Data Collection and Preprocessing Phase

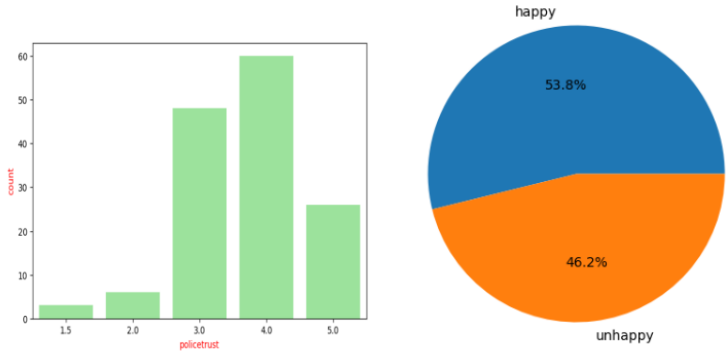
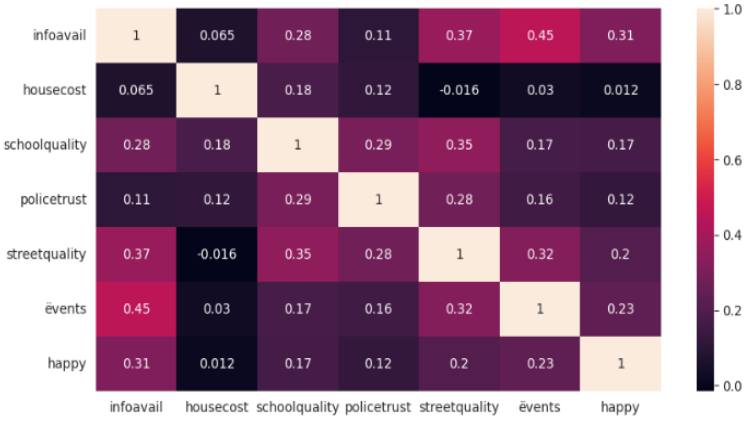
Date	21 June 2024
Team ID	739812
Project Title	Eudaimonia Engine: Machine Learning Delving into Happiness Classification
Maximum Marks	6 Marks

Data Exploration and Preprocessing Report

Dataset variables will be statistically analyzed to identify patterns and outliers, with Python employed for preprocessing tasks like normalization and feature engineering. Data cleaning will address missing values and outliers, ensuring quality for subsequent analysis and modeling, and forming a strong foundation for insights and predictions.

Section	Description
Data Overview	Dimension: 143 rows × 7 columns Descriptive statistics:
Univariate Analysis	

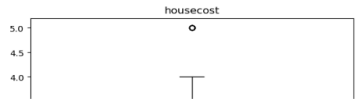
	infoavail	housecost	schoolquality	policetrust	streetquality	ëvents	happy
count	143.000000	143.000000	143.000000	143.000000	143.000000	143.000000	143.000000
mean	4.325175	2.513986	3.265734	3.699301	3.615385	4.216783	0.538462
std	0.765126	1.068011	0.992586	0.888383	1.131639	0.848693	0.500271
min	2.500000	1.000000	1.000000	1.000000	1.000000	1.000000	0.000000
25%	4.000000	2.000000	3.000000	3.000000	3.000000	4.000000	0.000000
50%	5.000000	3.000000	3.000000	4.000000	4.000000	4.000000	1.000000
75%	5.000000	3.000000	4.000000	4.000000	4.000000	5.000000	1.000000
max	5.000000	4.500000	5.000000	5.000000	5.000000	5.000000	1.000000

<table>
<tr><td></td><td> 
</td></tr>
<tr><td>Bivariate Analysis</td><td> 
</td></tr>
<tr><td>Multivariate Analysis</td><td> 
</td></tr>
</table>

Outliers and Anomalies

```
#HANDLING OUTLIERS

plt.figure(figsize=[1,1])
for i in df:
    plt.boxplot(df[i])
    plt.title(i)
    plt.show()
```




```
IQt = []
IQt.append(df["infoavail"].quantile(0.75) - df["infoavail"].quantile(0.25))
IQt
```

```
[1.0]
```

```
upper = []
IQt.append(df["infoavail"].quantile(0.75) + 1.5 * IQt[0])
IQt
```

```
[1.0, 6.5]
```

```
lower=[]
IQt.append(df["infoavail"].quantile(0.25) - 1.5 * IQt[0])
IQt
```

```
[1.0, 6.5, 2.5]
```

```
import numpy as np

df["infoavail"] = np.where(df["infoavail"] == 6.5, 6.5, np.where(df["infoavail"] == 2.5, 2.5, df["infoavail"]))
```

```
IQt = []
IQt.append(df["housecost"].quantile(0.75) - df["housecost"].quantile(0.25))
IQt
```

```
[1.0]
```

```
upper = []
IQt.append(df["housecost"].quantile(0.75) + 1.5 * IQt[0])
IQt
```

```
[1.0, 4.5]
```

```
lower=[]
IQt.append(df["housecost"].quantile(0.25) - 1.5 * IQt[0])
IQt
```

```
[1.0, 4.5, 0.5]
```

```
plt.figure(figsize=(5,5))
for i in df:
    plt.boxplot(df[i])
    plt.title(i)
    plt.show()
```

Data Preprocessing Code Screenshots

Loading Data

```
[ ]  #READ THE DATASET

     df=pd.read_csv("/content/happydata.csv")
     df.head( )
```

	infoavail	housecost	schoolquality	policetrust	streetquality	ëvents	happy
0	3	3	3	4	2	4	0
1	3	2	3	5	4	3	0
2	5	3	3	3	3	5	1
3	5	4	3	3	3	5	0
4	5	4	3	3	3	5	0

<table>
<tr>
<td>Handling Missing Data</td>
<td>

```
#DATA PREPARATION
#HANDLING MISSING VALUES

df.isnull().any()
```

```
infoavail        False
housecost        False
schoolquality    False
policetrust      False
streetquality    False
ëvents           False
happy            False
dtype: bool
```

```
df.isnull().sum()
```

```
infoavail        0
housecost        0
schoolquality    0
policetrust      0
streetquality    0
ëvents           0
happy            0
dtype: int64
```

```
#HANDLING DUPLICATES VALUES

df.duplicated().sum()
```

```
18
```

```
df.drop_duplicates()
```

	infoavail	housecost	schoolquality	policetrust	streetquality	ëvents	happy
0	3	3	3	4	2	4	0
1	3	2	3	5	4	3	0
2	5	3	3	3	3	5	1
3	5	4	3	3	3	5	0
5	5	5	3	5	5	5	1
...	...	...	...	...	...	...	...
137	5	2	3	4	4	3	1
138	5	3	3	1	3	5	0
139	5	2	3	4	2	5	1
141	4	3	3	4	4	5	0
142	5	3	2	5	5	5	0

125 rows × 7 columns

</td>
</tr>
<tr>
<td>Data Transformation</td>
<td>

```
# Separate the independent variables
x = df.drop(columns='happy',axis=1)

# Separate the target variable
y = df['happy']

from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(x,y, test_size=0.2, random_state=0)
```

</td>
</tr>
<tr>
<td>Feature Engineering</td>
<td>Attached are the codes in the final submission.</td>
</tr>
<tr>
<td>Save Processed Data</td>
<td>-</td>
</tr>
</table>