

CS156 (Introduction to AI), Spring 2021

Assignment 3

Roster Name: GURSIMRAN SINGH

Preferred Name (if different): SIMRAN

Student ID: 015212210

Email address: gursimransingh@sjsu.edu

References and sources

<https://towardsdatascience.com/how-to-read-csv-file-using-pandas-ab1f5e7e7b58>

https://pandas.pydata.org/docs/getting_started/intro_tutorials/03_subset_data.html

List all your references and sources here. This includes all sites/discussion boards/blogs/posts/etc. where you grabbed some code examples.

Solution

Load libraries and set random number generator seed

Code the solution

```
In [34]: import pandas as pd
        from sklearn.model_selection import train_test_split
        import matplotlib.pyplot as plt
        import numpy as np
```

```
In [35]: df = pd.read_csv (r'C:\Users\gursi\Downloads\homework3_input_data.csv', usecols=['Economy', 'Happiness', 'Family', 'Health (Life Expectancy)', 'Freedom', 'Trust (Government Corruption)', 'Generosity', 'Dystopia Residual'])
        df.head()
```

```
Out[35]:
```

	Happiness Score	Economy (GDP per Capita)	Family	Health (Life Expectancy)	Freedom	Trust (Government Corruption)	Generosity	Dystopia Residual
0	7.587	1.39651	1.34951	0.94143	0.66557	0.41978	0.29678	2.51738
1	7.561	1.30232	1.40223	0.94784	0.62877	0.14145	0.43630	2.70201
2	7.527	1.32548	1.36058	0.87464	0.64938	0.48357	0.34139	2.49204
3	7.522	1.45900	1.33095	0.88521	0.66973	0.36503	0.34699	2.46531
4	7.427	1.32629	1.32261	0.90563	0.63297	0.32957	0.45811	2.45176

```
In [36]: df.describe()
```

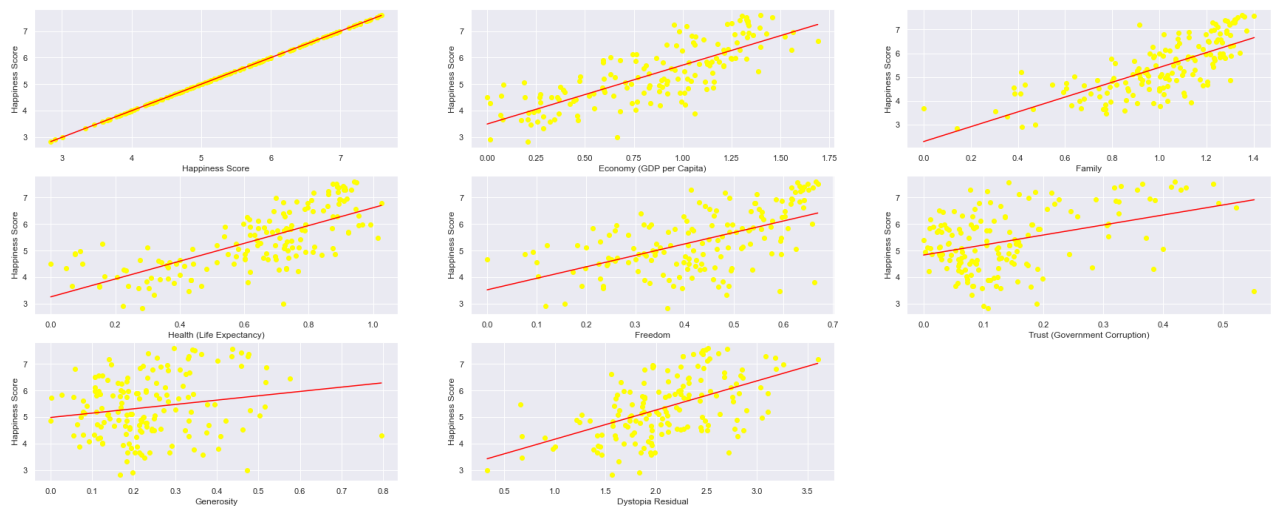
Out[36]:

	Happiness Score	Economy (GDP per Capita)	Family	Health (Life Expectancy)	Freedom	Trust (Government Corruption)	Generosity	Dystop Residu
count	158.000000	158.000000	158.000000	158.000000	158.000000	158.000000	158.000000	158.000000
mean	5.375734	0.846137	0.991046	0.630259	0.428615	0.143422	0.237296	2.09897
std	1.145010	0.403121	0.272369	0.247078	0.150693	0.120034	0.126685	0.55355
min	2.839000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.32858
25%	4.526000	0.545808	0.856823	0.439185	0.328330	0.061675	0.150553	1.75941
50%	5.232500	0.910245	1.029510	0.696705	0.435515	0.107220	0.216130	2.09541
75%	6.243750	1.158448	1.214405	0.811013	0.549092	0.180255	0.309883	2.46241
max	7.587000	1.690420	1.402230	1.025250	0.669730	0.551910	0.795880	3.60214

In [37]:

```
plt.figure(figsize=(30,20))
for i, columns in enumerate(df.columns):
    plt.subplot(5, 3, i+1)
    x = df[columns]
    y = df['Happiness Score']
    plt.scatter(x, y, color='yellow')

    #plt.plot(x, y, '.', color="forestgreen",)
    plt.xlabel(columns)
    plt.ylabel('Happiness Score')
    plt.plot(np.unique(x), np.poly1d(np.polyfit(x, y, 1))(np.unique(x)),color="red") #h
```

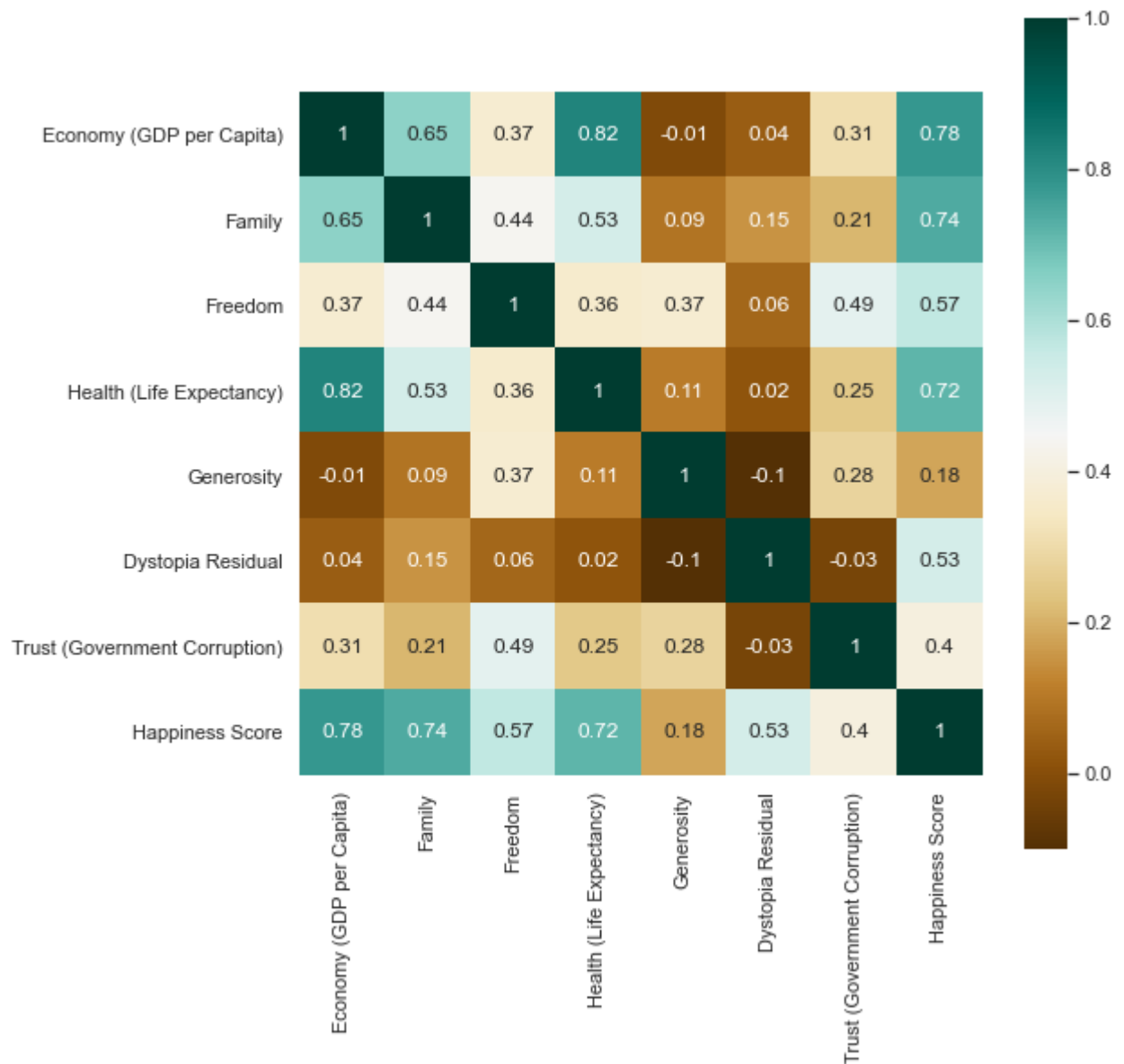


In [38]:

```
import seaborn as sns

features = df[['Economy (GDP per Capita)', 'Family', 'Freedom', 'Health (Life Expectancy)']
sns.set(rc={'figure.figsize': (8.5,8.5)})
sns.heatmap(features.corr().round(2), square=True, cmap='BrBG', annot=True)
```

Out[38]: <AxesSubplot:>



```
In [39]: X_train,X_test,Y_train,Y_test = train_test_split(df[['Economy (GDP per Capita)','Family
```

```
In [40]: X_train.shape , X_test.shape
```

```
Out[40]: ((126, 7), (32, 7))
```

```
In [41]: from sklearn import linear_model
          model= linear_model.LinearRegression().fit(X_train, Y_train)
```

```
In [42]: pred_values=model.predict(X_test)
```

```
In [43]: from sklearn.metrics import mean_squared_error, mean_absolute_error, r2_score
```

```
In [44]: mean_squared_error(pred_values,Y_test)
```

```
Out[44]: 9.863758664525458e-08
```

```
In [45]: r2_score(pred_values,Y_test)
```

Out[45]: 0.9999999035244795

In []:

In []:

In []:

In []:

In []: