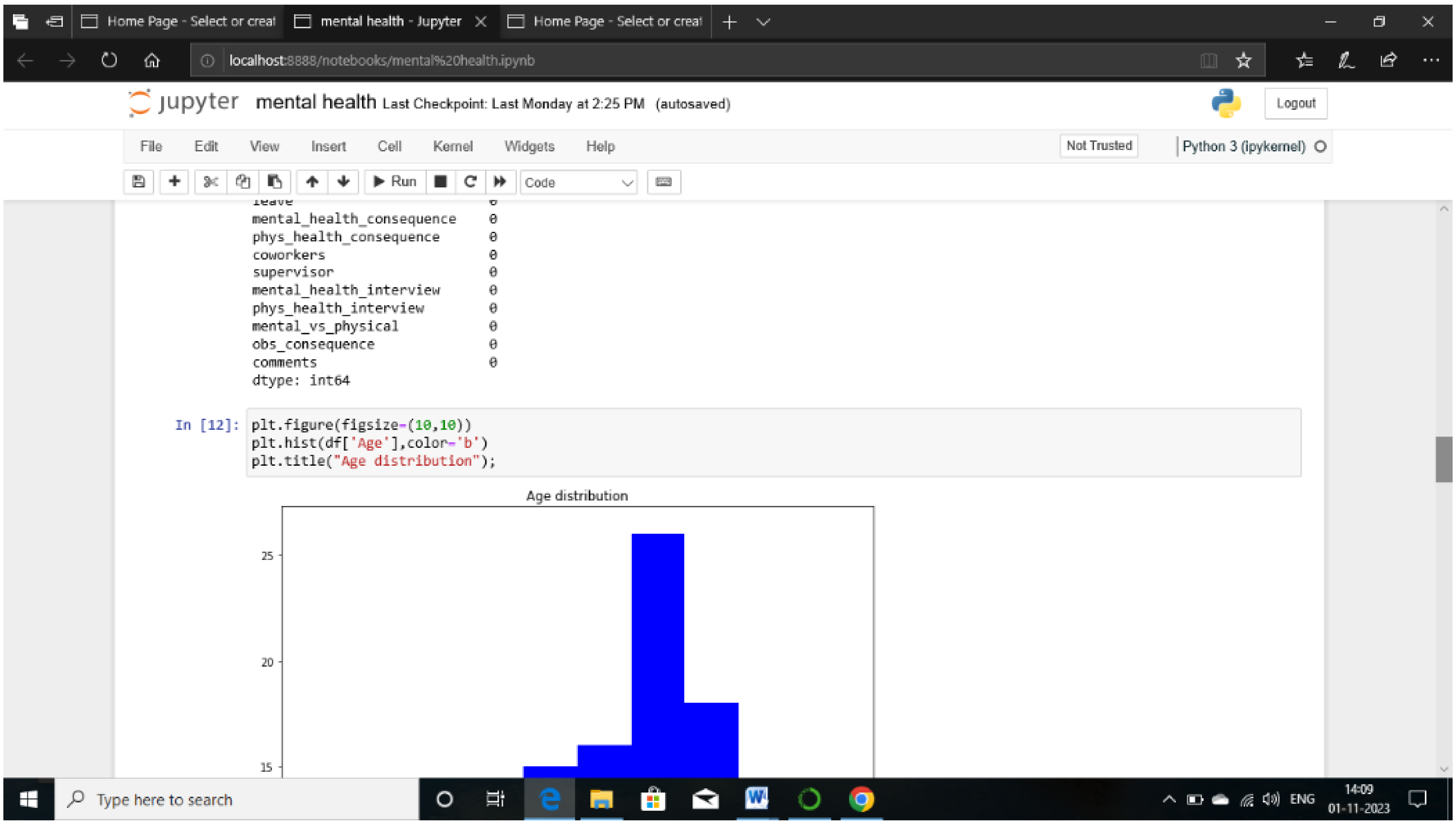
Out[8]:
```
Timestamp                       0
Age                             0
Gender                          0
Country                         0
state                         515
self_employed                  18
family_history                  0
treatment                       0
work_interfere                264
no_employees                    0
remote_work                     0
tech_company                    0
benefits                        0
care_options                    0
wellness_program                0
seek_help                       0
anonymity                       0
leave                           0
mental_health_consequence       0
phys_health_consequence         0
coworkers                       0
supervisor                      0
mental_health_interview         0
phys_health_interview           0
mental_vs_physical              0
obs_consequence                 0
comments                     1095
```


Home Page - Select or creat
mental health - Jupyter
Home Page - Select or creat
localhost:8888/notebooks/mental%20health.ipynb
jupyter
mental health Last Checkpoint: Last Monday at 2:25 PM (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Not Trusted
Python 3 (ipykernel)
Run
Code
Type here to search
ENG
14:08
01-11-2023

In [10]:
```
df=df.dropna(how='any',axis=0)
```

In [11]:
```
df.isnull().sum()
```

Out[11]:
```
Timestamp                    0
Age                          0
Gender                       0
Country                      0
state                        0
self_employed                0
family_history               0
treatment                    0
work_interfere               0
no_employees                 0
remote_work                  0
tech_company                 0
benefits                     0
care_options                 0
wellness_program             0
seek_help                    0
anonymity                    0
leave                        0
mental_health_consequence    0
phys_health_consequence      0
coworkers                    0
supervisor                   0
mental_health_interview      0
phys_health_interview        0
mental_vs_physical           0
obs_consequence              0
```



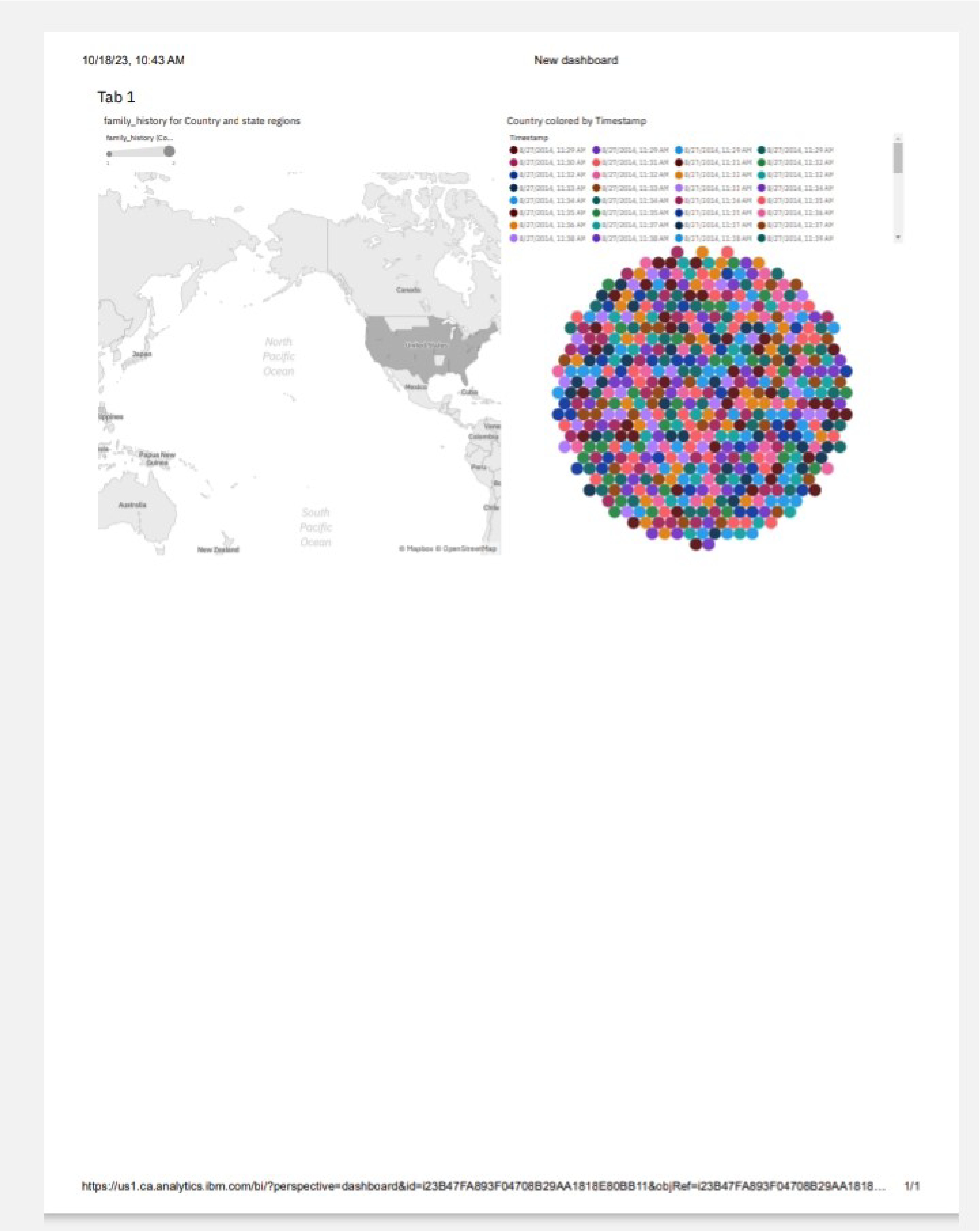
```
leave                        0
mental_health_consequence    0
phys_health_consequence      0
coworkers                    0
supervisor                   0
mental_health_interview      0
phys_health_interview        0
mental_vs_physical           0
obs_consequence              0
comments                     0
dtype: int64
```

In [12]:

```
plt.figure(figsize=(10,10))
plt.hist(df['Age'],color='b')
plt.title("Age distribution");
```

DATASETS IN IBM DASHBOARD:

DASHBOARD 1:



10/18/23, 10:43 AM

New dashboard

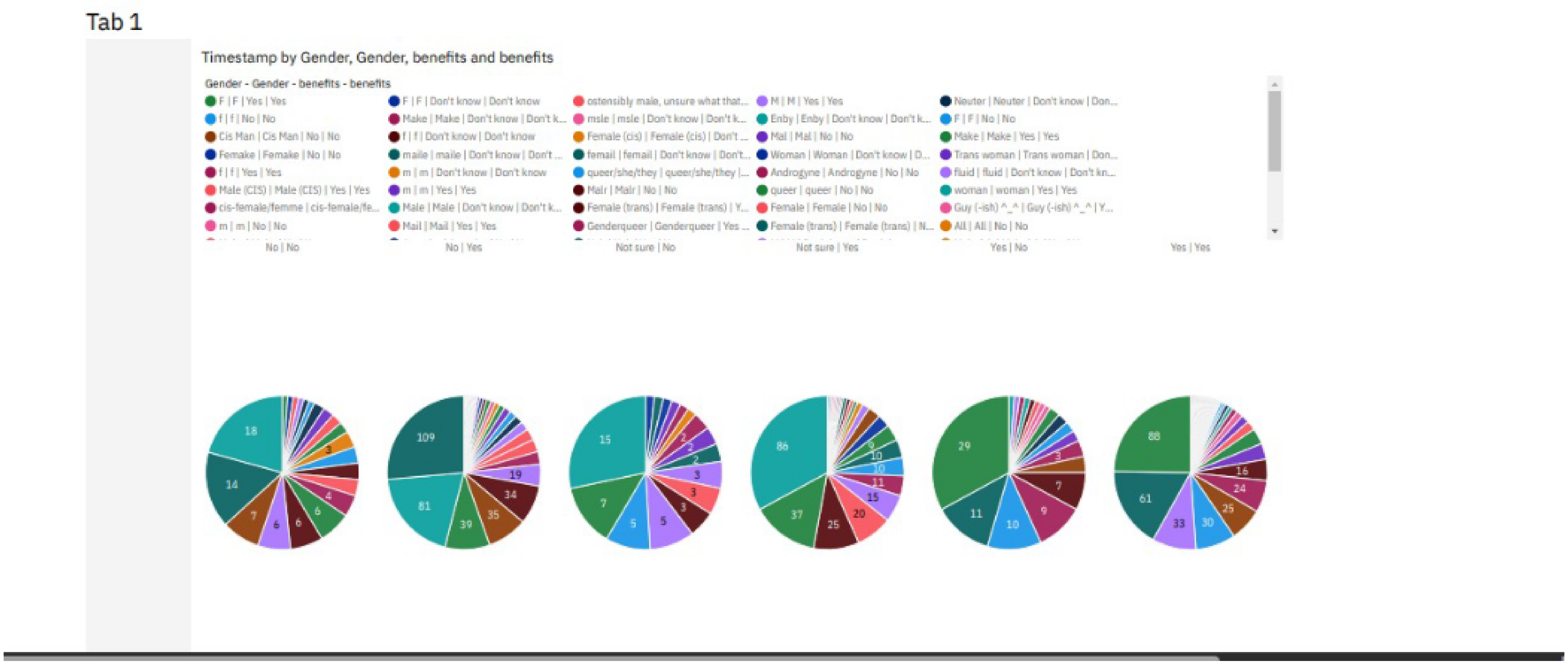
Tab 1

family_history for Country and state regions

Country colored by Timestamp

https://us1.ca.analytics.ibm.com/bi/?perspective=dashboard&id=i23B47FA893F04708B29AA1818E80BB11&objRef=i23B47FA893F04708B29AA1818... 1/1

DASHBOARD 2:

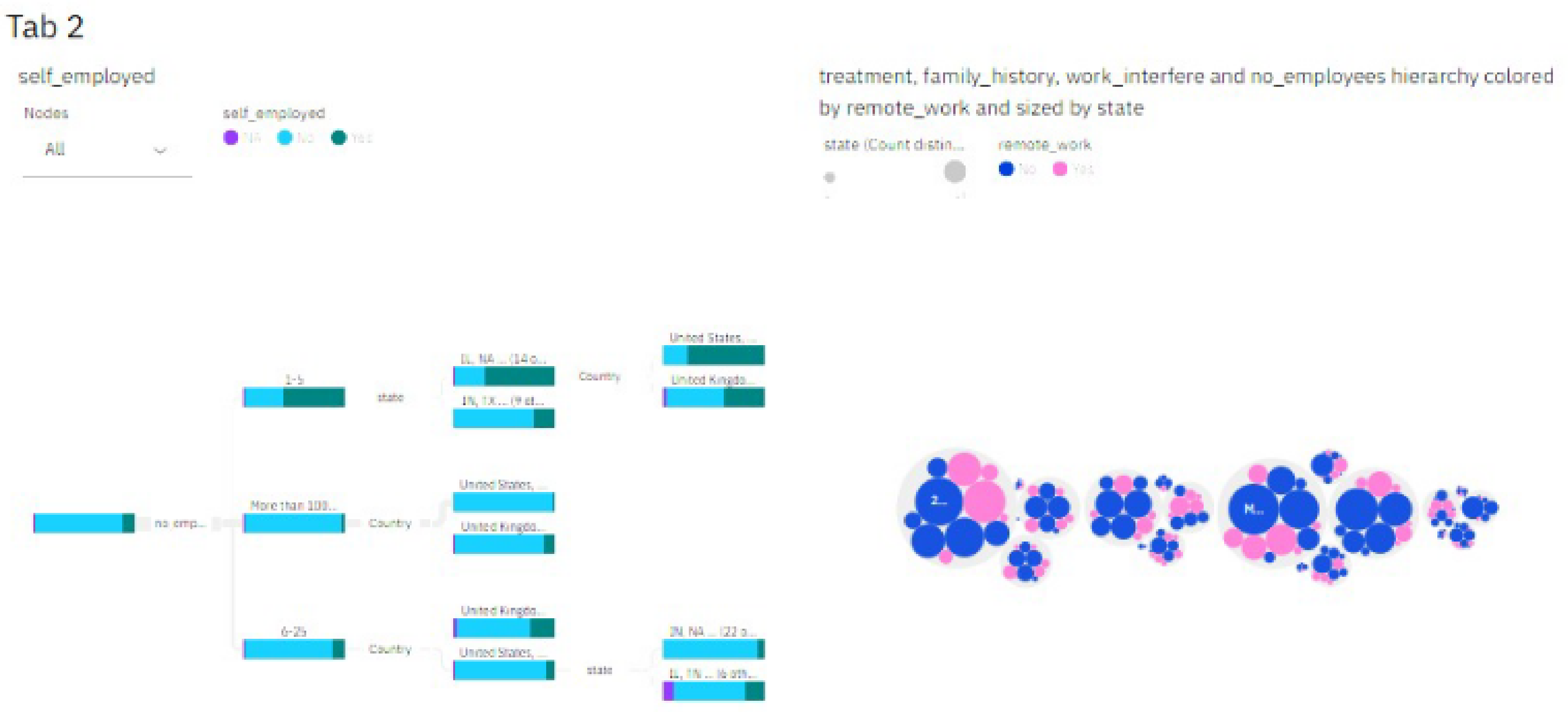


Tab 1

Timestamp by Gender, Gender, benefits and benefits

Gender - Gender - benefits - benefits
F | F | Yes | Yes
f | f | No | No
Cis Man | Cis Man | No | No
Femake | Femake | No | No
f | f | Yes | Yes
Male (CIS) | Male (CIS) | Yes | Yes
cis-female/femme | cis-female/fe...
m | m | No | No
F | F | Don't know | Don't know
Make | Make | Don't know | Don't k...
f | f | Don't know | Don't know
maile | maile | Don't know | Don't ...
m | m | Don't know | Don't know
m | m | Yes | Yes
Male | Male | Don't know | Don't k...
Mail | Mail | Yes | Yes
ostensibly male, unsure what that...
msle | msle | Don't know | Don't k...
Female (cis) | Female (cis) | Don't ...
femail | femail | Don't know | Don't...
queer/she/they | queer/she/they |...
Malr | Malr | No | No
Female (trans) | Female (trans) | Y...
Genderqueer | Genderqueer | Yes ...
M | M | Yes | Yes
Enby | Enby | Don't know | Don't k...
Mal | Mal | No | No
Woman | Woman | Don't know | D...
Androgyne | Androgyne | No | No
queer | queer | No | No
Female | Female | No | No
Female (trans) | Female (trans) | N...
Neuter | Neuter | Don't know | Don...
F | F | No | No
Make | Make | Yes | Yes
Trans woman | Trans woman | Don...
fluid | fluid | Don't know | Don't kn...
woman | woman | Yes | Yes
Guy (-ish) ^_^ | Guy (-ish) ^_^ | Y...
All | All | No | No
No | No
No | Yes
Not sure | No
Not sure | Yes
Yes | No
Yes | Yes
18
14
7
6
6
6
4
3
109
81
39
35
34
19
15
7
5
5
3
3
3
2
2
86
37
25
20
15
11
10
10
9
29
11
10
9
7
3
88
61
33
30
25
24
16

DASHBOARD 3:

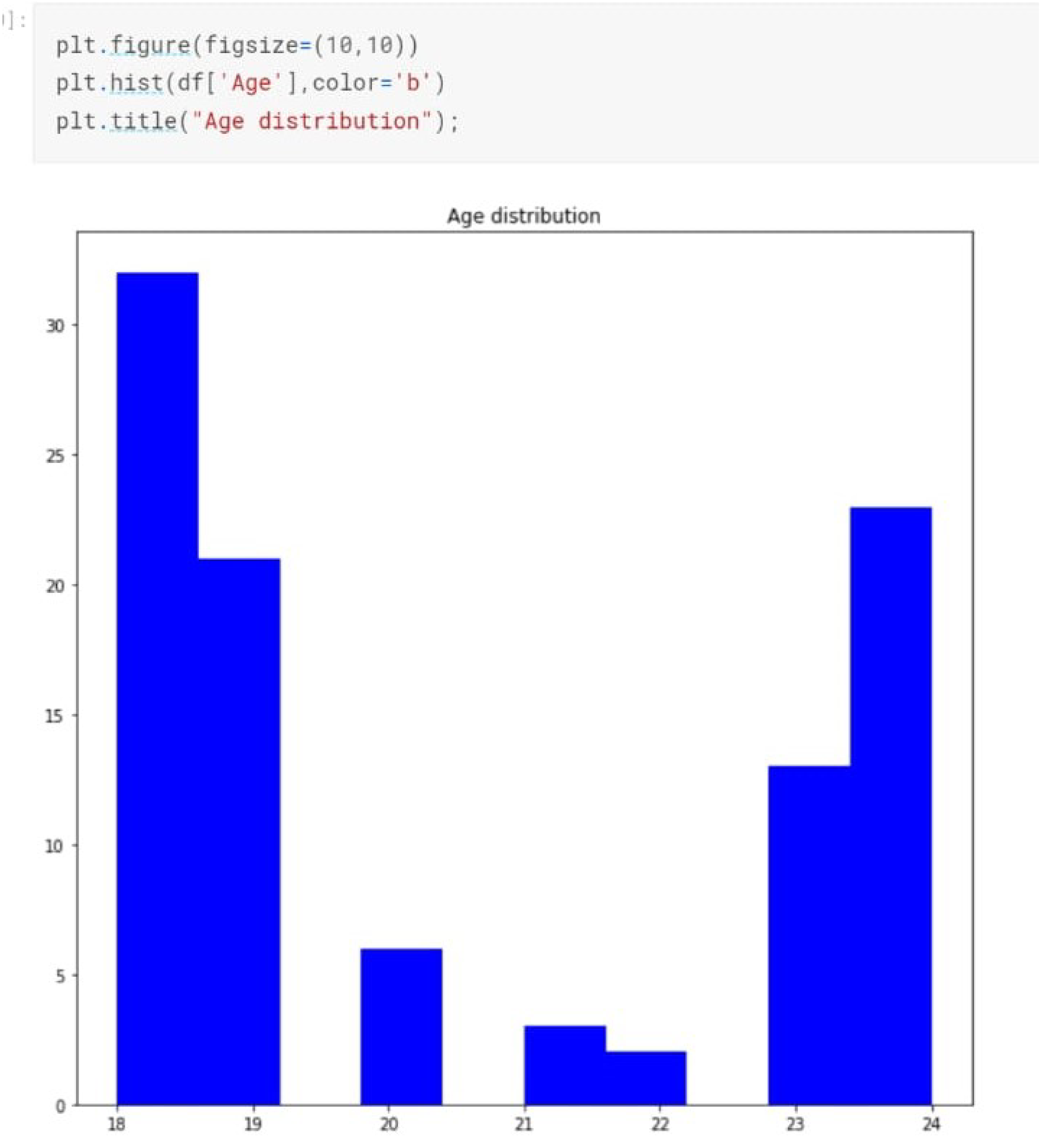


Tab 2

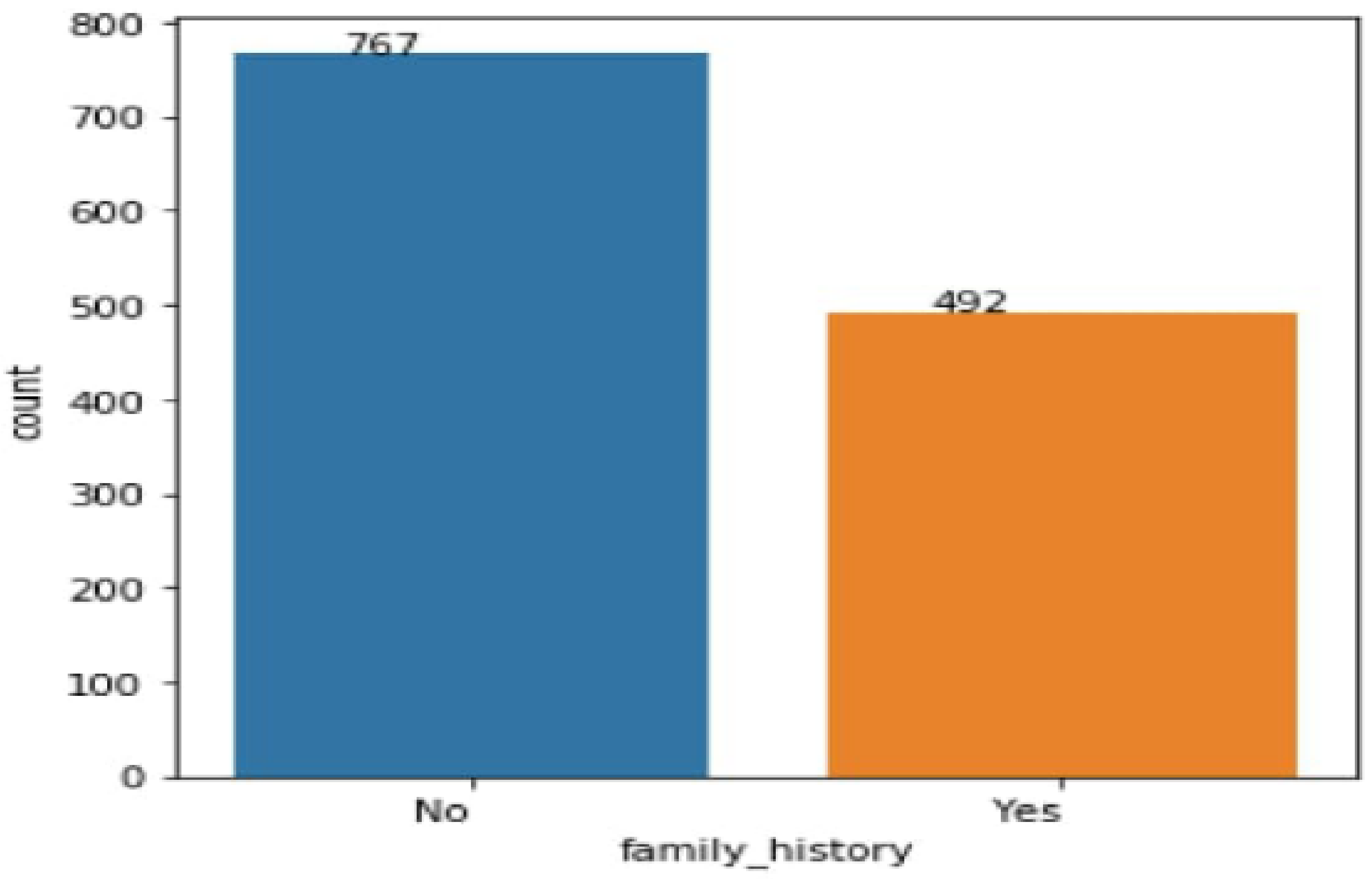
self_employed

treatment, family_history, work_interfere and no_employees hierarchy colored by remote_work and sized by state

DATA VISUALISATION:

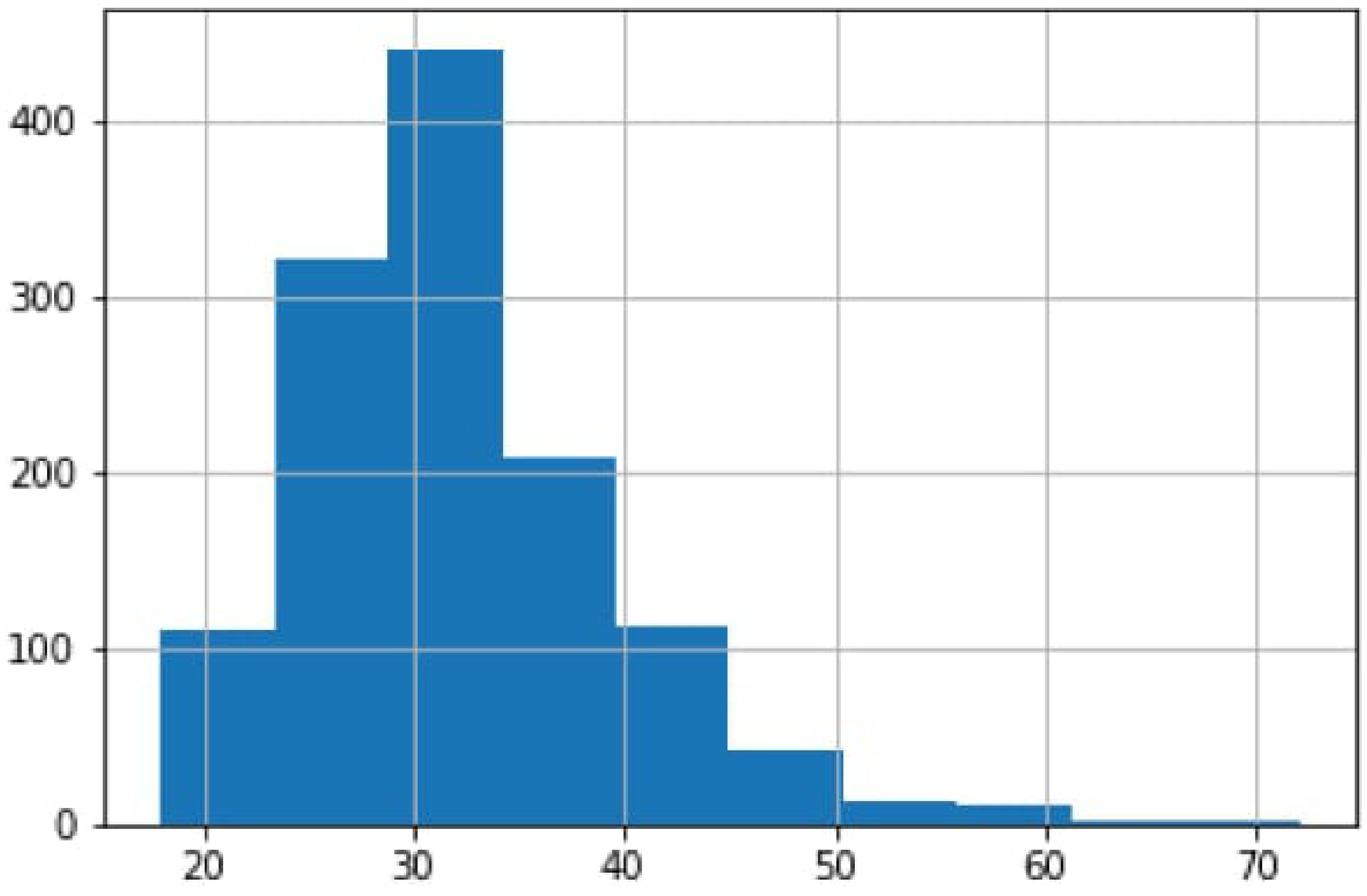


```
]:
plt.figure(figsize=(10,10))
plt.hist(df['Age'],color='b')
plt.title("Age distribution");
```

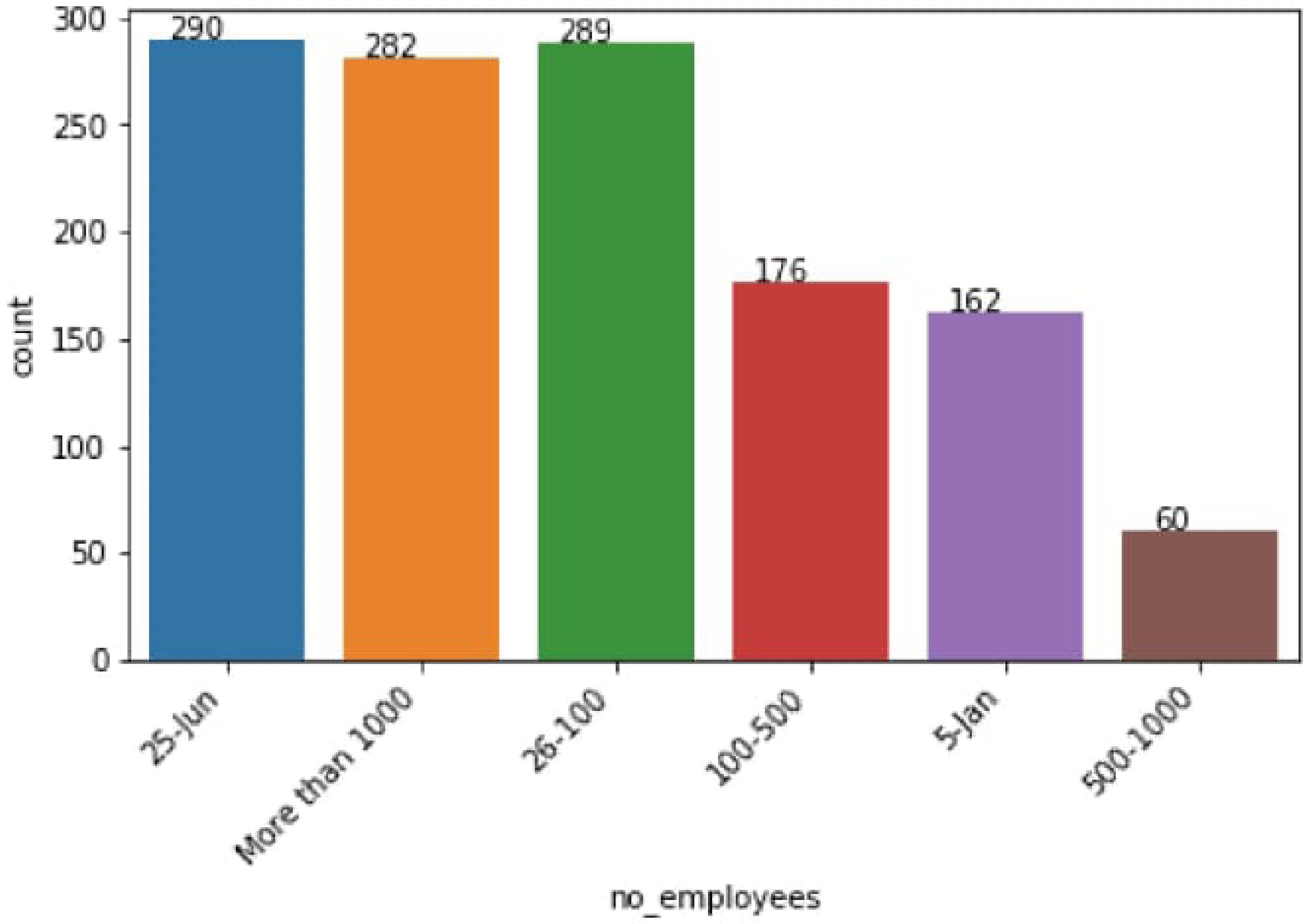


```
In [48]: plt.figure(figsize=(5,5))
         ax = sns.countplot(x='remote_work', data=df)
         ax.set_xticklabels(ax.get_xticklabels(),
                            horizontalalignment='right')
         for p in ax.patches:
            ax.annotate(p.get_height(), (p.get_x()+0.25, p.get_height()+0.01), ha='center')
```

```
In [39]: # We have outliers of 5years, 8years
         # With our domain knowledge, we can confidently go further to treat these as invalid.
         # Only 18 and above is accepted to be a legal tech employee.
         # I will replace them with the median.

In [40]: df.loc[df.Age<18, ["Age"]] = df["Age"].median()

In [41]: df["Age"].hist()

Out[41]: <matplotlib.axes._subplots.AxesSubplot at 0x1d203ddba58>
```

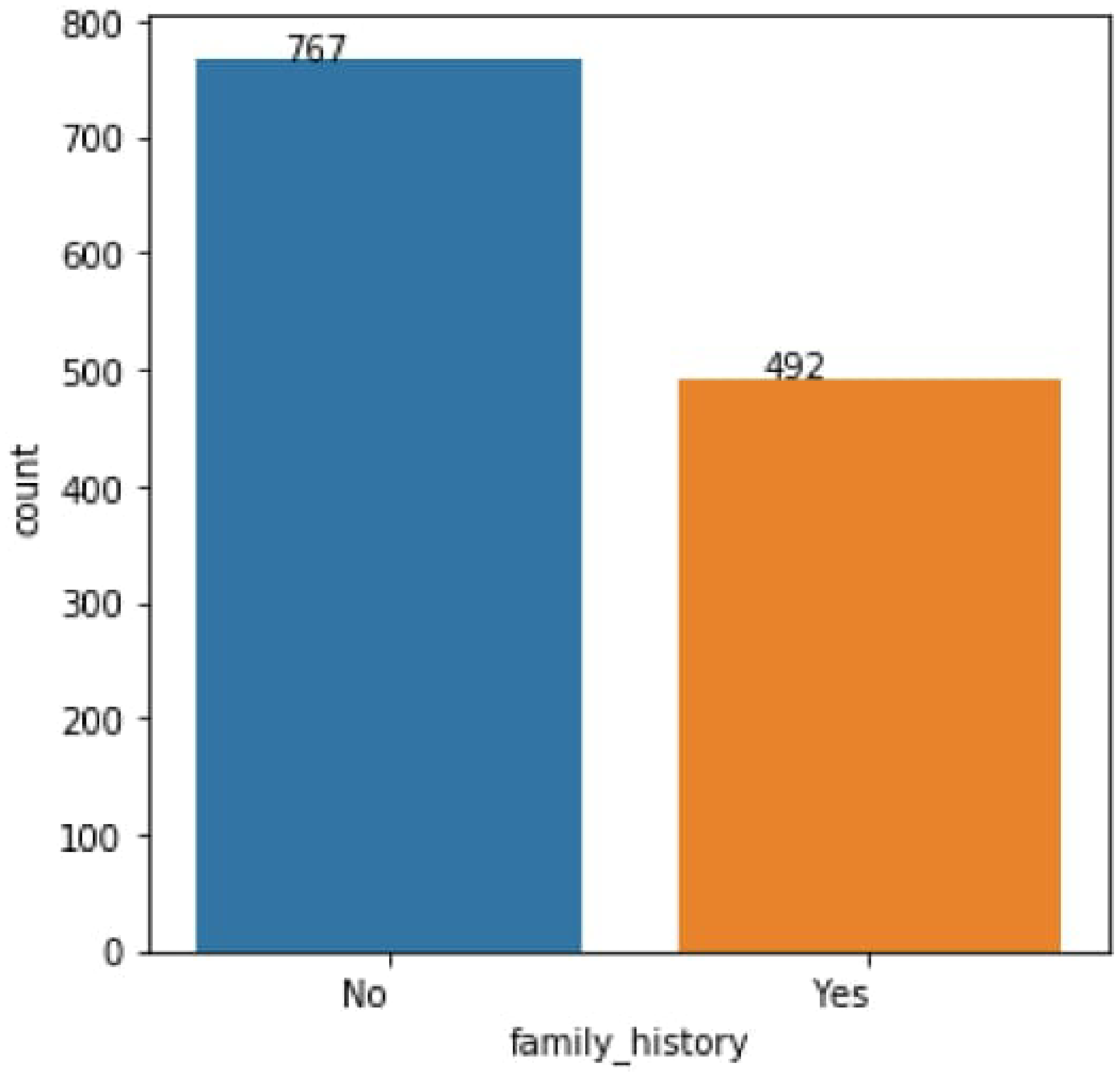


```
In [42]: sns.boxplot(x=df["Age"])

Out[42]: <matplotlib.axes._subplots.AxesSubplot at 0x1d2052d04e0>
```



```
In [46]: # This doesn't help us much, because number of employees isn't directly proportional wi
         # In fact, there are identical mental health issues with employees working in companies
             # 6 - 25
             # 25 - 100
             # More than 1000
```

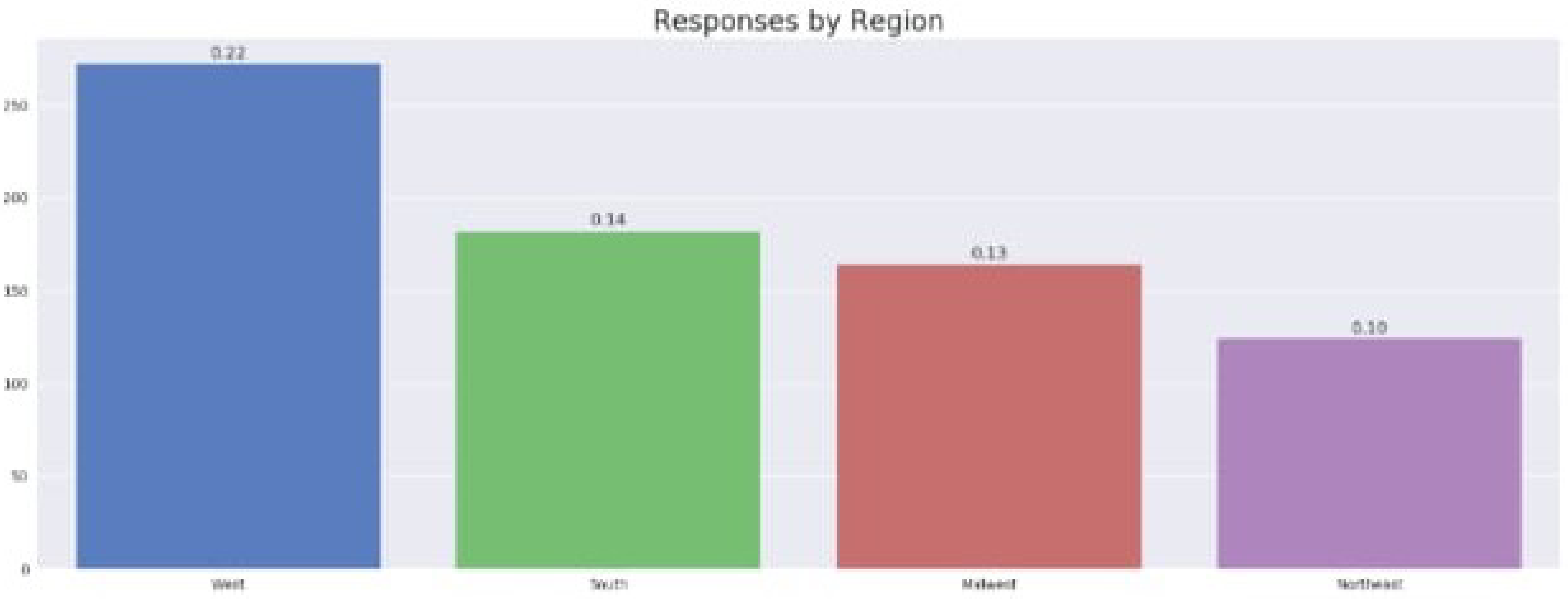
```
In [47]: plt.figure(figsize=(5,5))
         ax = sns.countplot(x='family_history', data=df)
         ax.set_xticklabels(ax.get_xticklabels(),
                            horizontalalignment='right')
         for p in ax.patches:
            ax.annotate(p.get_height(), (p.get_x()+0.25, p.get_height()+0.01), ha='center')
```



```
In [48]: plt.figure(figsize=(5,5))
         ax = sns.countplot(x='remote_work', data=df)
         ax.set_xticklabels(ax.get_xticklabels(),
                            horizontalalignment='right')
         for p in ax.patches:
            ax.annotate(p.get_height(), (p.get_x()+0.25, p.get_height()+0.01), ha='center')
```

[27]:

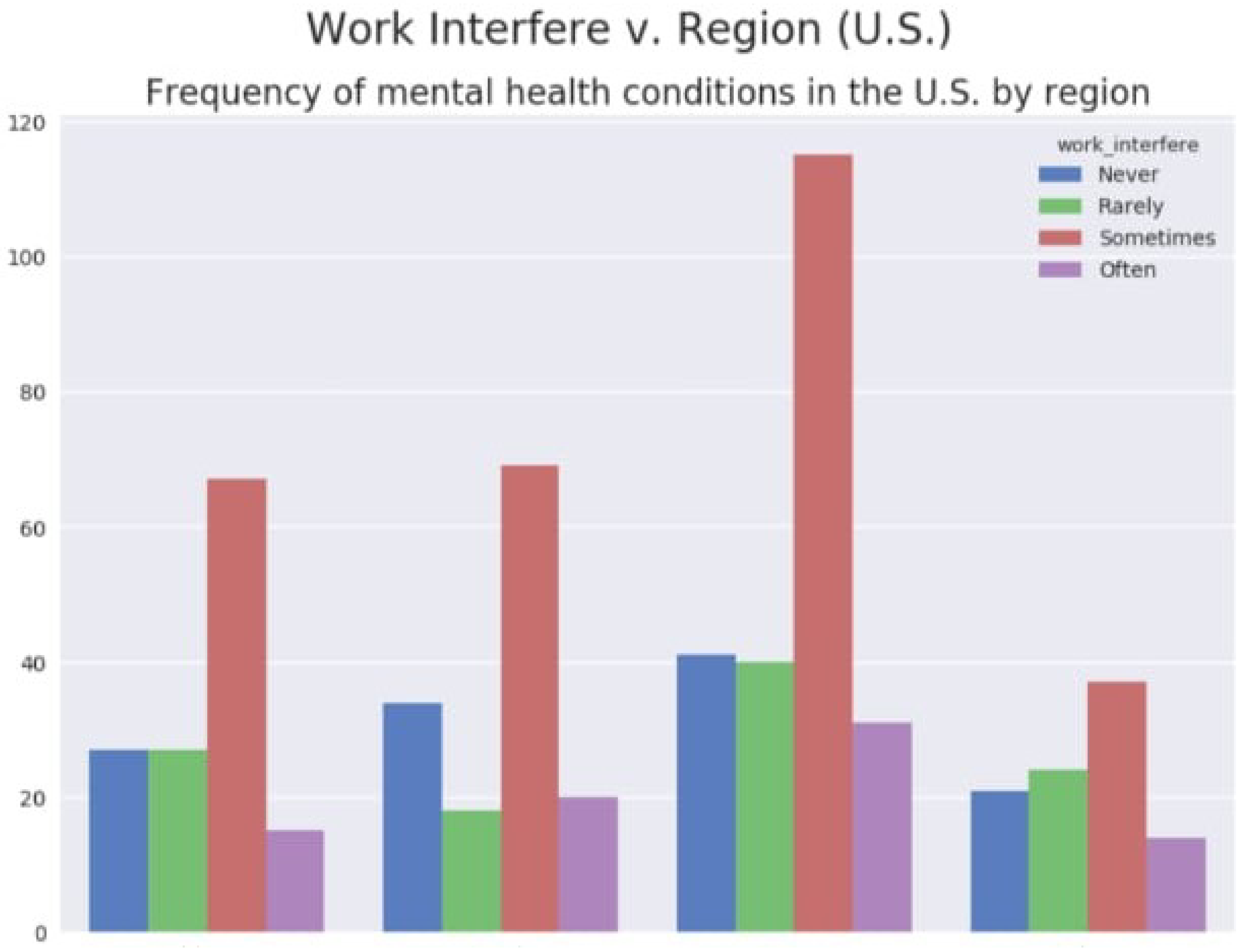
```
#### Survey Responses by region

total = float(len(mh))
plt.figure(figsize=(20, 7))
ax = sns.countplot(x='region', order = mh['region'].value_counts().index, data=m
h)
for p in ax.patches:
    height = p.get_height()
    ax.text(p.get_x()+p.get_width()/2.,
            height + 3,
            '{:1.2f}'.format(height/total),
            ha="center")
plt.title('Responses by Region', fontsize=20)
plt.xlabel('')
plt.ylabel('')
plt.show()
```

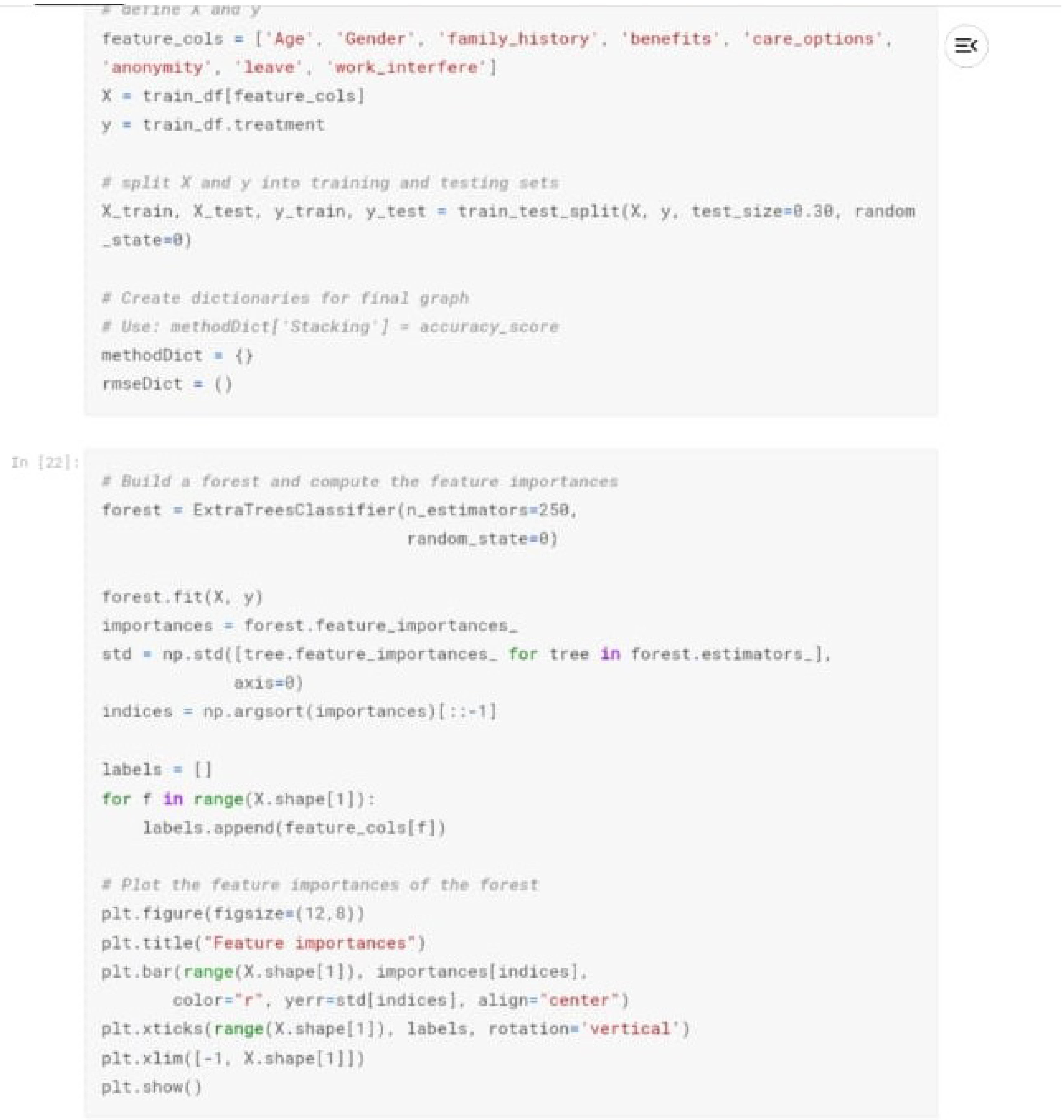


[28]:

[28]:

```
plt.figure(figsize=(10,7))
sns.countplot(x="region", hue="work_interfere", hue_order = ["Never", "Rarely", "Sometimes", "Often"], data=mh)
plt.suptitle("Work Interfere v. Region (U.S.)", fontsize=20)
plt.title("Frequency of mental health conditions in the U.S. by region", fontsize=16)
plt.xlabel("")
plt.ylabel("")
plt.show()
```

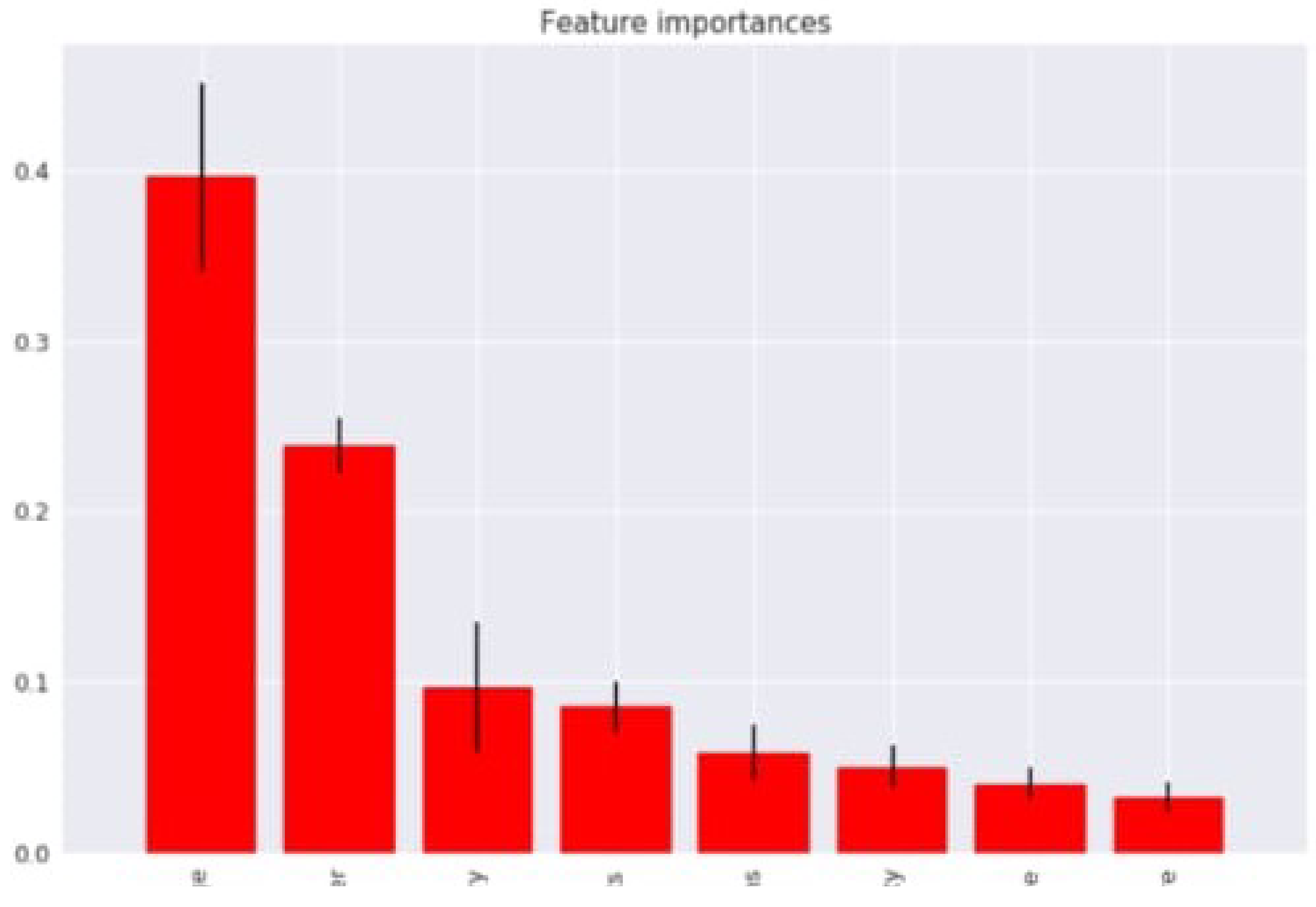




```
# Define X and y
feature_cols = ['Age', 'Gender', 'family_history', 'benefits', 'care_options',
'anonymity', 'leave', 'work_interfere']
X = train_df[feature_cols]
y = train_df.treatment

# split X and y into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.30, random
_state=0)

# Create dictionaries for final graph
# Use: methodDict['Stacking'] = accuracy_score
methodDict = {}
rmseDict = ()
```

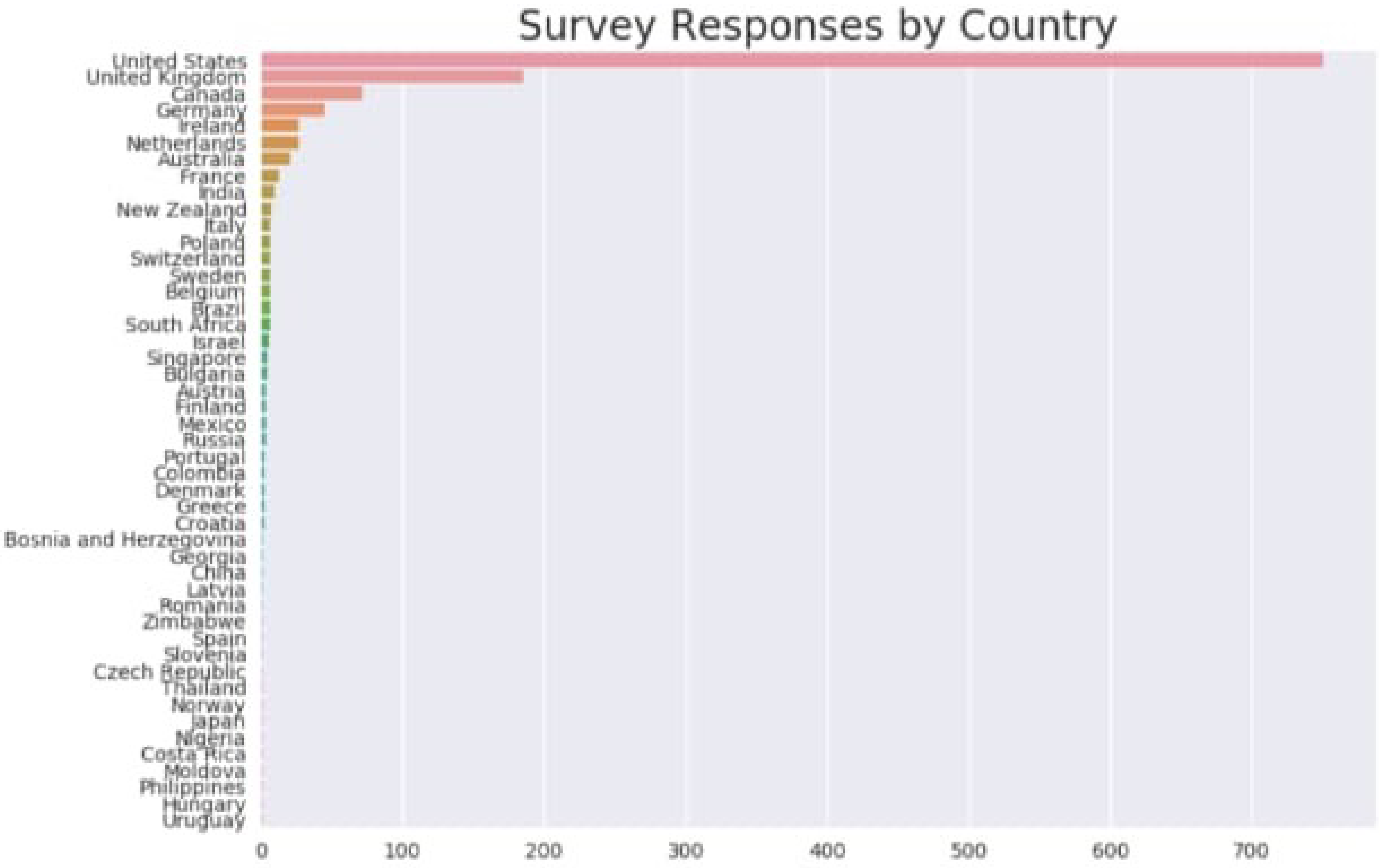
In [22]:

```
# Build a forest and compute the feature importances
forest = ExtraTreesClassifier(n_estimators=250,
                              random_state=0)

forest.fit(X, y)
importances = forest.feature_importances_
std = np.std([tree.feature_importances_ for tree in forest.estimators_],
             axis=0)
indices = np.argsort(importances)[::-1]

labels = []
for f in range(X.shape[1]):
    labels.append(feature_cols[f])

# Plot the feature importances of the forest
plt.figure(figsize=(12,8))
plt.title("Feature importances")
plt.bar(range(X.shape[1]), importances[indices],
       color="r", yerr=std[indices], align="center")
plt.xticks(range(X.shape[1]), labels, rotation='vertical')
plt.xlim([-1, X.shape[1]])
plt.show()
```



[24]:

```
# Create a frequency chart for "country"

plt.figure(figsize=(10, 7))
sns.countplot(y='country', order = mh['country'].value_counts().index, data=mh)
plt.title('Survey Responses by Country', fontsize=20)
plt.xlabel('')
plt.ylabel('')
plt.show()
```



[25]:

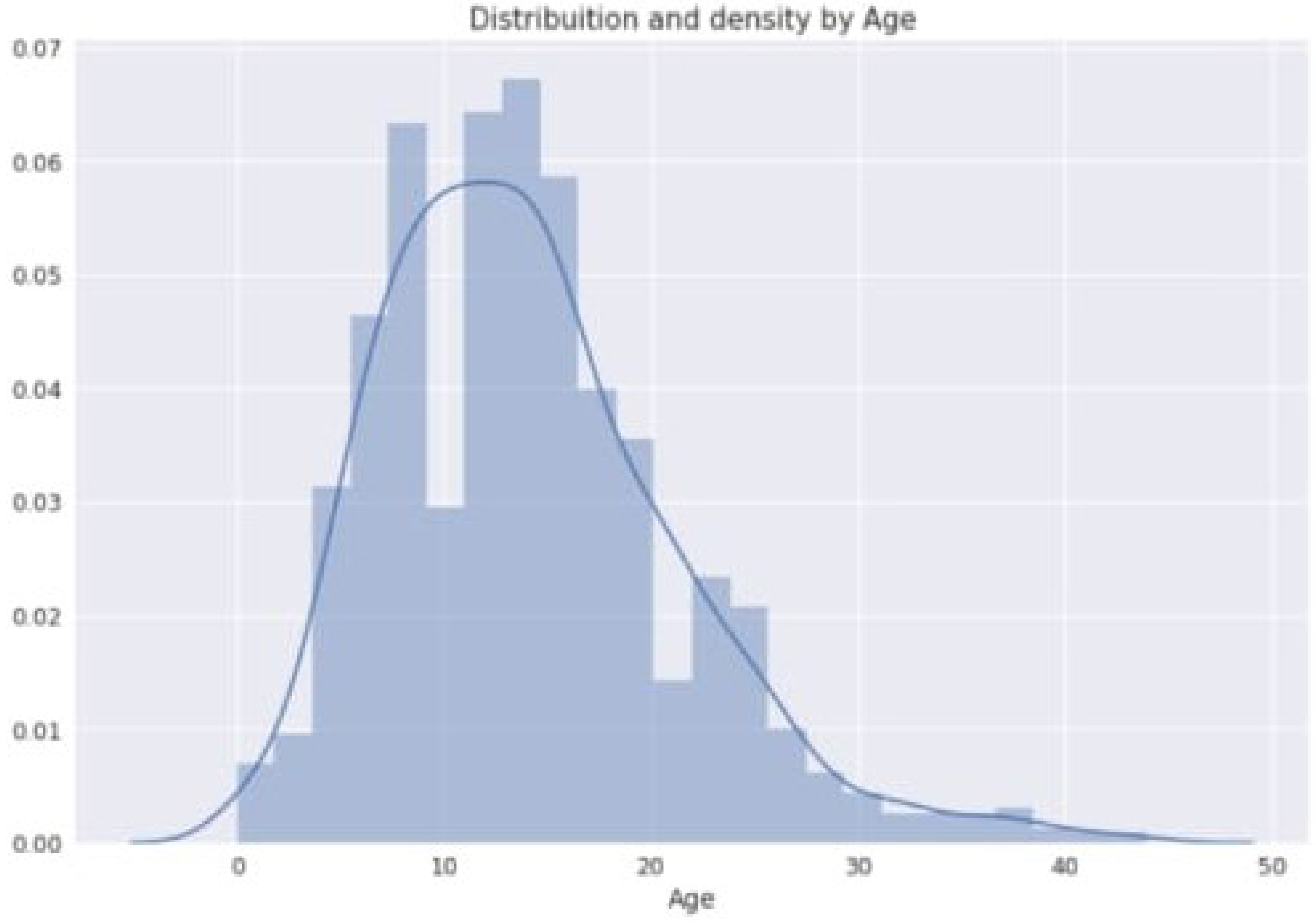
```
#### Survey Responses by state

total = float(len(mh))
plt.figure(figsize=(20, 7))
ax = sns.countplot(x='state', order = mh['state'].value_counts().index, data=mh)
for p in ax.patches:
    height = p.get_height()
    ax.text(p.get_x()+p.get_width()/2.,
            height + 3,
            '{:1.2f}'.format(height/total),
            ha="center")
plt.title('Responses by State', fontsize=20)
plt.xlabel('')
plt.ylabel('')
plt.show()
```

In [12]:

```
# Distribiution and density by Age
plt.figure(figsize=(12,8))
sns.distplot(train_df["Age"], bins=24)
plt.title("Distribuition and density by Age")
plt.xlabel("Age")
```

Out[12]:

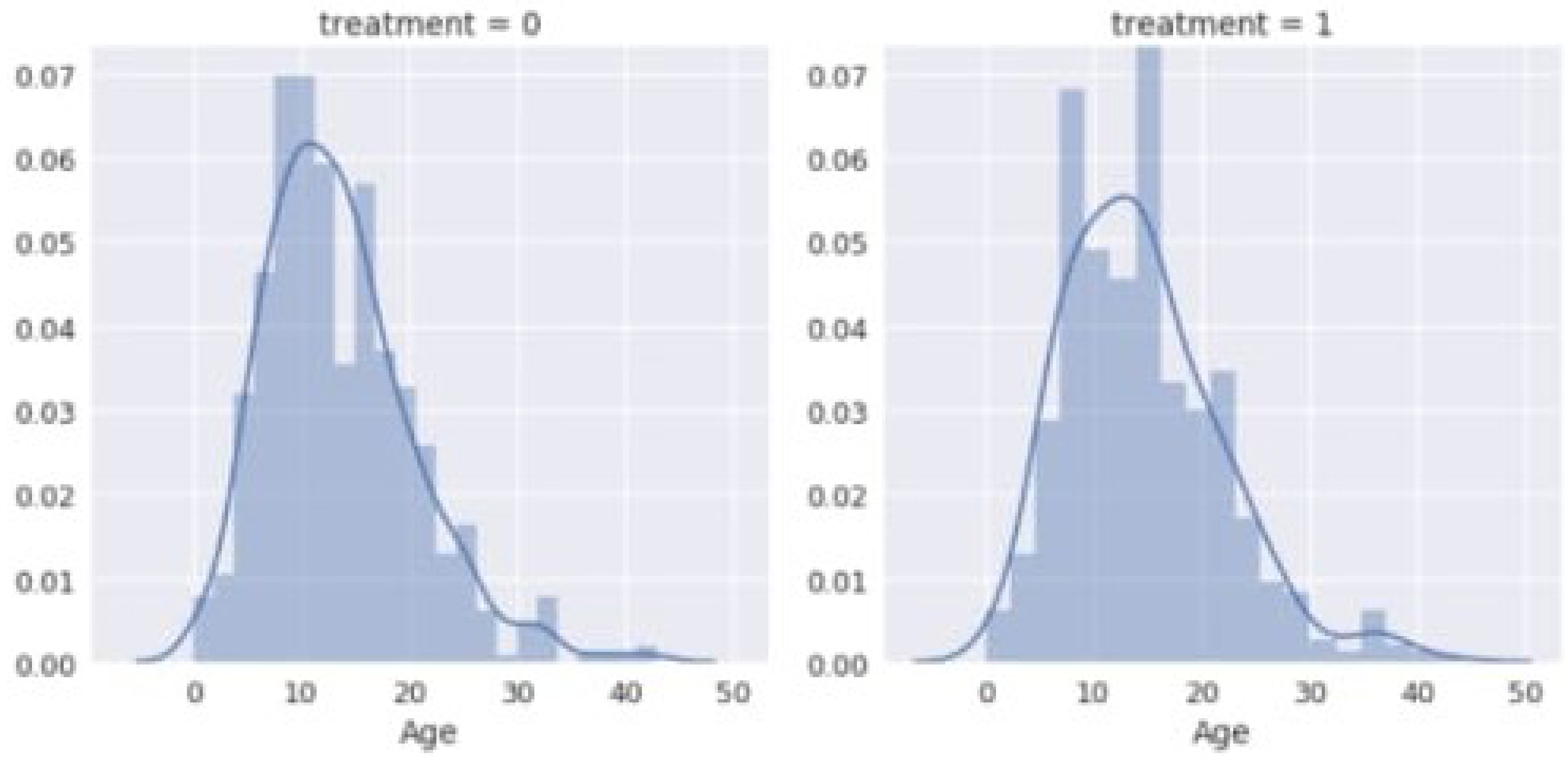
```
Text(0.5,0,'Age')
```

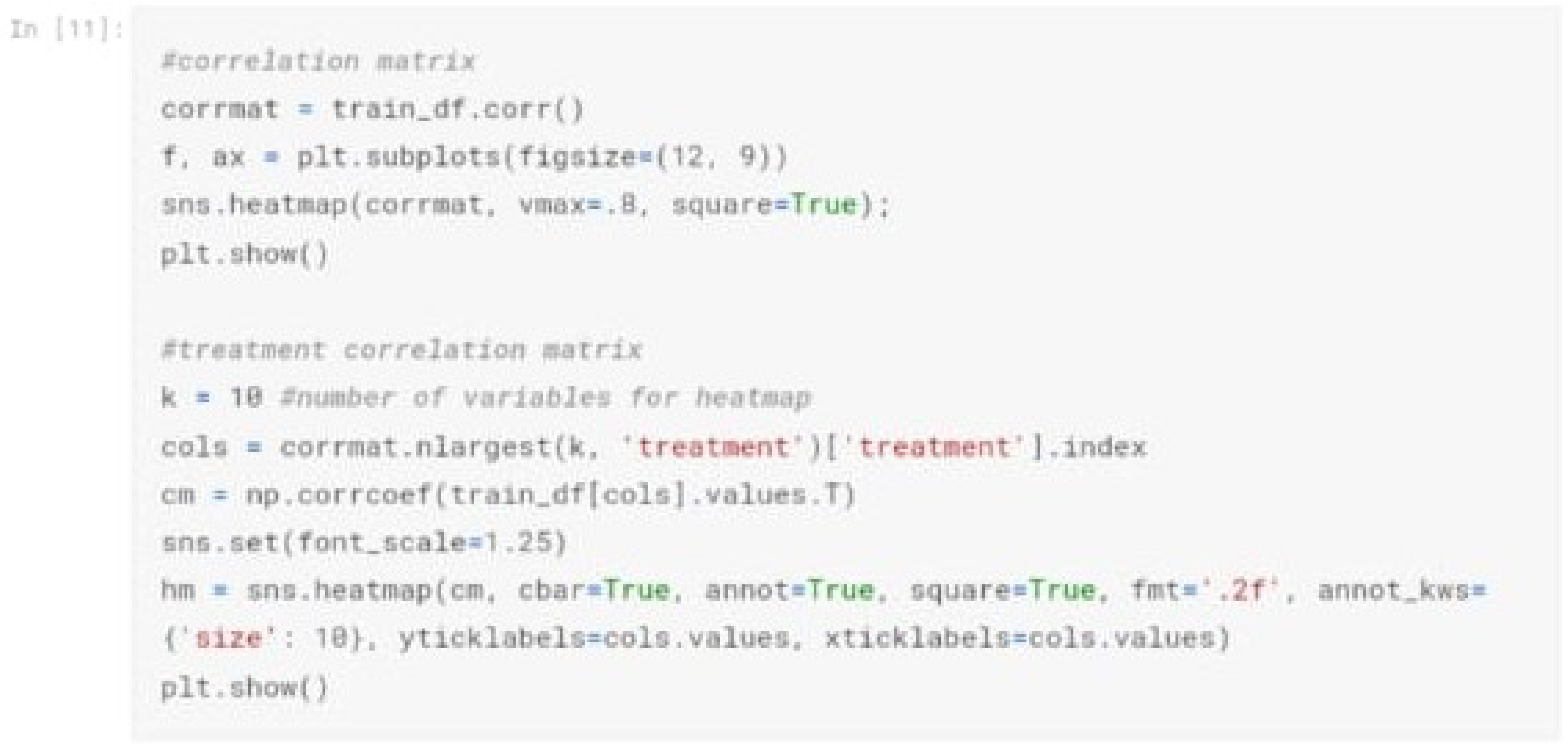


Separate by treatment

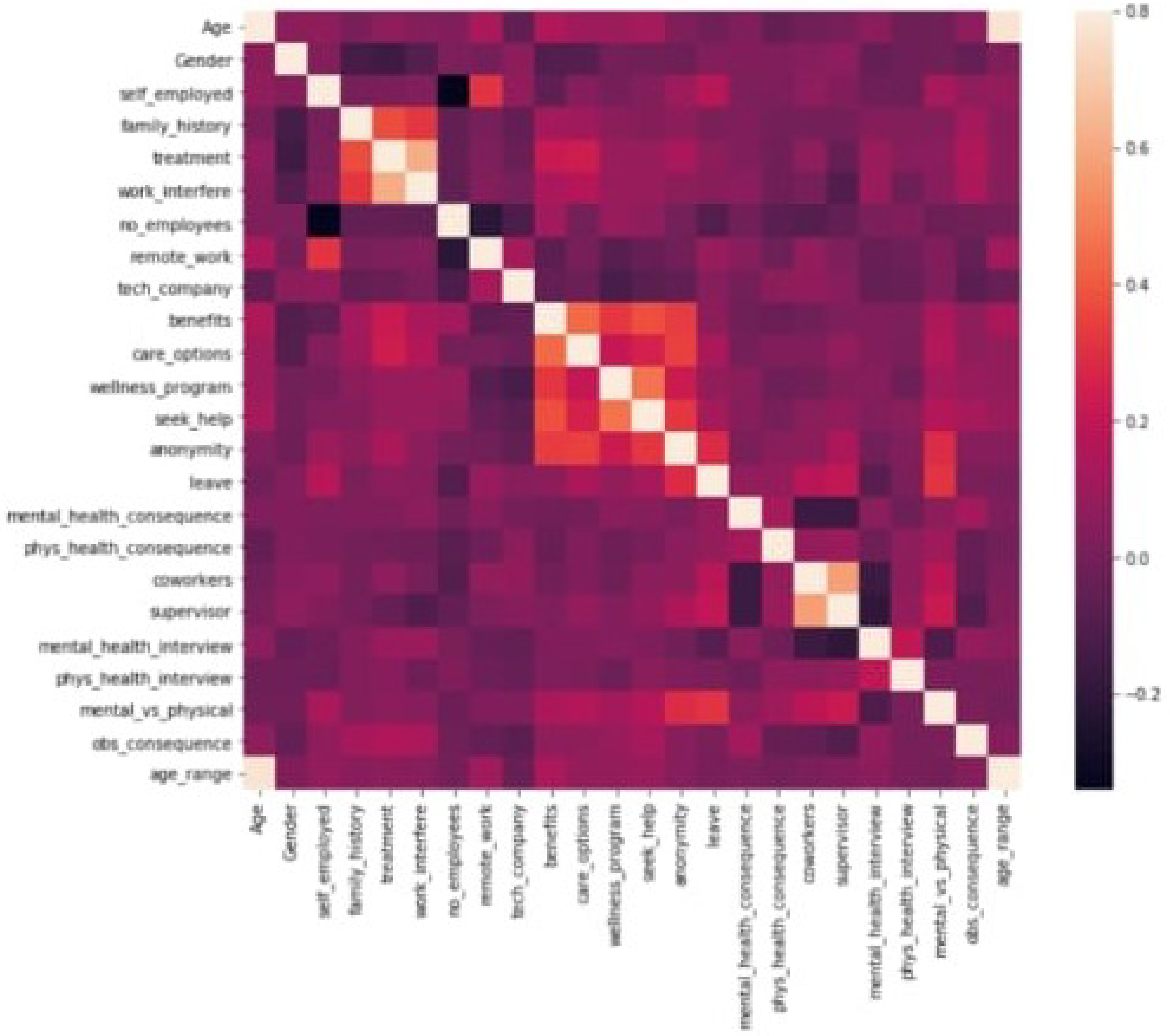
In [13]:

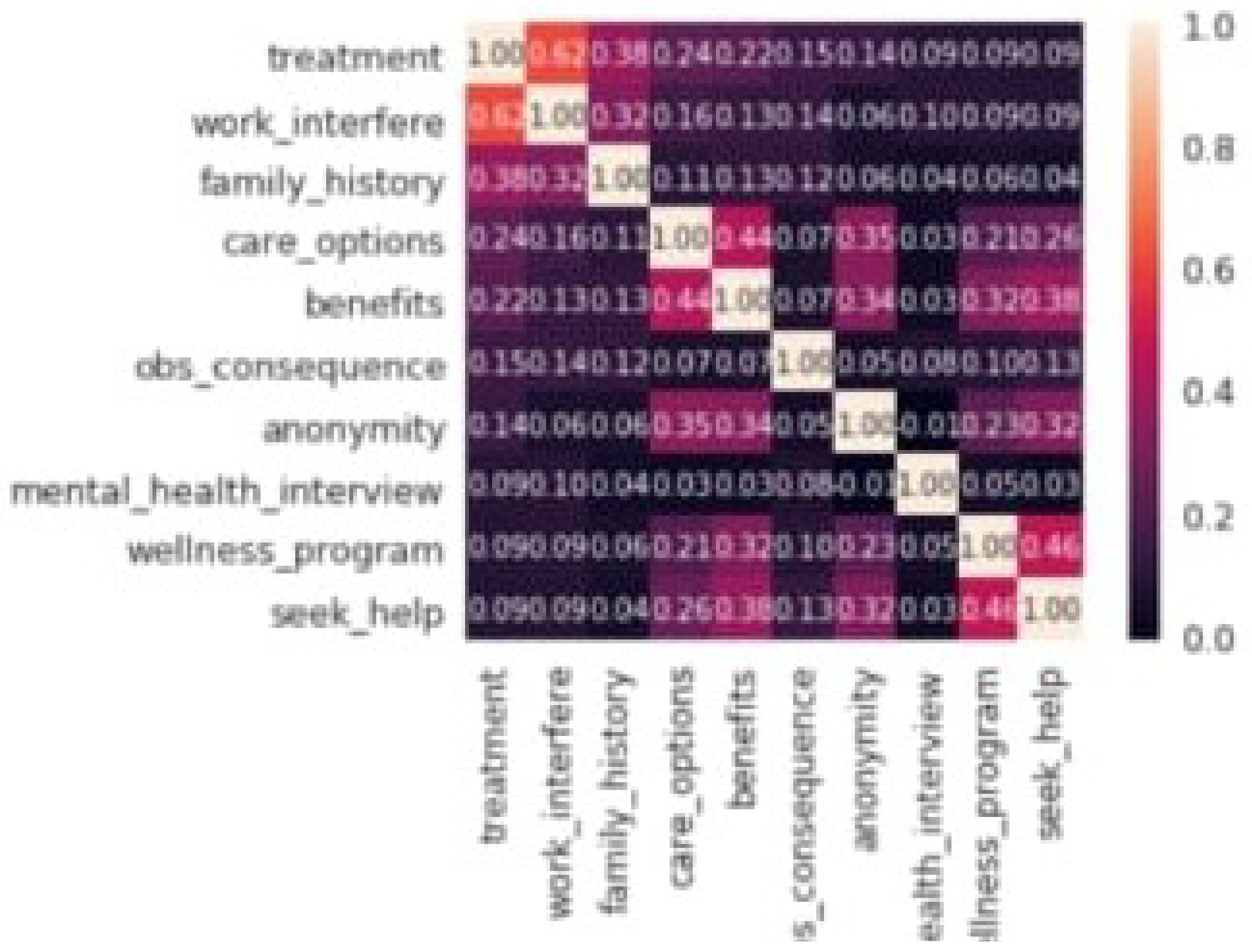
```
# Separate by treatment or not

g = sns.FacetGrid(train_df, col='treatment', size=5)
g = g.map(sns.distplot, "Age")
```





```
In [11]:
#correlation matrix
corrmat = train_df.corr()
f, ax = plt.subplots(figsize=(12, 9))
sns.heatmap(corrmat, vmax=.8, square=True);
plt.show()

#treatment correlation matrix
k = 10 #number of variables for heatmap
cols = corrmat.nlargest(k, 'treatment')['treatment'].index
cm = np.corrcoef(train_df[cols].values.T)
sns.set(font_scale=1.25)
hm = sns.heatmap(cm, cbar=True, annot=True, square=True, fmt='.2f', annot_kws=
{'size': 10}, yticklabels=cols.values, xticklabels=cols.values)
plt.show()
```



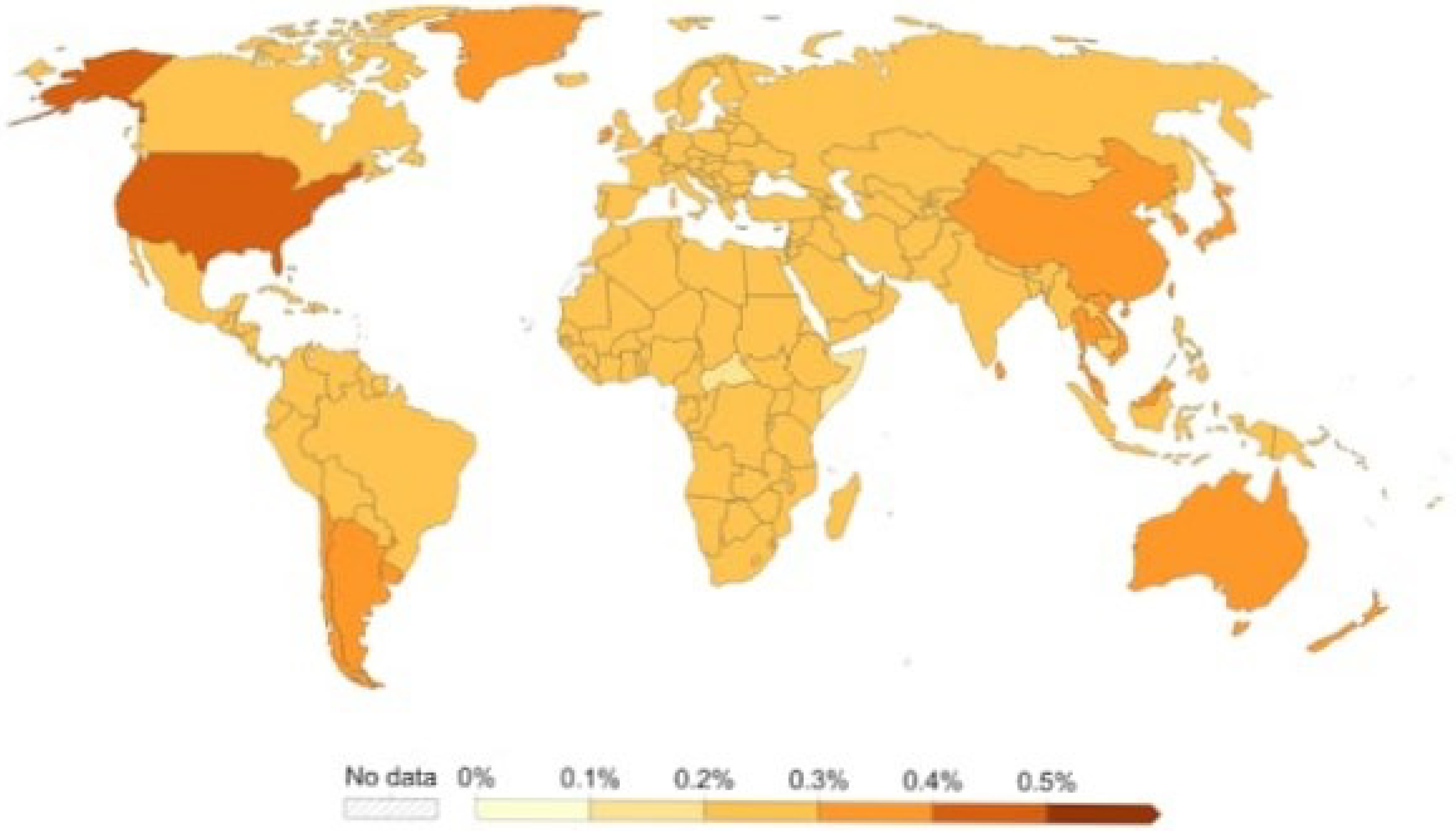


	treatment	work_interfere	family_history	care_options	benefits	s_consequence	anonymity	ealth_interview	llness_program	seek_help
treatment	1.00	0.62	0.38	0.24	0.22	0.15	0.14	0.09	0.09	0.09
work_interfere	0.62	1.00	0.32	0.16	0.13	0.14	0.06	0.10	0.09	0.09
family_history	0.38	0.32	1.00	0.11	0.13	0.12	0.06	0.04	0.06	0.04
care_options	0.24	0.16	0.11	1.00	0.44	0.07	0.35	0.03	0.21	0.26
benefits	0.22	0.13	0.13	0.44	1.00	0.07	0.34	0.03	0.32	0.38
obs_consequence	0.15	0.14	0.12	0.07	0.07	1.00	0.05	0.08	0.10	0.13
anonymity	0.14	0.06	0.06	0.35	0.34	0.05	1.00	-0.01	0.23	0.32
mental_health_interview	0.09	0.10	0.04	0.03	0.03	0.08	-0.01	1.00	0.05	0.03
wellness_program	0.09	0.09	0.06	0.21	0.32	0.10	0.23	0.05	1.00	0.46
seek_help	0.09	0.09	0.04	0.26	0.38	0.13	0.32	0.03	0.46	1.00

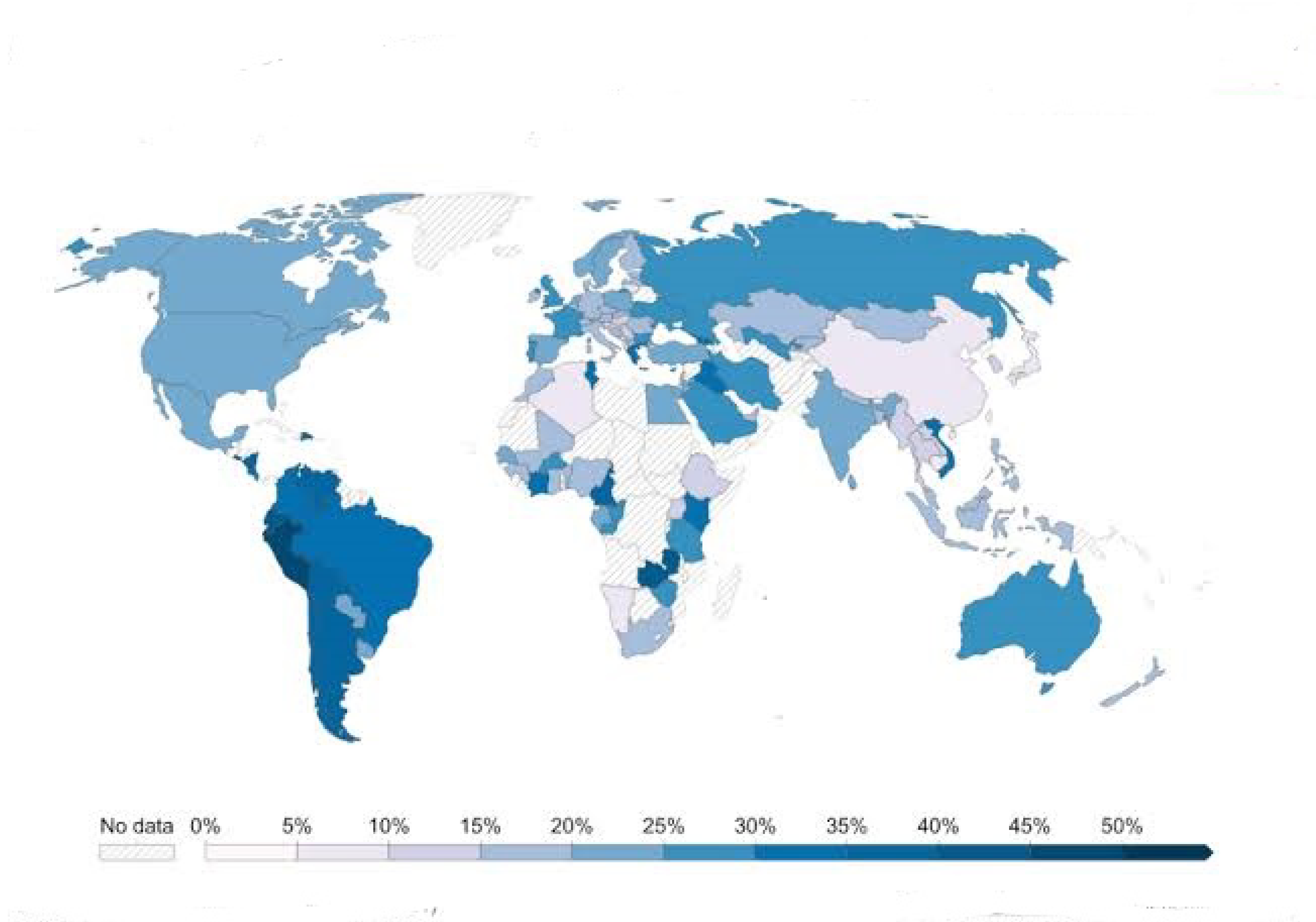
CONCLUSIONS:

- The potential for substantial adverse or benefits health effects or irreversible or catastrophic effects, even if the effects have a low likelihood
- Studies showing that social relationships both quality and quantity are having short and long-term effects on our health.
- Mental health in world level data analysis

Mental health analysis in 2019:



Mental health analysis in 2020:



- This are the mental well-being is important for the affected and the non-affected.