Smartphone Addiction

Discription:

The study aims to assess prevalence of smartphone addiction symptoms, and to ascertain whether depression or anxiety, independently, contributes to smartphone addiction level among a sample of university students, while adjusting simultaneously for important sociodemographic, academic, lifestyle, personality trait, and smartphone-related variables.

Methodology:

A random sample of 688 undergraduate university students (mean age = 20.64 ±1.88 years; 53% men) completed a survey composed of a) questions about socio-demographics, academics, lifestyle behaviors, personality type, and smartphone use-related variables; b) 26- item Smartphone Addiction Inventory (SPAI) Scale; and c) brief screeners of depression and anxiety (PHQ-2 and GAD-2), which constitute the two core DSM-IV items for major depressive disorder and generalized anxiety disorder, respectively.

Importing Libraries

In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

In [2]:
```
from plotly import __version__
from plotly.offline import download_plotlyjs, init_notebook_mode, plot, iplot

print(__version__) # requires version >= 1.9.0
import cufflinks as cf
init_notebook_mode(connected=True)
cf.go_offline()
```

```
4.14.3
```

Exploring the Data

In [3]:
```
df = pd.read_csv('final dataset.csv')
```

In [4]:
```
rawdf = pd.read_csv('S1Dataset.csv')
```

In [5]:
```
rawdf.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 688 entries, 0 to 687
Data columns (total 57 columns):
ID                                        688 non-null int64
AGE                                       688 non-null object
Gender                                    688 non-null object
Faculty                                   688 non-null object
Class                                     688 non-null object
WrkHrs_Wk                                 688 non-null object
sty_prsnt                                 688 non-null object
Prsnlty_type                              688 non-null object
Smoking                                   688 non-null object
Alcohol_drnk                              688 non-null object
Rlgn_Prctces                              688 non-null object
AgeStrt_useSmrtPhne                       688 non-null object
HrsSmrtPhnUse_Wkday                       681 non-null float64
CallFamMem                                688 non-null object
CallFrnds                                 688 non-null object
Txtng                                     688 non-null object
Entertainment                             688 non-null object
RdNews                                    688 non-null object
OthRsns                                   688 non-null object
Study_Purposes                            688 non-null object
ExcessveSmrtPhn_Use                       688 non-null object
TnseStp_Smrtphn_Use                       688 non-null object
hook_smrtPhn                              688 non-null object
Rstlss_NoSmrtphn                          688 non-null object
Vig_SmrtPhnUse                            688 non-null object
SmrtPhnUse_MreTmeMny                      688 non-null object
SlptLss4HrsMreTh1_SmrtPhnUse              688 non-null object
SameTmeIntrnt_Ngtv_Relations              688 non-null object
UpstStp_SmrtPhnUse                        688 non-null object
RcntSigIncTime_SmrtPhnUseWk               688 non-null object
FailCntrlImplse_smrtphnUse                688 non-null object
FavorSmrtPhn_SpndTimefrnds                688 non-null object
PainBckEye_ExcSmrtPhnUse                  688 non-null object
FrstThghtSmrtphnUse_WakeUp                688 non-null object
NgtveSchlJob_SmrtPhnUse                   688 non-null object
MissStp_SmrtPhnUse                        688 non-null object
DcreasdFmlyIntrction_SmrtPhnUse           688 non-null object
DcreasdHobbies_SmrtPhnUse                 688 non-null object
UrgeSmrtPhnUse_OnceStpUse                 688 non-null object
LifeJoylss_NoSmrtPhn                      688 non-null object
NgtvePhysHlthEffcts_SmrtPhnUse            688 non-null object
Spnd_LsstimeSmrtPhn_EffortsUselss         688 non-null object
DcreasdSlpTimeQulty_SmrtPhnUse            688 non-null object
IncrsdTimeSmrtPhnUse_SameSatsfction       688 non-null object
CannotHveMeal_NosmrtPhn                   688 non-null object
TiredDaytime_latenightSmrtPhnUse          688 non-null object
Compulsive_Behavior                       688 non-null object
Functional_Impairment                     688 non-null object
Withdrawal                                688 non-null object
Tolerance                                 688 non-null object
TotAddiction_Score                        688 non-null object
Lttl_IntrstDoingThngs                     688 non-null object
Feel_Deprssd                              688 non-null object
Depression_score                          688 non-null object
```

```
Feel_anxious                            688 non-null object
NotAble_Stpworry                        688 non-null object
Anxiety_score                           688 non-null object
dtypes: float64(1), int64(1), object(55)
memory usage: 306.5+ KB
```

In [6]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 266 entries, 0 to 265
Data columns (total 14 columns):
Prsnlty_type                          266 non-null int64
HrsSmrtPhnUse_Wkday                   266 non-null int64
CallFamMem                            266 non-null int64
CallFrnds                             266 non-null int64
Txtng                                 266 non-null int64
Entertainment                         266 non-null int64
NgtvePhysHlthEffcts_SmrtPhnUse        266 non-null int64
DcreasdSlpTimeQulty_SmrtPhnUse        266 non-null int64
TiredDaytime_latenightSmrtPhnUse      266 non-null int64
Compulsive_Behavior                   266 non-null int64
Functional_Impairment                 266 non-null int64
TotAddiction_Score                    266 non-null int64
Depression_score                      266 non-null int64
Anxiety_score                         266 non-null int64
dtypes: int64(14)
memory usage: 29.2 KB
```

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

Out[7]:

	Prsnlty_type	HrsSmrtPhnUse_Wkday	CallFamMem	CallFrnds	Txtng	Entertainment	NgtvePhy
0	1	1	1	1	1	1	
1	0	1	1	1	1	1	
2	0	1	1	1	1	1	
3	0	1	0	1	1	1	
4	0	1	0	0	1	1	

In [8]:
```
df['CallFamMem'] = df['CallFamMem']+df['CallFrnds']+df['Txtng']
```

In [9]:
```
df = df.rename(columns={'CallFamMem': 'Socializing'})
```

In [10]:
```
df.drop('CallFrnds',axis=1,inplace=True)
```

In [11]:
```
df.drop('Txtng',axis=1,inplace=True)
```

In [12]:
```
df.head()
```

Out[12]:

	Prsnlty_type	HrsSmrtPhnUse_Wkday	Socializing	Entertainment	NgtvePhysHlthEffcts_SmrtPhn
0	1	1	3	1	
1	0	1	3	1	
2	0	1	3	1	
3	0	1	2	1	
4	0	1	1	1	

In [ ]:
```
```

In [13]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 266 entries, 0 to 265
Data columns (total 12 columns):
Prsnlty_type                          266 non-null int64
HrsSmrtPhnUse_Wkday                   266 non-null int64
Socializing                           266 non-null int64
Entertainment                         266 non-null int64
NgtvePhysHlthEffcts_SmrtPhnUse        266 non-null int64
DcreasdSlpTimeQulty_SmrtPhnUse        266 non-null int64
TiredDaytime_latenightSmrtPhnUse      266 non-null int64
Compulsive_Behavior                   266 non-null int64
Functional_Impairment                 266 non-null int64
TotAddiction_Score                    266 non-null int64
Depression_score                      266 non-null int64
Anxiety_score                         266 non-null int64
dtypes: int64(12)
memory usage: 25.0 KB
```

In [14]:
```
df.columns
```

Out[14]:
```
Index(['Prsnlty_type', 'HrsSmrtPhnUse_Wkday', 'Socializing', 'Entertainment',
       'NgtvePhysHlthEffcts_SmrtPhnUse', 'DcreasdSlpTimeQulty_SmrtPhnUse',
       'TiredDaytime_latenightSmrtPhnUse', 'Compulsive_Behavior',
       'Functional_Impairment', 'TotAddiction_Score', 'Depression_score',
       'Anxiety_score'],
      dtype='object')
```

In [ ]:
```
```

In [15]:
```
df.tail()
```

Out[15]:

	Prsnlty_type	HrsSmrtPhnUse_Wkday	Socializing	Entertainment	NgtvePhysHlthEffcts_SmrtPl
261	0	1	1	1	
262	1	1	3	1	
263	1	0	3	1	
264	1	1	3	1	
265	0	0	3	1	

In [ ]:
```
```

In [16]:
```
df.describe()
```

Out[16]:

	Prsnlty_type	HrsSmrtPhnUse_Wkday	Socializing	Entertainment	NgtvePhysHlthEffcts_Smr
count	266.000000	266.000000	266.000000	266.000000	26
mean	0.364662	0.819549	2.184211	0.736842	
std	0.482242	0.385288	1.049661	0.441177	
min	0.000000	0.000000	0.000000	0.000000	
25%	0.000000	1.000000	1.000000	0.000000	
50%	0.000000	1.000000	3.000000	1.000000	
75%	1.000000	1.000000	3.000000	1.000000	
max	1.000000	1.000000	3.000000	1.000000	

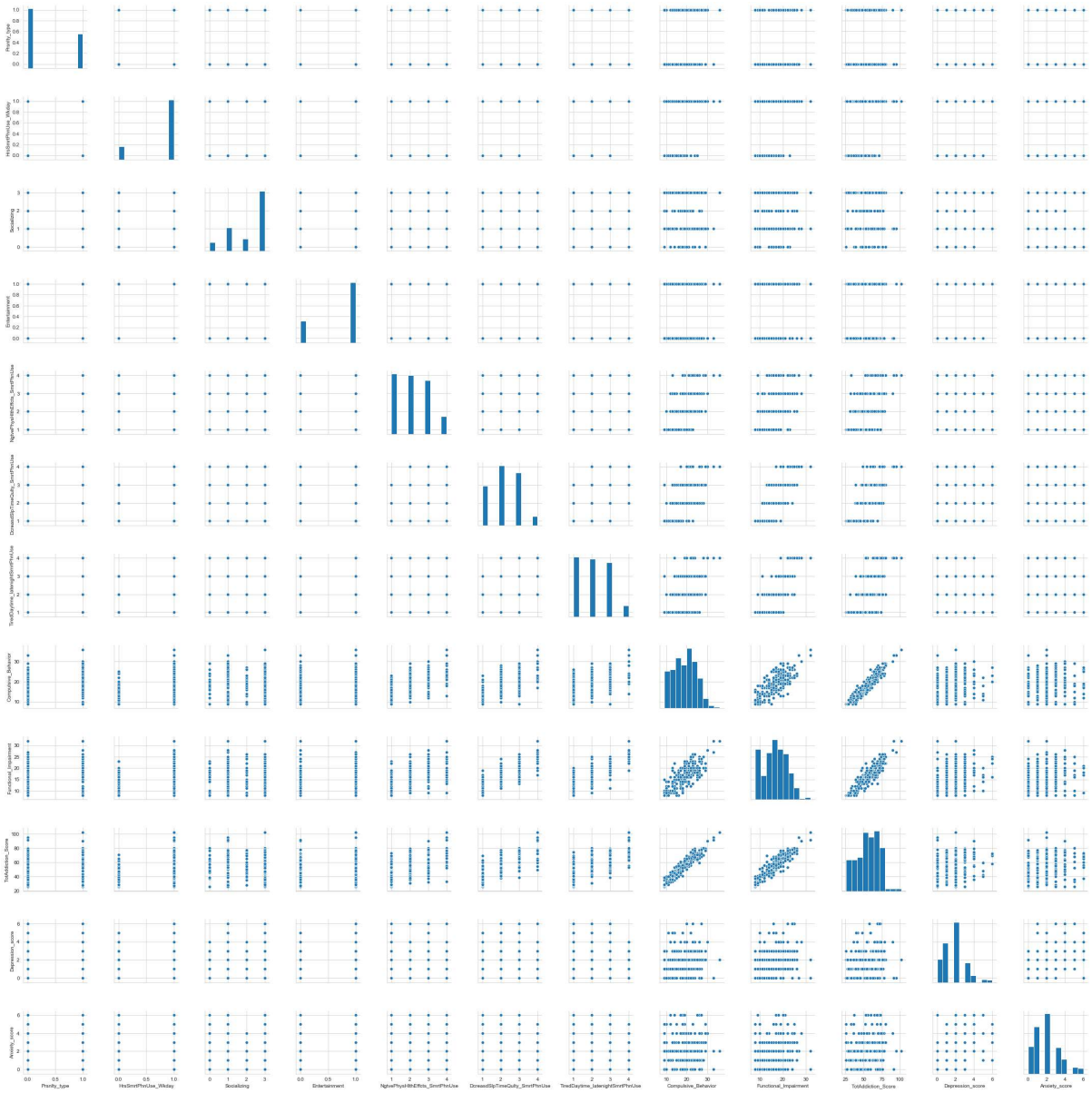
In [17]:
```
df.columns
```

Out[17]:
```
Index(['Prsnlty_type', 'HrsSmrtPhnUse_Wkday', 'Socializing', 'Entertainment',
       'NgtvePhysHlthEffcts_SmrtPhnUse', 'DcreasdSlpTimeQulty_SmrtPhnUse',
       'TiredDaytime_latenightSmrtPhnUse', 'Compulsive_Behavior',
       'Functional_Impairment', 'TotAddiction_Score', 'Depression_score',
       'Anxiety_score'],
      dtype='object')
```

Exploratory Data Analysis

In [18]:
```
sns.set_style('whitegrid')
sns.pairplot(df)
```

Out[18]: <seaborn.axisgrid.PairGrid at 0x1b29ab34978>



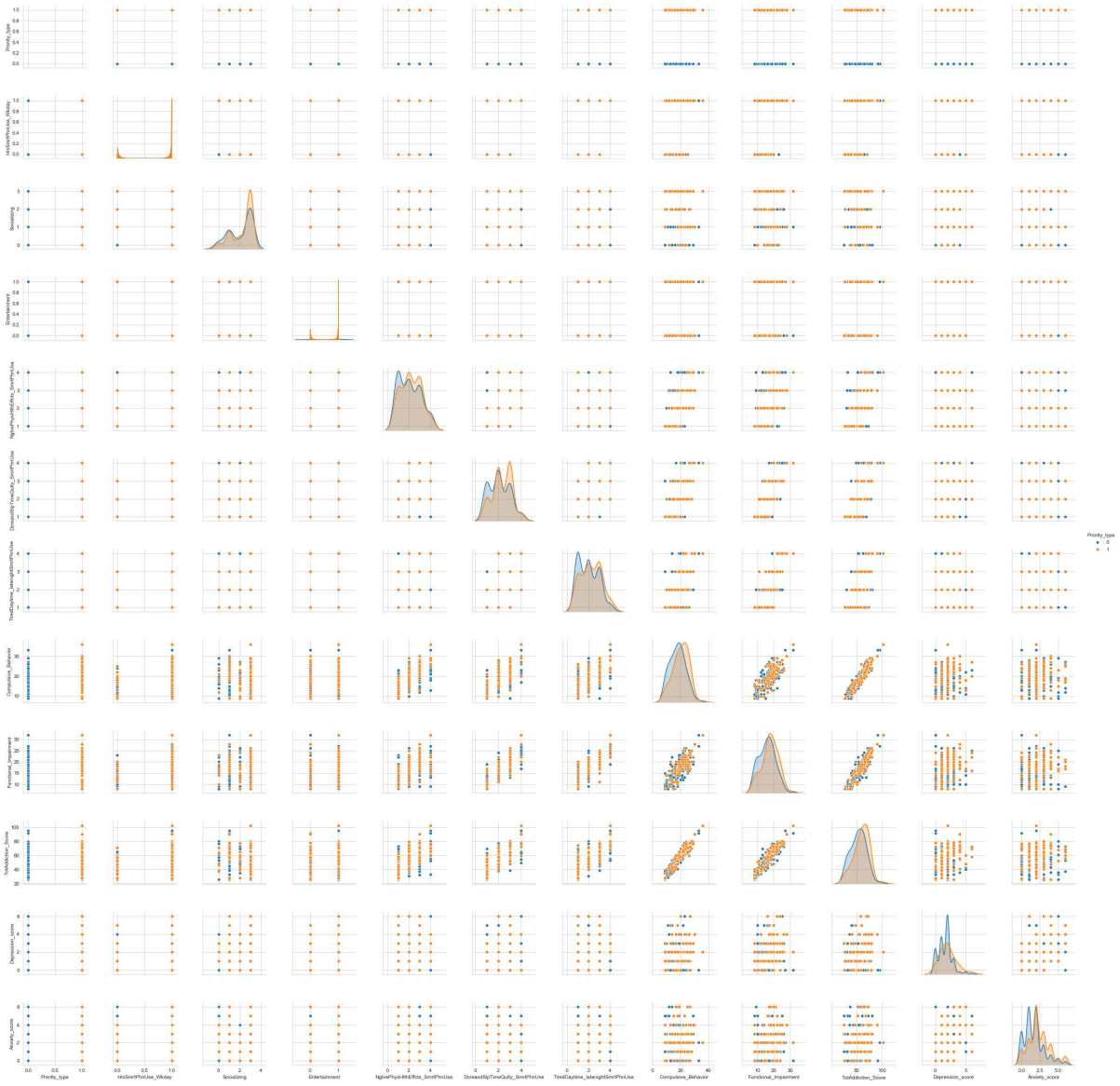
Considered Personality Types:

A personality are more competitive, ambitious, impatient, anxious, aggressive, and more likely to be workaholic. Individuals with type B personality are their counterparts

In [19]:
```
sns.pairplot(df, hue="Prsnlty_type")
```

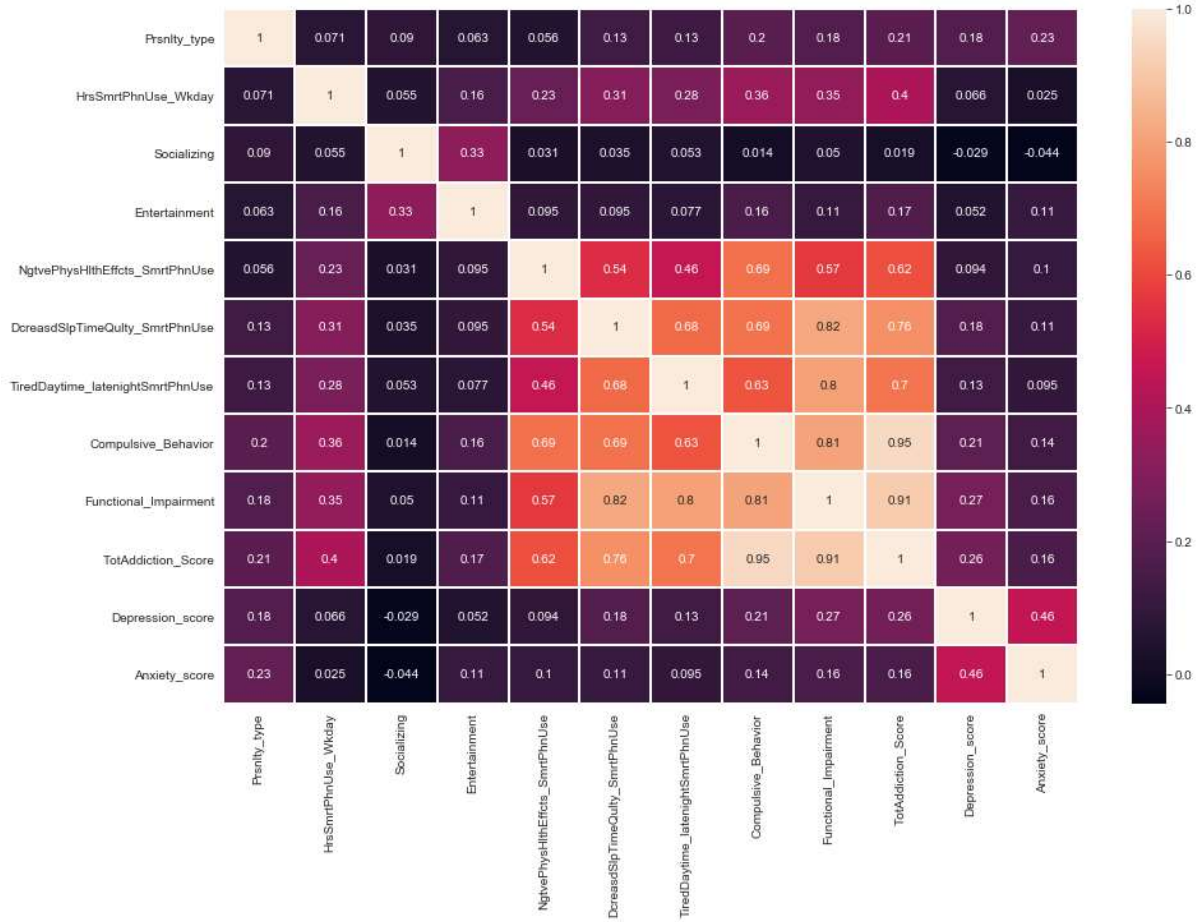
```
C:\Users\Asus\Anaconda3\lib\site-packages\statsmodels\nonparametric\kde.py:48
7: RuntimeWarning:

invalid value encountered in true_divide

C:\Users\Asus\Anaconda3\lib\site-packages\statsmodels\nonparametric\kdetools.
py:34: RuntimeWarning:

invalid value encountered in double_scalars
```

Out[19]: <seaborn.axisgrid.PairGrid at 0x1b2a0dca9b0>



In [20]:
```
plt.figure(figsize = (15,10))
sns.heatmap(df.corr(), annot = True,linewidths=1, linecolor='white',)
```

Out[20]: <matplotlib.axes._subplots.AxesSubplot at 0x1b2a7c93048>



Results (research paper):

1) Revalence rates of smartphone-related compulsive behavior, functional impairment, toler□ance and withdrawal symptoms were substantial.

2) Depression and anxiety scores emerged as independent positive predictors of smartphone addiction, with depression score being a more powerful predictor compared to anxiety score

3) Whereas gender, residence, work hours per week, faculty, academic performance (GPA), lifestyle habits (smoking and alcohol drinking), and religious practice did not associate with smartphone addiction score;

4) Personality type A, class (year 2 vs. year 3), younger age at first smartphone use, excessive use during a weekday, using it for entertainment and not using it to call family members, and having depression or anxiety, showed statistically significant associations with smartphone addiction.

Conclusion

Several independent positive predictors of smartphone addiction emerged including depression and anxiety. It could be that young adults with personality type A experiencing high stress level and low mood may lack positive stress coping mechanisms and mood management techniques and are thus highly susceptible to smartphone addiction.

Training a Linear Regression Model

X and y arrays

In [21]:
```
df.columns
```

Out[21]:
```
Index(['Prsnlty_type', 'HrsSmrtPhnUse_Wkday', 'Socializing', 'Entertainment',
       'NgtvePhysHlthEffcts_SmrtPhnUse', 'DcreasdSlpTimeQulty_SmrtPhnUse',
       'TiredDaytime_latenightSmrtPhnUse', 'Compulsive_Behavior',
       'Functional_Impairment', 'TotAddiction_Score', 'Depression_score',
       'Anxiety_score'],
      dtype='object')
```

In [22]:
```
X = df[['Prsnlty_type', 'HrsSmrtPhnUse_Wkday', 'Socializing', 'Entertainment',
       'NgtvePhysHlthEffcts_SmrtPhnUse', 'DcreasdSlpTimeQulty_SmrtPhnUse',
       'TiredDaytime_latenightSmrtPhnUse', 'Compulsive_Behavior',
       'Functional_Impairment', 'TotAddiction_Score']]
y = df['Depression_score']
```

Train Test Split

Now let's split the data into a training set and a testing set. We will train out model on the training set and then use the test set to evaluate the model.

In [23]:
```
from sklearn.model_selection import train_test_split
```

In [24]:
```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3)
```

Creating and Training the Model

In [25]:
```
from sklearn.linear_model import LinearRegression
```

In [26]:
```
lm = LinearRegression()
```

In [27]:
```
lm.fit(X_train,y_train)
```

Out[27]:
```
LinearRegression(copy_X=True, fit_intercept=True, n_jobs=None, normalize=Fals
e)
```

Model Evaluation

In [28]:
```
# print the intercept
print(lm.intercept_)
```

```
0.4626167205767193
```

In [29]:
```
coeff_df = pd.DataFrame(lm.coef_,X.columns,columns=['Coefficient'])
coeff_df
```

Out[29]:

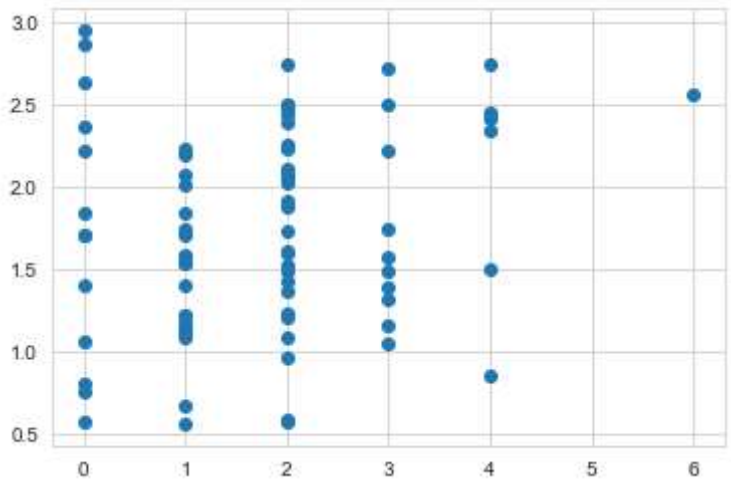
	Coefficient
PrsnIty_type	0.355763
HrsSmrtPhnUse_Wkday	0.150873
Socializing	-0.196556
Entertainment	0.032964
NgtvePhysHlthEffcts_SmrtPhnUse	-0.119057
DcreasdSlpTimeQulty_SmrtPhnUse	-0.108522
TiredDaytime_latenightSmrtPhnUse	-0.255388
Compulsive_Behavior	-0.004183
Functional_Impairment	0.084442
TotAddiction_Score	0.020787

Predictions from our Model

In [30]:
```
predictions = lm.predict(X_test)
```

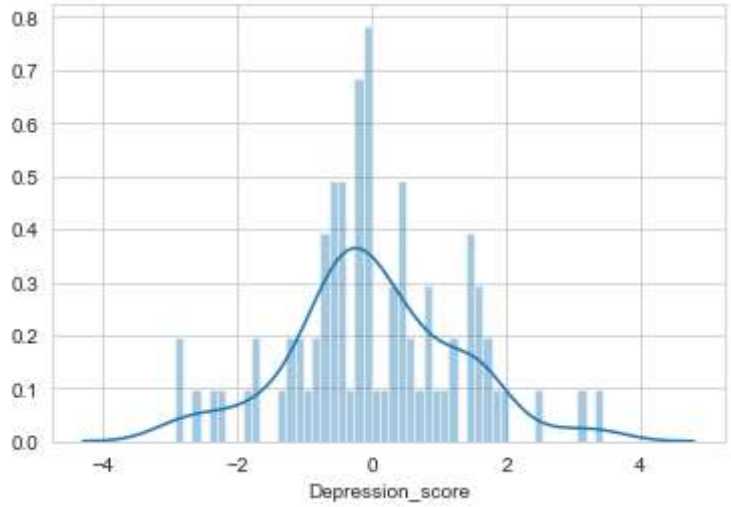
In [31]:
```
plt.scatter(y_test,predictions)
```

Out[31]: <matplotlib.collections.PathCollection at 0x1b2a8975940>



Residual Histogram

In [32]:
```
sns.distplot((y_test-predictions),bins=50);
```



Regression Evaluation Metrics

In [33]:
```
from sklearn import metrics
```

In [34]:
```
print('MAE:', metrics.mean_absolute_error(y_test, predictions))
print('MSE:', metrics.mean_squared_error(y_test, predictions))
print('RMSE:', np.sqrt(metrics.mean_squared_error(y_test, predictions)))
```

```
MAE: 0.949056726936283
MSE: 1.55192857404979
RMSE: 1.2457642529988529
```

In [ ]:

A Suggestion:

--> Incentive-

As an incentive, we could communicate with people that at the end of the study they will be provided with a detailed report from some of the best doctors in this field free of cost. Like their data will be analysed and it will be like a free checkup of their mental health. Since this is a interdisciplinary thing, like if we could get some help from AIIMS also.

In [ ]: