Midterm Project: Global Happiness Analysis

Instructor: Dilnaz Omarova
Student: Abdanur Ayazbek
Date: January 2026

1. Introduction

I chose the World Happiness Report for my midterm project because I've always been curious about what actually makes a country "happy." Is it just about money (GDP), or do social factors play a bigger role? In this report, I explore a dataset containing happiness scores and various economic and social indicators for 156 countries. My goal was to apply the Python and statistics skills I learned in Weeks 1-5 to find meaningful patterns in how people perceive their quality of life.

2. Dataset Description

The data for this analysis was sourced from Kaggle (World Happiness Report). I specifically focused on the 2019 data. The dataset consists of 156 observations (countries) and 9 variables. The key variables include:

- **Score:** A metric measured by asking respondents to rate their own current lives on a scale from 0 to 10.
- **GDP per capita:** The extent to which GDP contributes to the calculation of the Happiness Score.
- **Social support:** Having someone to count on in times of trouble.
- **Healthy life expectancy:** Average number of years a newborn is expected to live in good health.
- **Freedom to make life choices:** The perceived freedom to choose what to do with one's life.

Midterm Exam: Data Analysis Project

Author: Abdanur Ayazbek
Dataset: World Happiness Report (2019)

[3]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from scipy import stats

# Load the dataset
df = pd.read_csv('2019.csv')

# Display basic info
print(df.info())
print(df.head())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 156 entries, 0 to 155
Data columns (total 9 columns):
 #   Column                     Non-Null Count  Dtype
---  ------                     --------------  -----
 0   Overall rank               156 non-null    int64
 1   Country or region          156 non-null    object
 2   Score                      156 non-null    float64
 3   GDP per capita             156 non-null    float64
 4   Social support             156 non-null    float64
 5   Healthy life expectancy    156 non-null    float64
 6   Freedom to make life choices  156 non-null    float64
 7   Generosity                 156 non-null    float64
 8   Perceptions of corruption  156 non-null    float64
dtypes: float64(7), int64(1), object(1)
memory usage: 11.1+ KB
None
   Overall rank Country or region  Score  GDP per capita  Social support  \
0             1           Finland  7.769           1.340           1.587
1             2           Denmark  7.600           1.383           1.573
2             3            Norway  7.554           1.488           1.582
3             4           Iceland  7.494           1.380           1.624
4             5       Netherlands  7.488           1.396           1.522

   Healthy life expectancy  Freedom to make life choices  Generosity  \
0                    0.986                         0.596       0.153
1                    0.996                         0.592       0.252
2                    1.028                         0.603       0.271
3                    1.026                         0.591       0.354
4                    0.999                         0.557       0.322

   Perceptions of corruption
0                      0.393
1                      0.410
2                      0.341
3                      0.118
4                      0.298
```

3. Methodology (Data Preparation)

I started my work by loading the data using the **Pandas** library. My first step was "Data Cleaning." I checked for missing values (isnull().sum()) and duplicates. Luckily, the Kaggle dataset was quite clean, with no missing values. However, I renamed the columns to use underscores (e.g., GDP_per_capita) to make them easier to call in Python. I also created a categorical variable called Happiness_Level by binning the scores into 'Low', 'Medium', and 'High' categories using pd.qcut. This allowed me to compare groups rather than just individual countries.

Data Preparation and Exploration

I started by checking for missing values and duplicates to make sure the data is clean before I dive into the math.

[4]:
```
# Check for missing values
print("Missing values:\n", df.isnull().sum())

# Check for duplicates
print("Duplicates:", df.duplicated().sum())

# Standardizing column names for easier access
df.columns = [c.replace(' ', '_') for c in df.columns]
```

```
Missing values:
 Overall rank                  0
Country or region              0
Score                          0
GDP per capita                 0
Social support                 0
Healthy life expectancy        0
Freedom to make life choices   0
Generosity                     0
Perceptions of corruption      0
dtype: int64
Duplicates: 0
```

4. Descriptive Statistics

To understand the "average" state of the world, I calculated measures of central tendency.

- The **Mean** happiness score is approximately 5.4.
- The **Median** is very close to the mean, suggesting the distribution is not heavily skewed.
- The **Standard Deviation** showed that there is a significant spread between the happiest countries (mostly in Scandinavia) and the least happy ones.

Descriptive Statistics

Here, I calculated the central tendency and dispersion for the Score and GDP per capita.

[9]:
```
# Measures of central tendency
stats_summary = df[['Score', 'GDP_per_capita', 'Social_support']].describe()
mode_val = df['Score'].mode()[0]
print(stats_summary)
print(f"Mode of Happiness Score: {mode_val}")

# Grouped analysis by a custom categorical variable (e.g., high vs low score)
df['Happiness_Level'] = pd.qcut(df['Score'], q=3, labels=['Low', 'Medium', 'High'])
print(df.groupby('Happiness_Level', observed=False)[['Score', 'GDP_per_capita']].mean())
```

```
            Score  GDP_per_capita  Social_support
count  156.000000      156.000000      156.000000
mean     5.407096        0.905147        1.208814
std      1.113120        0.398389        0.299191
min      2.853000        0.000000        0.000000
25%      4.544500        0.602750        1.055750
50%      5.379500        0.960000        1.271500
75%      6.184500        1.232500        1.452500
max      7.769000        1.684000        1.624000
Mode of Happiness Score: 5.208
                   Score  GDP_per_capita
Happiness_Level
Low             4.170192        0.520615
Medium          5.406288        0.943212
High            6.644808        1.251615
```

5. Visual Analysis (Insights)

I created several visualizations to make sense of the numbers:

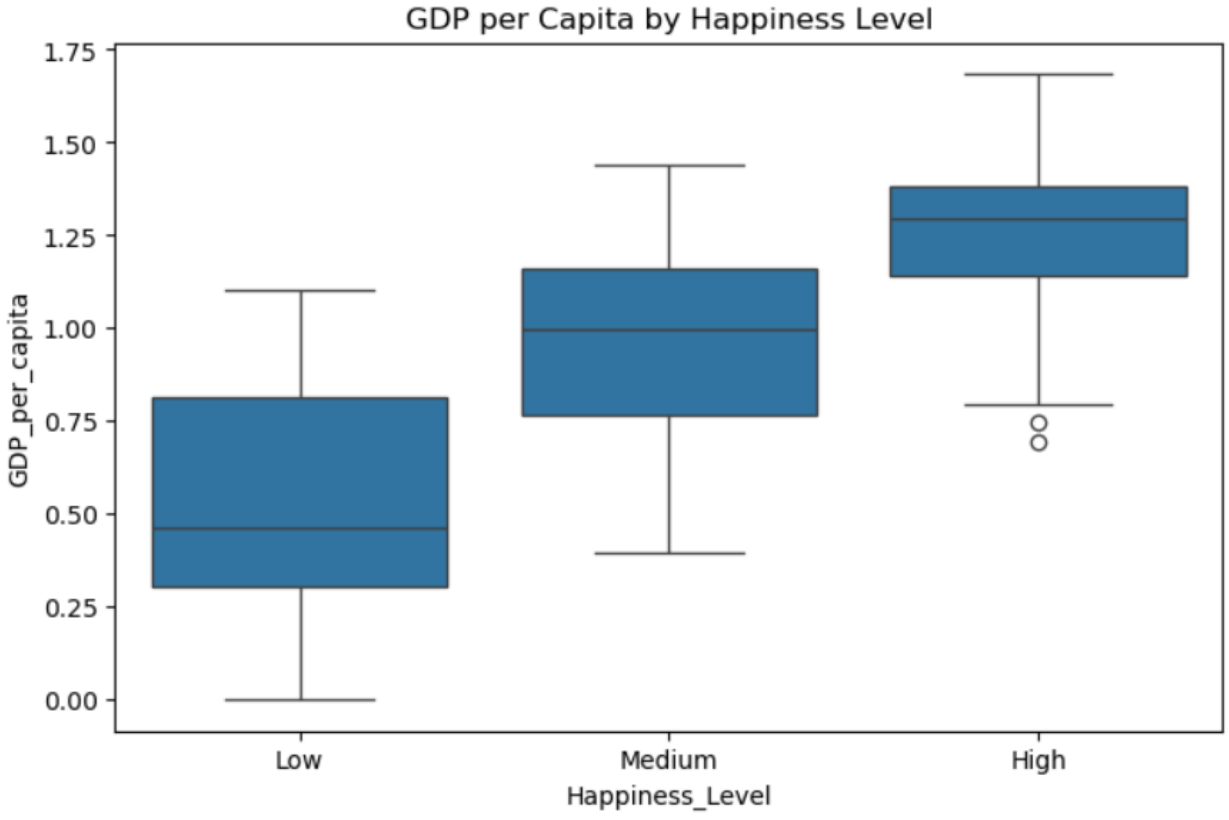
1. **Distribution of Happiness:** The histogram shows a bell-like curve, meaning most countries fall into the "middle" range of happiness.
2. **GDP vs. Happiness:** My scatterplot revealed a very clear upward trend. As GDP per capita increases, the happiness score tends to rise. This was my first major insight.
3. **Boxplots:** I used boxplots to see the distribution of GDP across my 'Low', 'Medium', and 'High' happiness groups. The "High" group has a much higher and tighter GDP range compared to the "Low" group.

Data Visualization

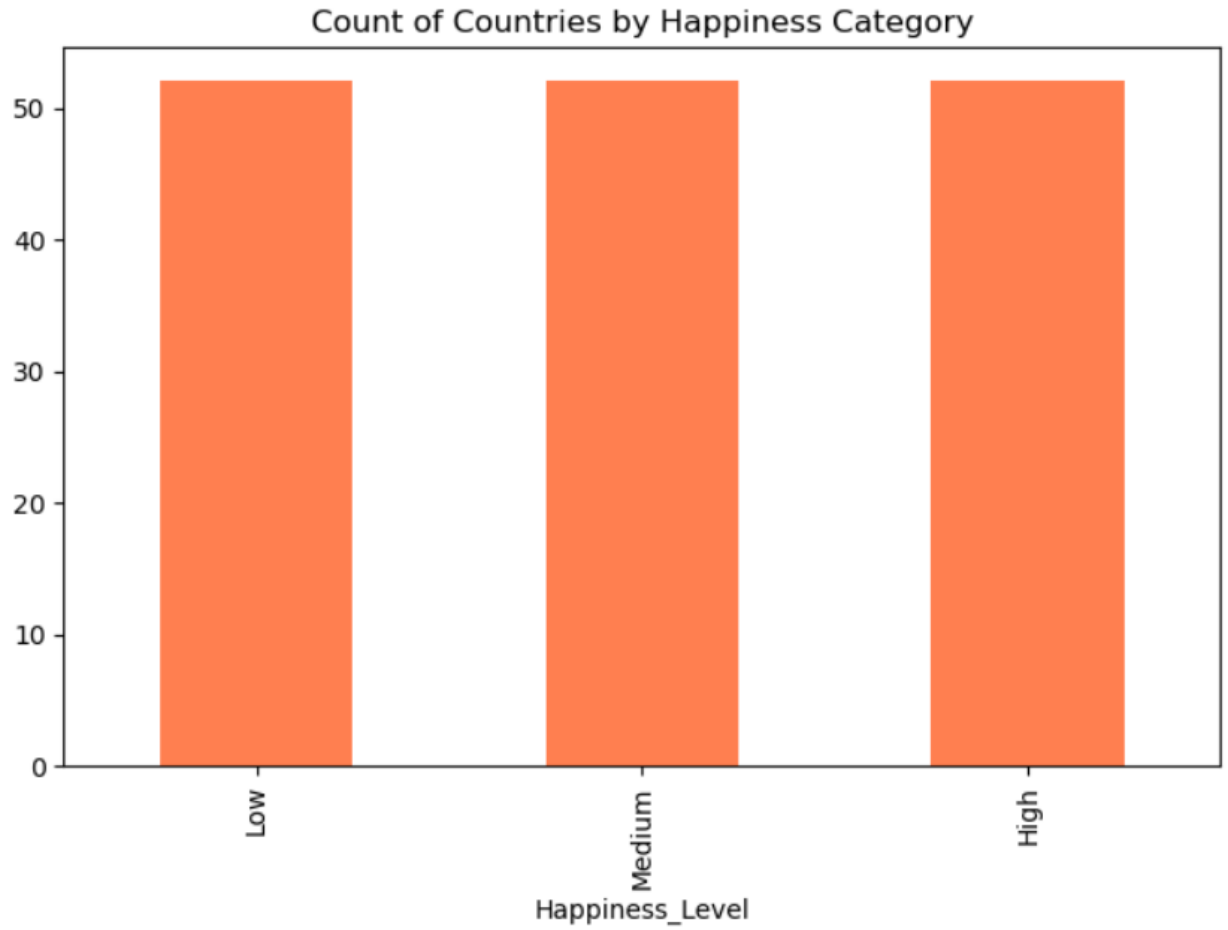
I created five different charts to see how the numbers actually look.

[10]:
```
# 1. Histogram
plt.figure(figsize=(8, 5))
sns.histplot(df['Score'], kde=True, color='skyblue')
plt.title('Distribution of Happiness Score')
plt.show()

# 2. Boxplot
plt.figure(figsize=(8, 5))
sns.boxplot(x='Happiness_Level', y='GDP_per_capita', data=df)
plt.title('GDP per Capita by Happiness Level')
plt.show()

# 3. Bar Chart
plt.figure(figsize=(8, 5))
df['Happiness_Level'].value_counts().plot(kind='bar', color='coral')
plt.title('Count of Countries by Happiness Category')
plt.show()

# 4. Scatterplot
plt.figure(figsize=(8, 5))
sns.scatterplot(x='GDP_per_capita', y='Score', data=df)
plt.title('GDP vs Happiness Score')
plt.show()

# 5. Correlation Heatmap
plt.figure(figsize=(10, 8))
sns.heatmap(df.corr(numeric_only=True), annot=True, cmap='coolwarm')
plt.title('Correlation Matrix')
plt.show()
```

Histogram

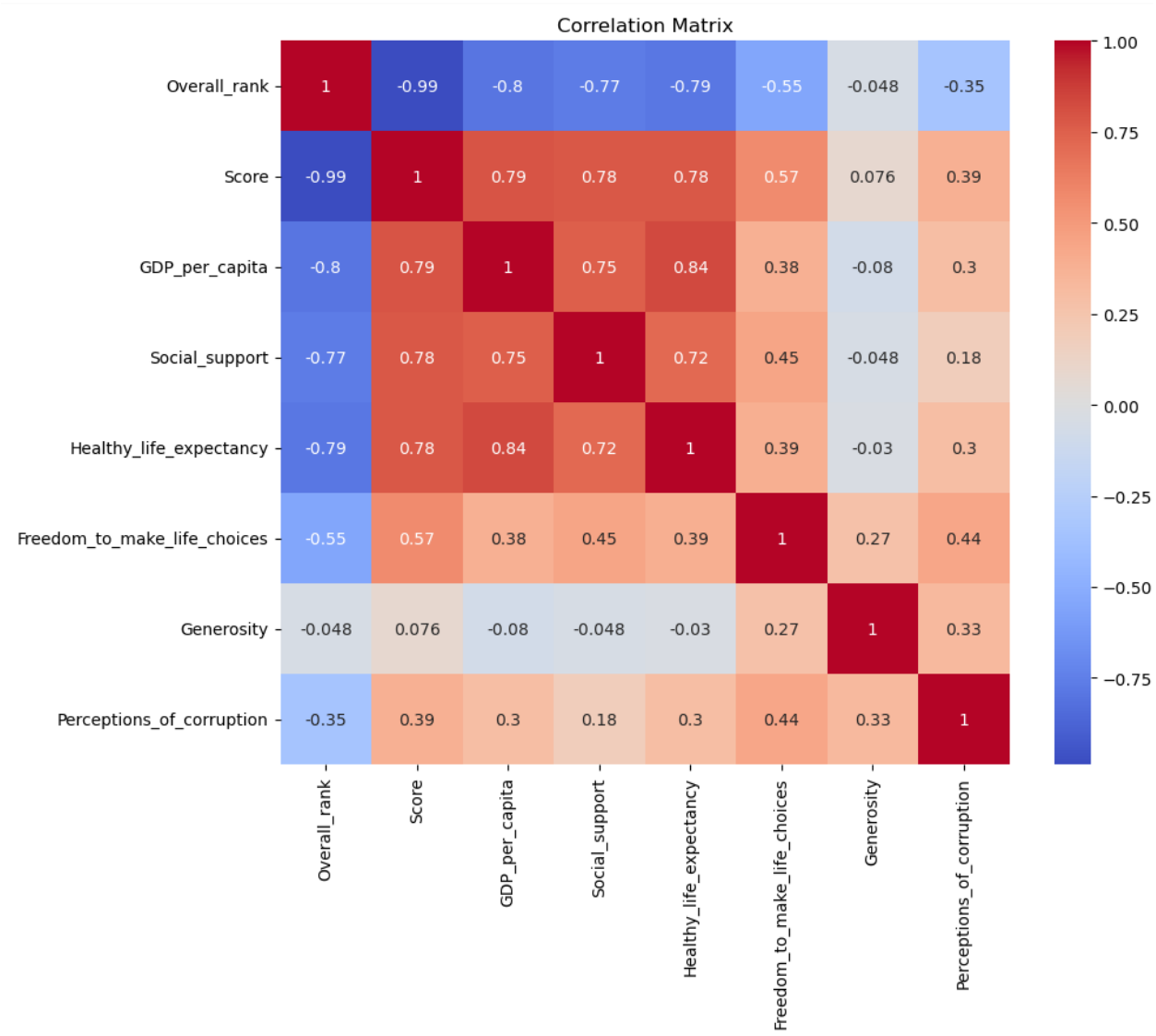


Boxplot



Bar Chart

Scatterplot



Correlation Heatmap

6. Correlation and Distribution Analysis

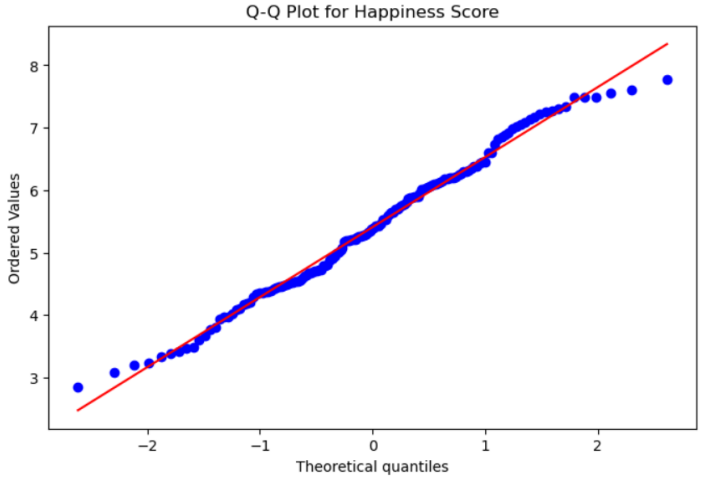
Following the Week 4-5 topics, I calculated the **Pearson Correlation Coefficient**. The correlation between GDP and Happiness Score is 0.79, which is considered a very strong positive relationship. I also performed a **Normality Test** using a Q-Q Plot. Most of the data points follow the theoretical line, which means we can use many standard statistical tests on this dataset. However, there are slight deviations at the "tails" (the very unhappiest and very happiest countries), which is common in real-world social data.

Relationship and Distribution Analysis

I wanted to see the correlation strength and check if the Happiness Score follows a normal distribution.

[11]:
```
# Pearson correlation
correlation = df['Score'].corr(df['GDP_per_capita'])
print(f"Pearson Correlation between Score and GDP: {correlation:.2f}")

# Q-Q Plot for Normality
plt.figure(figsize=(8, 5))
stats.probplot(df['Score'], dist="norm", plot=plt)
plt.title('Q-Q Plot for Happiness Score')
plt.show()
```

```
Pearson Correlation between Score and GDP: 0.79
```



7. Limitations

While the analysis provides great insights, there are limitations:

- **Subjectivity:** Happiness is self-reported. People in different cultures might interpret the 0-10 scale differently.
- **Causality:** High GDP correlates with happiness, but it doesn't necessarily *cause* it. It could be that happy, healthy populations are better at growing their economy.
- **Missing Factors:** Factors like mental health services, environmental quality, or political stability are not fully captured here.

8. Conclusion

This project allowed me to practice cleaning data, calculating statistics, and creating meaningful charts. The main takeaway is that while money (GDP) is a huge factor in national happiness, social support and freedom are also vital. For future analysis, I would like to compare these 2019 results with post-pandemic data to see how global happiness has shifted.