Project No #8

Big Five Personality test

Using Machine Learning Algorithms

Spectral Clustering

	EXT1	EXT2	EXT3	EXT4	EXT5	EXT6	EXT7	EXT8	EXT9	EXT10	...	OPN1	OPN2
0	4.0	1.0	5.0	2.0	5.0	1.0	5.0	2.0	4.0	1.0	...	5.0	1.0
1	3.0	5.0	3.0	4.0	3.0	3.0	2.0	5.0	1.0	5.0	...	1.0	2.0
2	2.0	3.0	4.0	4.0	3.0	2.0	1.0	3.0	2.0	5.0	...	5.0	1.0
3	2.0	2.0	2.0	3.0	4.0	2.0	2.0	4.0	1.0	4.0	...	4.0	2.0
4	3.0	3.0	3.0	3.0	5.0	3.0	3.0	5.0	3.0	4.0	...	5.0	1.0

Data Set:

Big Five Personality Test

1M Answers to 50 personality items, and technical information

kaggle.com

EXT (Extraversion): The degree to which a person is outgoing, sociable, and assertive.
AGR (Agreeableness): Reflects how kind, cooperative, and compassionate a person is.
CSN (Conscientiousness): Measures how organized, responsible, and reliable an individual is.
EST (Emotional Stability): Also referred to as Neuroticism, it gauges emotional resilience and stability.
OPN (Openness to Experience): Captures an individual's openness to new ideas, creativity, and curiosity.

This data was collected (2016-2018) through an interactive on-line personality test.
The personality test was constructed with the "Big-Five Factor Markers" from the IPIP.
https://ipip.ori.org/newBigFiveSbroadKey.htm
Participants were informed that their responses would be recorded and used for research at the beginning of the test, and asked to confirm their consent at the end of the test.

The following items were presented on one page and each was rated on a five point scale using radio buttons. The order on page was was EXT1, AGR1, CSN1, EST1, OPN1, EXT2, etc.
The scale was labeled 1=Disagree, 3=Neutral, 5=Agree

EXT1	I am the life of the party.
EXT2	I don't talk a lot.
EXT3	I feel comfortable around people.
EXT4	I keep in the background.
EXT5	I start conversations.
EXT6	I have little to say.
EXT7	I talk to a lot of different people at parties.
EXT8	I don't like to draw attention to myself.
EXT9	I don't mind being the center of attention.
EXT10	I am quiet around strangers.
EST1	I get stressed out easily.
EST2	I am relaxed most of the time.
EST3	I worry about things.
EST4	I seldom feel blue.
EST5	I am easily disturbed.
EST6	I get upset easily.
EST7	I change my mood a lot.
EST8	I have frequent mood swings.
EST9	I get irritated easily.
EST10	I often feel blue.
AGR1	I feel little concern for others.
AGR2	I am interested in people.
AGR3	I insult people.
AGR4	I sympathize with others' feelings.
AGR5	I am not interested in other people's problems.
AGR6	I have a soft heart.
AGR7	I am not really interested in others.
AGR8	I take time out for others.
AGR9	I feel others' emotions.
AGR10	I make people feel at ease.

CSN1	I am always prepared.
CSN2	I leave my belongings around.
CSN3	I pay attention to details.
CSN4	I make a mess of things.
CSN5	I get chores done right away.
CSN6	I often forget to put things back in their proper place.
CSN7	I like order.
CSN8	I shirk my duties.
CSN9	I follow a schedule.
CSN10	I am exacting in my work.
OPN1	I have a rich vocabulary.
OPN2	I have difficulty understanding abstract ideas.
OPN3	I have a vivid imagination.
OPN4	I am not interested in abstract ideas.
OPN5	I have excellent ideas.
OPN6	I do not have a good imagination.
OPN7	I am quick to understand things.
OPN8	I use difficult words.
OPN9	I spend time reflecting on things.
OPN10	I am full of ideas.

Negative Impact

Positive Impact

The time spent on each question is also recorded in milliseconds. These are the variables ending in _E. This was calculated by taking the time when the button for the question was clicked minus the time of the most recent other button click.

dateload The timestamp when the survey was started.
screenw The width the of user's screen in pixels
screenh The height of the user's screen in pixels
introelapse The time in seconds spent on the landing / intro page
testelapse The time in seconds spent on the page with the survey questions
endelapse The time in seconds spent on the finalization page (where the user was asked to indicate if they has answered accurately and their answers could be stored and used for research. Again: this dataset only includes users who answered "Yes" to this question, users were free to answer no and could still view their results either way)
IPC The number of records from the user's IP address in the dataset. For max cleanliness, only use records where this value is 1. High values can be because of shared networks (e.g. entire universities) or multiple submissions
country The country, determined by technical information (NOT ASKED AS A QUESTION)

lat_appx_lots_of_err approximate latitude of user, determined by technical information, THIS IS NOT VERY ACCURATE. Read the article "How an internet mapping glitch turned a random Kansas farm into a digital hell" https://splinternews.com/how-an-internet-mapping-glitch-turned-a-random-kansas-f-1793856052 to learn about the perils of relying on this information

long_appx_lots_of_err approximate longitude of user

```
1 import pandas as pd
2 import numpy as np
3 from sklearn.cluster import KMeans
4 import matplotlib.pyplot as plt
```

```
: 1 data=pd.read_csv("D:\Lesson\samadhi sir workshops\Machne Learing python\metirials
```

```
: 1 data.head()
```

0	4.0	1.0	5.0	2.0	5.0	1.0	5.0	2.0	4.0	1.0	...	2016-03-03 02:01:01	768.0	1024.0
1	3.0	5.0	3.0	4.0	3.0	3.0	2.0	5.0	1.0	5.0	...	2016-03-03 02:01:20	1360.0	768.0
2	2.0	3.0	4.0	4.0	3.0	2.0	1.0	3.0	2.0	5.0	...	2016-03-03 02:01:56	1366.0	768.0
3	2.0	2.0	2.0	3.0	4.0	2.0	2.0	4.0	1.0	4.0	...	2016-03-03 02:02:02	1920.0	1200.0
4	3.0	3.0	3.0	3.0	5.0	3.0	3.0	5.0	3.0	4.0	...	2016-03-03	1366.0	768.0

```
1 data.shape
```

```
(1015341, 110)
```

```
1 x=data.iloc[:,:50]
```

```
1 x.head()
```

	EXT1	EXT2	EXT3	EXT4	EXT5	EXT6	EXT7	EXT8	EXT9	EXT10	...	OPN1	OPN2	OPN3	C
0	4.0	1.0	5.0	2.0	5.0	1.0	5.0	2.0	4.0	1.0	...	5.0	1.0	4.0	
1	3.0	5.0	3.0	4.0	3.0	3.0	2.0	5.0	1.0	5.0	...	1.0	2.0	4.0	
2	2.0	3.0	4.0	4.0	3.0	2.0	1.0	3.0	2.0	5.0	...	5.0	1.0	2.0	
3	2.0	2.0	2.0	3.0	4.0	2.0	2.0	4.0	1.0	4.0	...	4.0	2.0	5.0	
4	3.0	3.0	3.0	3.0	5.0	3.0	3.0	5.0	3.0	4.0	...	5.0	1.0	5.0	

5 rows × 50 columns

```
1 x=x.fillna(0)
```

```
1 x.head()
```

	EXT1	EXT2	EXT3	EXT4	EXT5	EXT6	EXT7	EXT8	EXT9	EXT10	...	OPN1
0	4.0	1.0	5.0	2.0	5.0	1.0	5.0	2.0	4.0	1.0	...	5.0
1	3.0	5.0	3.0	4.0	3.0	3.0	2.0	5.0	1.0	5.0	...	1.0
2	2.0	3.0	4.0	4.0	3.0	2.0	1.0	3.0	2.0	5.0	...	5.0
3	2.0	2.0	2.0	3.0	4.0	2.0	2.0	4.0	1.0	4.0	...	4.0
4	3.0	3.0	3.0	3.0	5.0	3.0	3.0	5.0	3.0	4.0	...	5.0

Clustering

```
kmc=KMeans(n_clusters=10)
```

```
kmc.fit(x)
```

```
C:\Users\Prasa\anaconda3\new\lib\site-packages\sklearn\cluster\_km
ll change from 10 to 'auto' in 1.4. Set the value of `n_init` expl
  super()._check_params_vs_input(X, default_n_init=10)
```

KMeans

KMeans(n_clusters=10)

Creating personality scores

```
one,two,three,four,five,six,seven,eight,nine,ten=kmc.cluster_centers_
```

```
one
```

```
array([2.04590322, 3.22777318, 2.52878443, 3.85802789, 2.67249085,
       2.55246083, 2.01717057, 3.86441292, 2.49736833, 4.32894146,
       4.0846276 , 2.62372299, 4.48477083, 1.96007628, 3.31407469,
       3.61690654, 3.86747601, 3.59984814, 3.69369952, 3.91696866,
       1.9248119 , 3.91444916, 2.45591737, 4.26427142, 2.0483537 ,
       4.08916615, 2.13644129, 3.75529785, 4.09627597, 3.37237696,
       2.55866466, 3.99804998, 3.84461931, 3.73449472, 1.68933354,
       4.03016498, 3.24229482, 3.17389729, 2.40708911, 3.30517878,
       4.00210534, 1.82971802, 4.47484814, 1.62065127, 3.80664561,
       1.59629323, 3.96395044, 3.6480379 , 4.52511735, 4.25119072])
```

```
type_scores={}

for name,personality_type in types.items():
    personality_score={}

    personality_score["Extroversion_Score"]=personality_type[0]-personality_type[1]+personality_type[2]-personality_type[3]+
    personality_score["Neuroticism_Score"]=personality_type[10]-personality_type[11]+personality_type[12]-personality_type[1
    personality_score["Agreebleness_Score"]=-personality_type[20]+personality_type[21]-personality_type[22]+personality_type
    personality_score["Conscientiousness_Score"]=personality_type[30]-personality_type[31]+personality_type[32]-personality_
    personality_score["Openness_Score"]=personality_type[40]-personality_type[41]+personality_type[42]-personality_type[43]+

    type_scores[name]=personality_score
```

```
type_scores
```


```
{'One': {'Extroversion_Score': -6.069898874852605,
  'Neuroticism_Score': 25.99457272036968,
  'Agreebleness_Score': 14.926313246358292,
  'Conscientiousness_Score': -4.374016359495304,
  'Openness_Score': 16.329157175398436},
 'Two': {'Extroversion_Score': 1.4455329749994572,
  'Neuroticism_Score': 18.729319032017802,
  'Agreebleness_Score': 11.350825476421376,
  'Conscientiousness_Score': 1.512652498859011,
  'Openness_Score': 8.696588351210234},
 'Three': {'Extroversion_Score': 8.39368552917453,
  'Neuroticism_Score': 11.297422145666456,
  'Agreebleness_Score': 14.567764097946394,
  'Conscientiousness_Score': -1.9600990428471667,
  'Openness_Score': 16.60685274716612},
 'Four': {'Extroversion_Score': -6.509029997758278,
  'Neuroticism_Score': 14.298396895237827,
  'Agreebleness_Score': 1.1342998360323677,
  'Conscientiousness_Score': 2.865676571549816,
  'Openness_Score': 15.994939426468214},
 'Five': {'Extroversion_Score': -5.216514303092444,
  'Neuroticism_Score': 24.877023811652418,
  'Agreebleness_Score': 16.64851423153242,
  'Conscientiousness_Score': 7.2843891443187765,
  'Openness_Score': 15.191709751865165},
 'Six': {'Extroversion_Score': -11.484118824140468,
  'Neuroticism_Score': 26.655312703833914,
  'Agreebleness_Score': 7.150509104175662,
  'Conscientiousness_Score': -1.6263916066904827,
  'Openness_Score': 9.393342398815298},
 'Seven': {'Extroversion_Score': 0.42726709996889567,
  'Neuroticism_Score': 1.1830702568862836,
  'Agreebleness_Score': 0.17409470752076772,
  'Conscientiousness_Score': 0.5775301764162872,
  'Openness_Score': 0.8696997833488667},
 'Eight': {'Extroversion_Score': 8.708742081150893,
  'Neuroticism_Score': 25.695137642496608,
  'Agreebleness_Score': 16.868971225791757,
  'Conscientiousness_Score': -0.194494300068071,
  'Openness_Score': 15.921323728271592},
 'Nine': {'Extroversion_Score': 8.795419098317602,
```

"Analyzing Personality Trait Scores and Normalizing Data"

```
]:  1  extro=[]
    2  neuroctic=[]
    3  aggreable=[]
    4  conscient=[]
    5  openn=[]
    6
    7  for personality_type,personality_score in type_scores.items():
    8      extro.append(personality_score["Extroversion_Score"])
    9      neuroctic.append(personality_score["Neuroticism_Score"])
   10      aggreable.append(personality_score["Agreebleness_Score"])
   11      conscient.append(personality_score["Conscientiousness_Score"])
   12      openn.append(personality_score["Openness_Score"])
```

```
]:  1  extro
```

```
]: [-6.069898874852605,
    1.4455329749994572,
    8.39368552917453,
    -6.509029997758278,
    -5.216514303092444,
    -11.484118824140468,
    0.42726709996889567,
    8.708742081150893,
    8.795419098317602,
    -6.391211511939123]
```

```
:  1  fun=lambda L:(np.array(L)-np.array(L).min())/(np.array(L).max()-np.array(L).min())
```

```
:  1  norm_extro=list(fun(extro))
   2  norm_neuroctic=list(fun(neuroctic))
   3  norm_aggreable=list(fun(aggreable))
   4  norm_conscient=list(fun(conscient))
   5  norm_openn=list(fun(openn))
```

```
:  1  norm_extro
```

```
: [0.2669794533775851,
   0.6375713218209624,
   0.9801902010450554,
   0.24532555156854202,
   0.30906051927875156,
   0.0,
   0.5873598288903021,
   0.9957258879616424,
   1.0,
   0.25113527397295043]
```

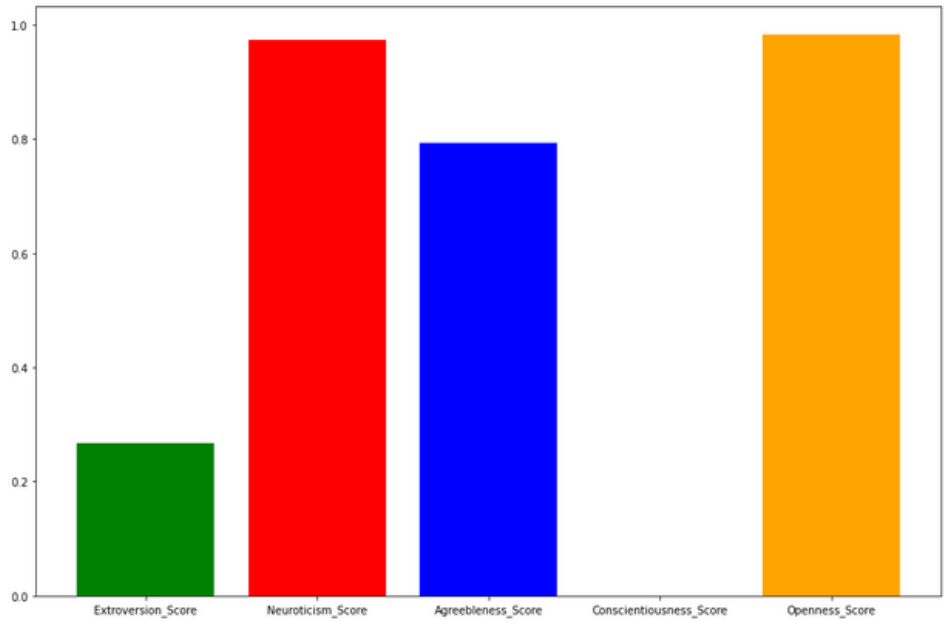
```
idx=0
norm_type_scores={}

for personality_type,personality_score in type_scores.items():
    norm_personality_score={}

    norm_personality_score["Extroversion_Score"]=norm_extro[idx]
    norm_personality_score["Neuroticism_Score"]=norm_neuroctic[idx]
    norm_personality_score["Agreebleness_Score"]=norm_aggreable[idx]
    norm_personality_score["Conscientiousness_Score"]=norm_conscient[idx]
    norm_personality_score["Openness_Score"]=norm_openn[idx]

    norm_type_scores[personality_type]=norm_personality_score
    idx=idx+1
```

```
norm_type_scores
```

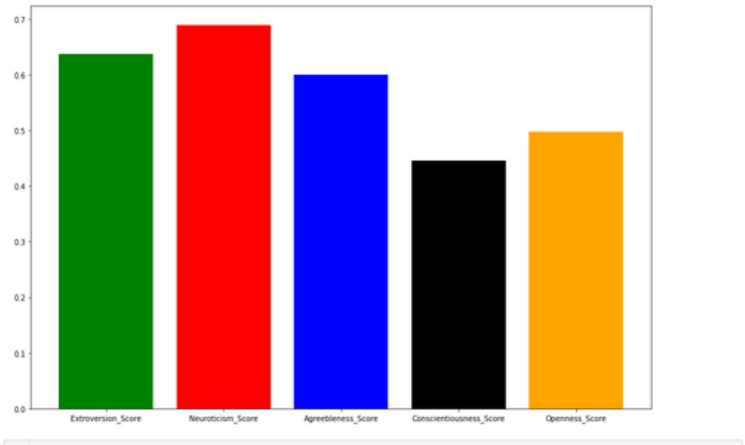

```
{'One': {'Extroversion_Score': 0.2669794533775851,
  'Neuroticism_Score': 0.974060392019258,
  'Agreebleness_Score': 0.7924975624368081,
  'Conscientiousness_Score': 0.0,
  'Openness_Score': 0.9823541416667831},
 'Two': {'Extroversion_Score': 0.6375713218209624,
  'Neuroticism_Score': 0.6888380091260518,
  'Agreebleness_Score': 0.6004203277661163,
  'Conscientiousness_Score': 0.4452447530088492,
  'Openness_Score': 0.49735098755517554},
 'Three': {'Extroversion_Score': 0.9801902010450554,
  'Neuroticism_Score': 0.39707347752542244,
  'Agreebleness_Score': 0.7732360984487202,
  'Conscientiousness_Score': 0.18257932377314867,
  'Openness_Score': 1.0},
 'Four': {'Extroversion_Score': 0.24532555156854202,
  'Neuroticism_Score': 0.5148870055578153,
  'Agreebleness_Score': 0.051582765113017966,
  'Conscientiousness_Score': 0.5475822351325949,
  'Openness_Score': 0.9611166440267301},
 'Five': {'Extroversion_Score': 0.30906051927875156,
  'Neuroticism_Score': 0.9301871872535277,
  'Agreebleness_Score': 0.8850151779523057,
  'Conscientiousness_Score': 0.8817964801359436,
  'Openness_Score': 0.9100763016948085},
 'Six': {'Extroversion_Score': 0.0,
  'Neuroticism_Score': 1.0,
  'Agreebleness_Score': 0.37477694553823554,
  'Conscientiousness_Score': 0.20781965723915183,
  'Openness_Score': 0.5416254537948464},
 'Seven': {'Extroversion_Score': 0.5873598288903021,
  'Neuroticism_Score': 0.0,
  'Agreebleness_Score': 0.0,
  'Conscientiousness_Score': 0.37451573503492674,
  'Openness_Score': 0.0},
 'Eight': {'Extroversion_Score': 0.9957258879616424,
  'Neuroticism_Score': 0.9623050438791513,
```

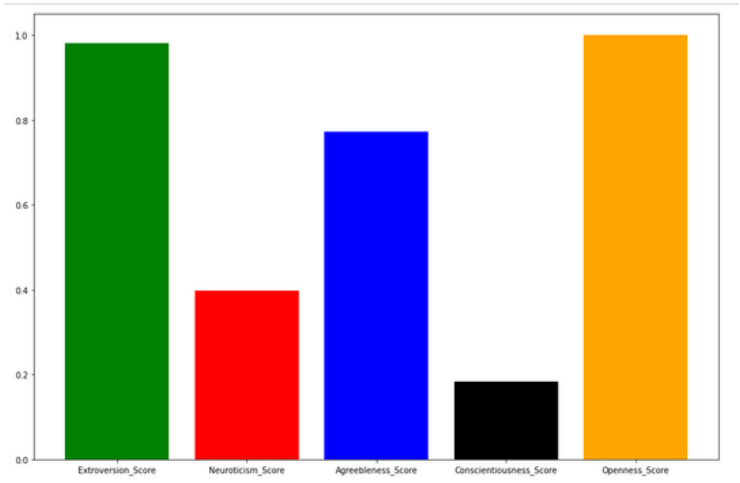
Visualizing

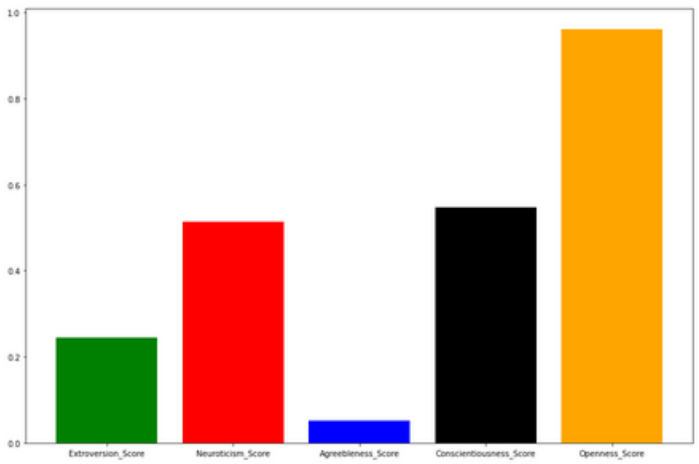
```
plt.figure(figsize=(15,10))
plt.bar(list(norm_type_scores["One"].keys()),norm_type_scores["One"].values(),color=["green","red","blue","black","orange"])
plt.show()
```

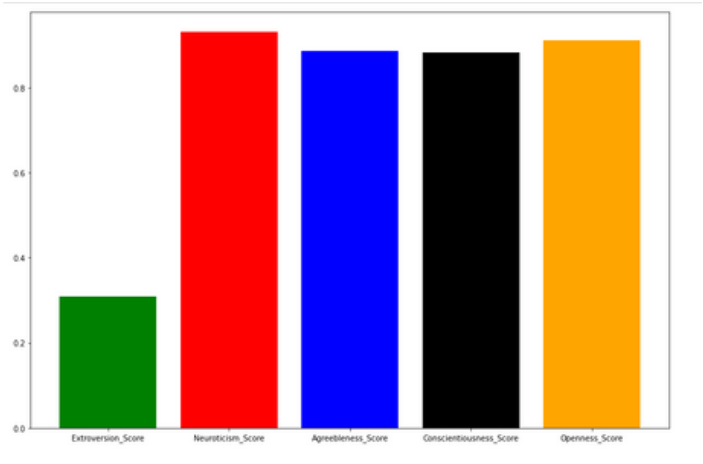


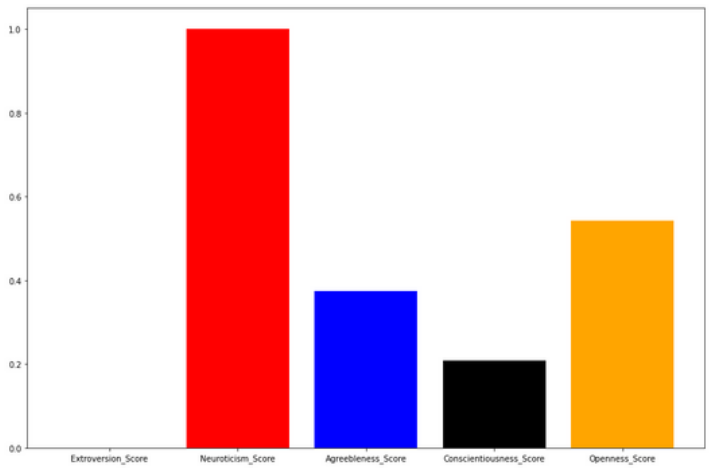

```
plt.figure(figsize=(15,10))
plt.bar(list(norm_type_scores["Two"].keys()),norm_type_scores["Two"].values(),color=["green","red","blue","black","orange"])
plt.show()
```

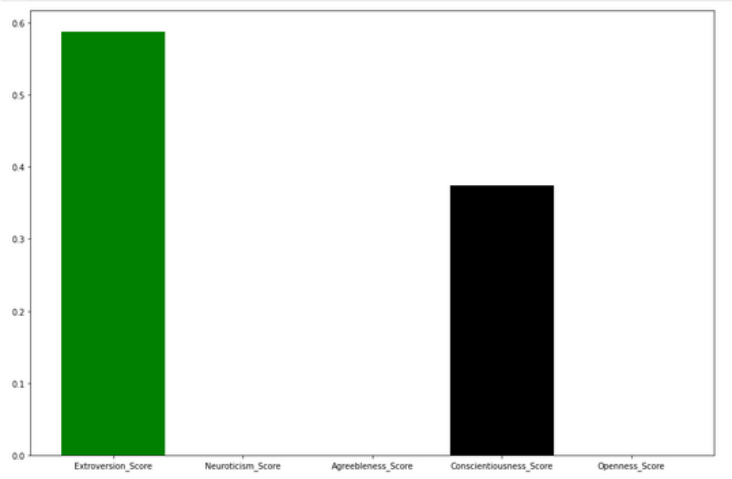


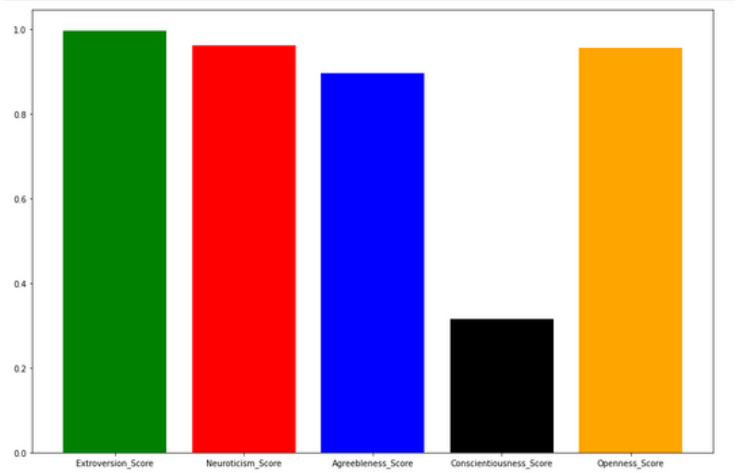


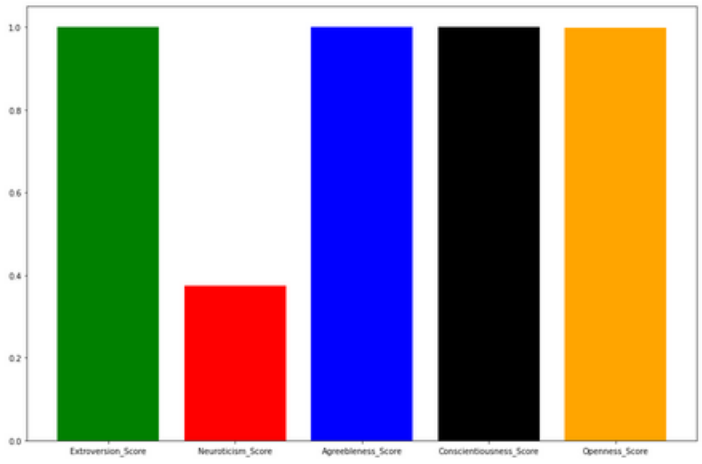













```
:  1 plt.figure(figsize=(15,10))
   2 plt.bar(list(norm_type_scores["Ten"].keys()),norm_type_scores["Ten"].values(),color=["green","red","blue","black","orange"])
   3 plt.show()
```