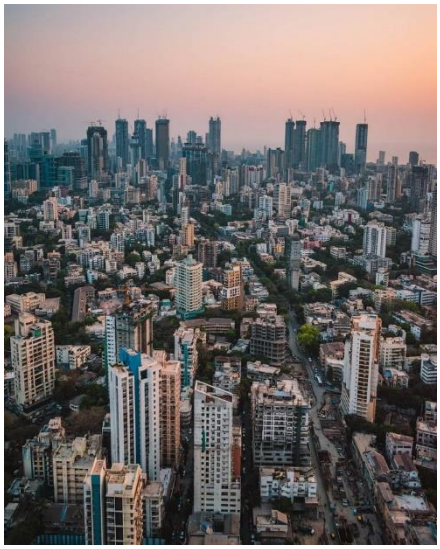





Global Happiness Index

Evaluating the Well-Being and Satisfaction Levels Across the World's Major Countries






A study conducted by Tanish Afre, Dundalk Institute of Technology

Index

World Happiness Report

The World Happiness Report measures global happiness based on factors like GDP per capita, social support, life expectancy, freedom to make life choices, generosity, and perceptions of corruption. It ranks countries, accordingly, providing insights into well-being and highlighting areas for improvement in policies and societal well-being.

Objective

The primary aim of developing a composite index from the World Happiness Report is to synthesize several measures of national well-being into one comprehensive indicator. This would allow clearer, at-a-glance comparisons between countries and give an insight into which of the other measures contribute most strongly towards overall happiness.

Reason for Choosing this Dataset

Global Relevance: The World Happiness Report serves as a proper review which reflects the state of global happiness, shedding light on how different nations prioritize the well-being of their citizens.

Multidimensional Approach: This report encompasses a variety of indicators, including economic wealth (GDP per capita), social support, life expectancy, freedom to make life choices, generosity, and perceptions of corruption, offering a holistic view of happiness and well-being.

Public Interest and Awareness: it highlights public awareness regarding the factors that contribute to happiness, promoting a better understanding of how societal values and policies align with individual well-being.

Link To Database: https://www.kaggle.com/datasets/ajaypalsinghlo/world-happiness-report-2023?resource=download

Identify the Problem and Justification

It remains one of the most central goals belonging to being human—the pursuit of happiness—and what its constituents are remains a hot issue for all people, particularly for policymakers. Over time, the conventional metrics used include the Gross Domestic Product (GDP) in measuring the success level of countries, and it, however, fails to capture quality of life and well-being among citizens. The World Happiness Report is a valuable database in its assessment, indicating progress in a much more holistic view, since it covers contributing factors to happiness and well-being.

The justification for choosing the World Happiness Report is twofold: One, the report is anchored on a significant empirical study for measuring the happiness of populations around the world. It may be economic wealth, social support, freedom, trust in governance, or longevity. These are the ingredients, which, when mixed, are somehow trying to describe a more vivid meaning of what makes life worth living.

The work contributes to an expanded discussion of human development that adds emotional health and satisfaction to the list of "goods" societies should be aiming for. As such, this year's edition provides a more solid and nuanced data set for the World Happiness Report, building an index that should reflect not just the multi-dimensionality of happiness but also possibly help guide policy for better lives for citizens around the world.

Relevance of the World Happiness Report Data

The World Happiness Report is a suitable database to be used in the construction of a composite happiness index, considering that it gives a strong, multifaceted picture of well-being all around the world. The report's data are collected consistently across countries, enabling comparative analysis. It does reflect the outcomes of national policies and the individual circumstances on perceived life satisfaction and, therefore, is highly useful for representing a tool in the study of the complexities in happiness and its drivers, central to human development and policy making.

Addressing Multidimensionality

This shows the complexity of happiness in its multidimensional form and the fact that some of its subjective variables, such as emotional well-being and cultural idiosyncrasies, may remain unelaborated. Still, this must be underlined with all due importance to unquantifiable variables that might play a role in influencing happiness. In this regard, future improvements should invite personalized metrics that mirror individual experiences and a wide diversity across different cultures in order to understand global happiness at much deeper levels.

Imputation of Missing Data

Step 1: I started by assessing the dataset to gain an insight about all the columns present.
This helped me understand the scope of data cleaning needed (such as handling missing values in 'Healthy life expectancy' and 'Dystopia + residual' columns if exhisted)

```
# Inspect the data
print(df.info())
print(df.head())
```

```
✓ 0.0s

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 137 entries, 0 to 136
Data columns (total 19 columns):
 #   Column                                Non-Null Count  Dtype
---  ------                                --------------  -----
 0   Country name                          137 non-null    object
 1   Ladder score                          137 non-null    float64
 2   Standard error of ladder score        137 non-null    float64
 3   upperwhisker                          137 non-null    float64
 4   lowerwhisker                          137 non-null    float64
 5   Logged GDP per capita                 137 non-null    float64
 6   Social support                        137 non-null    float64
 7   Healthy life expectancy               136 non-null    float64
 8   Freedom to make life choices          137 non-null    float64
 9   Generosity                            137 non-null    float64
 10  Perceptions of corruption             137 non-null    float64
 11  Ladder score in Dystopia              137 non-null    float64
 12  Explained by: Log GDP per capita      137 non-null    float64
 13  Explained by: Social support          137 non-null    float64
 14  Explained by: Healthy life expectancy 136 non-null    float64
 15  Explained by: Freedom to make life choices 137 non-null  float64
 16  Explained by: Generosity              137 non-null    float64
 17  Explained by: Perceptions of corruption 137 non-null  float64
 18  Dystopia + residual                   136 non-null    float64
dtypes: float64(18), object(1)
```

Step 1

Step 2: Dropping unnecessary columns reduces complexity and enhances the clarity by concentrating on the most relevant information. This approach not only simplifies the dataset but also speeds up data processing, reduces the risk of errors, and improves computation times, particularly beneficial when dealing with large datasets.
Specific reasons for removing these columns include:

- Ladder score: This was excluded as didn't find rankings important.
- Whisker Columns: Typically used for showing error margins in visualizations, these are unnecessary as rankings and not taken into consideration.
- Ladder score in Dystopia: This column, often representing a hypothetical baseline, this information is irrelevant after dropping ladder score.
- Generosity: Generosity isn't directly measured in the World Happiness Report; it focuses more on factors like GDP, social support, and freedom.

```
# Removing unnecessary columns
columns_to_drop = ['Ladder score', 'Ladder score', 'Standard error of ladder score',
                   'upperwhisker', 'lowerwhisker', 'Ladder score in Dystopia', 'Generosity']
df.drop(columns=columns_to_drop, inplace=True, errors='ignore')  # 'errors' param will ignore columns not in df
```

Step 2

Step 3: Step 1 shows there are missing values in the database, and I can deal with it using the following ways:

a) Remove the rows with missing values: this won't be a good idea as we will lose a few countries.
b) Dropping row with missing values.
c) Fill the missing values with the mean of the column.

Out of these methods, I filling the missing values with the mean of the column is a better option as it will not affect the data much.

```
# Filling missing values with the mean
columns_to_fill = ['Healthy life expectancy', 'Explained by: Healthy life expectancy', 'Dystopia + residual']
for column in columns_to_fill:
    df[column].fillna(df[column].mean(), inplace=True)

print(df.info())
print("\n\nValues have been filled with the mean")
```

Step 3

Step 4: After filling the empty values, it was time to check for duplicates. Luckily The dataset I chose was clean and hence I didn't have to deal with duplicate values.

```
# Check for duplicate rows
duplicates = df[df.duplicated()]

# If there are duplicates, it will show them
if not duplicates.empty:
    print("Duplicate rows found:")
    print(duplicates)
else:
    print("No duplicate rows found.")
```

Step 4

Step 5: In this step, we adjust the values in columns 'Logged GDP per capita' and 'Healthy life expectancy'. We use a tool called MinMaxScaler, which changes the numbers in these columns so that all values are between 0 and 1. This is important because it makes sure no single value is much bigger or smaller than the others, which can mess up further analysis.

```
from sklearn.preprocessing import MinMaxScaler
scaler = MinMaxScaler()
df[['Logged GDP per capita', 'Healthy life expectancy']] = scaler.fit_transform(df[['Logged GDP per capita', 'Healthy life expectancy']])

df.head(10)
```

Step 5

Step 6: It is super important to check if the dataset has negative values and Having negative values can result to wrong analysis. We checked for all the columns and found 'Dystopia + residual' to have negative values. We then looked for the row which has negative value and found it was for 'Lebanon'. Negative value itself is smallest and no value can be zero or less than zero, hence I replaced the value with the smallest value in that column so that the analysis won't go wrong.

```
# Check for negative values in the dataset
assert (df['Logged GDP per capita'] >= 0).all(), "Negative values found in Logged GDP per capita"
assert (df['Social support'] >= 0).all(), "Negative values found in social support"
assert (df['Healthy life expectancy'] >= 0).all(), "Negative values found in Healthy life expectancy"
assert (df['Freedom to make life choices'] >= 0).all(), "Negative values found in Freedom to make life choices"
assert (df['Perceptions of corruption'] >= 0).all(), "Negative values found in Perceptions of corruption"
assert (df['Explained by: Log GDP per capita'] >= 0).all(), "Negative values found in Explained by: Log GDP per capita"
assert (df['Explained by: Social support'] >= 0).all(), "Negative values found in Explained by: Social support"
assert (df['Explained by: Healthy life expectancy'] >= 0).all(), "Negative values found in Explained by: Healthy life expectancy"
assert (df['Explained by: Freedom to make life choices'] >= 0).all(), "Negative values found in Explained by: Freedom to make life choices"
assert (df['Explained by: Generosity'] >= 0).all(), "Negative values found in Explained by: Generosity"
assert (df['Explained by: Perceptions of corruption'] >= 0).all(), "Negative values found in Explained by: Perceptions of corruption"
#assert (df['Dystopia + residual'] >= 0).all(), "Negative values found in Dystopia + residual"
print("found negative value for Dystopia + residual")

# Assuming df is your DataFrame containing the 'Logged GDP per capita' column
negative_values = df[df['Logged GDP per capita'] < 0]['Logged GDP per capita']
print("Negative values in Logged GDP per capita column:")
print(negative_values)


# removing the negative values and replacing it with the smallest positive value in the column
if df.loc[df['Country name'] == 'Lebanon', 'Dystopia + residual'].values[0] < 0:
    min_positive = df[df['Dystopia + residual'] > 0]['Dystopia + residual'].min()
    df.loc[df['Country name'] == 'Lebanon', 'Dystopia + residual'] = min_positive

assert (df['Dystopia + residual'] >= 0).all(), "Negative values found in Dystopia + residual"
```

Step 6

Step 7: Feature engineering creates new insights or predictive power from existing data. The chosen columns, 'Logged GDP per capita' and 'Healthy life expectancy,' have a significant and combined impact on a country's overall development, making them valuable for deeper analysis.

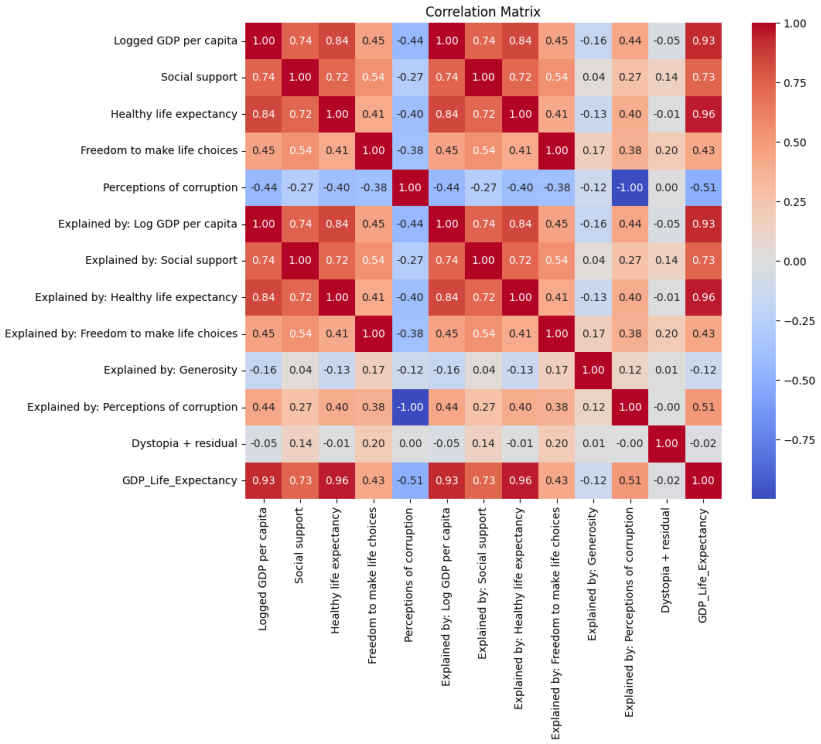
```
df['GDP_Life_Expectancy'] = df['Logged GDP per capita'] * df['Healthy life expectancy']
```

Step 7

Multivariate Analysis

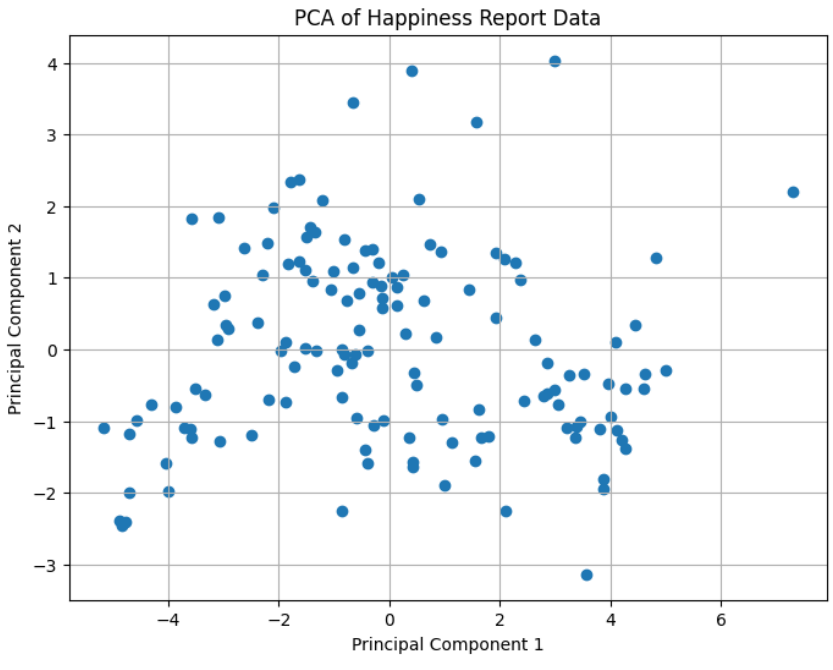
Heat Map

Used ‘heatmap’ for visualization of the correlation matrix. This process is a common multivariate analysis technique as color-coded representation making it easier to see the strength of relationships between variables.



PCA

PCA is a critical tool in Multivariate Analysis because it simplifies the complexity in high-dimensional data while retaining trends and patterns. This is particularly useful when dealing with multivariate data, allowing easier visualization and better understanding of relationships between variables.



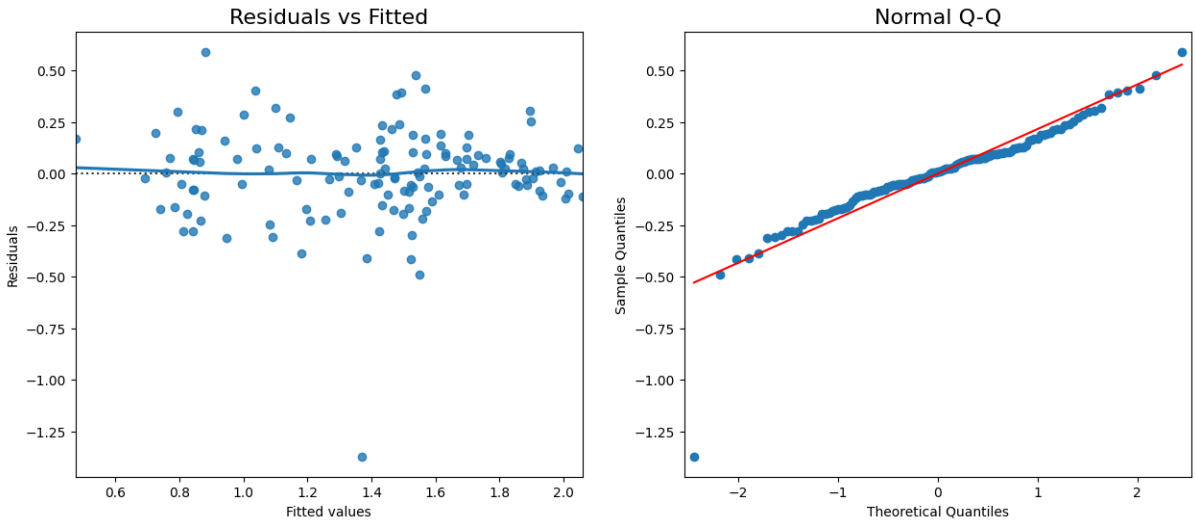
Multiple Regression Analysis

The multiple regression analysis is being used to understand how the variables 'Social support', 'Healthy life expectancy', 'Freedom to make life choices', and 'Perceptions of corruption' contribute to or influence the variable 'Explained by: Log GDP per capita'.

This step is essential because it helps examining the relationship and impact of several independent variables on a dependent variable. By doing this, you can evaluate how different factors contribute to an outcome and understand the relative importance of each factor.

Checking Model Fitting

The plot 'Residuals vs Fitted' helps to verify if the residuals (errors) have constant variance across all levels of the fitted values. This is crucial because non-constant variance (heteroscedasticity) can lead to inefficient estimates and affect the generalizability of the model. The 'Normal Q-Q' plot is used to assess if the residuals of the model are normally distributed. Normality of residuals is an important assumption in many statistical tests that involve regression models because it affects the validity of the hypothesis tests.

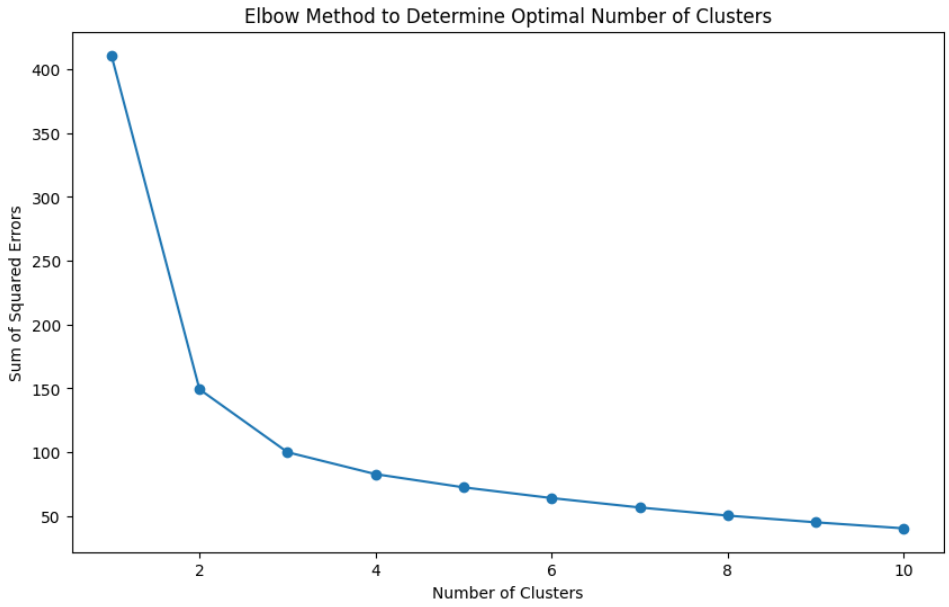


Factor Analysis

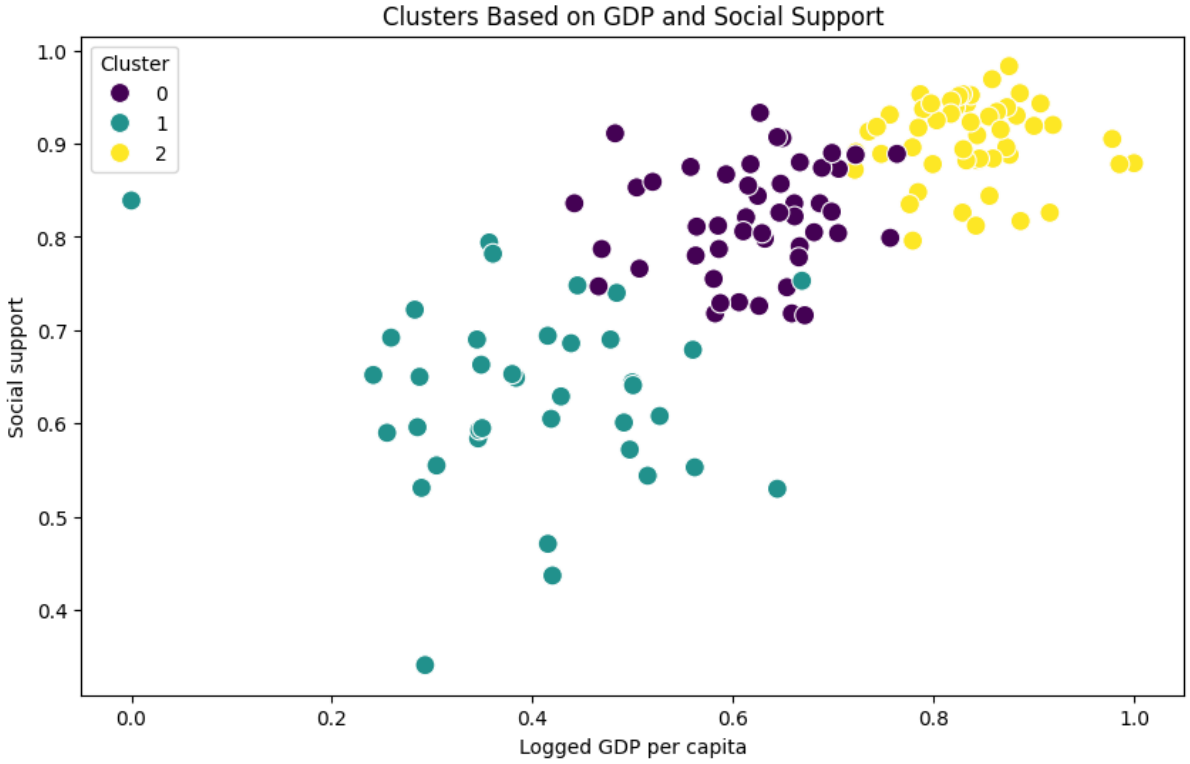
Factor Analysis is used to identify the underlying relationships between measured variables. This technique helps in simplifying data by reducing dimensions and identifying latent constructs, making it easier to analyse complex datasets with many variables.

K-Means clustering

K-Means clustering using the Elbow Method to determine the optimal number of clusters. partitioning the data into K distinct clusters and plotting the sum of squared errors (SSE) for different values of K and looking for a 'knee' in the curve.



By visually grouping data points based on similar characteristics, it provides insights into how different variables interact and cluster across the dataset.



Normalisation

Utilizes the ‘StandardScaler’ from ‘sklearn.preprocessing’ to standardize the data. I scaled the dataset so that it has a mean of zero and a standard deviation of one.

Code Example:

```python
import pandas as pd
from sklearn.preprocessing import StandardScaler


# Selecting relevant features for clustering
X = df[['Logged GDP per capita', 'Social support', 'Healthy life expectancy']]

# Standardizing the data
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
```

Weighting and Aggregation

I created a composite indicator from correlated variables and assigned explicit weights to each indicator. Then normalized these weights to ensure their total sums up to 1. Finally, recalculated a weighted average based on these normalized weights.

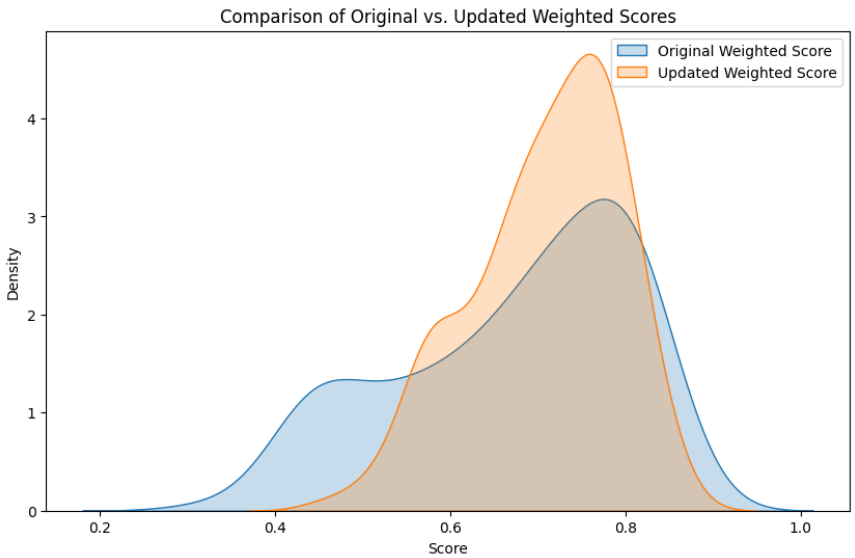
```python
# Create a composite indicator from highly correlated variables
df['Economic and Health Composite'] = df[['Logged GDP per capita', 'Social support', 'Healthy life expectancy']].mean(axis=1)
# Update weights
updated_weights = {
    'Economic and Health Composite': 0.45,
    'Freedom to make life choices': 0.25,
    'Perceptions of corruption': 0.3
}

# Normalize updated weights
total_weight = sum(updated_weights.values())
normalized_weights = {k: v / total_weight for k, v in updated_weights.items()}

# Recalculate weighted average with updated weights
df['Updated Weighted Score'] = df.apply(lambda x: sum(x[col] * normalized_weights[col] for col in normalized_weights), axis=1)
```

Code Example

Then created a graph to display original weighed score and updated weighed score.



Calculating composite index using weights.

```python
# Assigning weights (adjust these as necessary)
weights = {
    'Logged GDP per capita normalized': 0.1, 'Social support normalized': 0.2,
    'Healthy life expectancy normalized': 0.2, 'Freedom to make life choices normalized': 0.2,
    'Perceptions of corruption normalized': 0.1, 'Explained by: Log GDP per capita normalized': 0.05,
    'Explained by: Social support normalized': 0.05, 'Explained by: Healthy life expectancy normalized': 0.05,
    'Explained by: Freedom to make life choices normalized': 0.05, 'Explained by: Perceptions of corruption normalized': 0.05
}

# Calculating the Composite Index
df['Composite Index'] = sum(df[weight_name] * weight for weight_name, weight in weights.items())

# Output the updated DataFrame
print(df[['Country name', 'Composite Index']].head())

df.head(10)
```

Link To Other Indicators

Importing new dataset and comparing columns. Merged both datasets and compared all indicators.

```python
#checking how columns names are different.
print("Columns in df:", df.columns)
print("Columns in happiness_report:", happiness_report.columns)
```

✓ 0.0s

```
Columns in df: Index(['Country name', 'Logged GDP per capita', 'Social support',
       'Healthy life expectancy', 'Freedom to make life choices',
       'Perceptions of corruption', 'Explained by: Log GDP per capita',
       'Explained by: Social support', 'Explained by: Healthy life expectancy',
       'Explained by: Freedom to make life choices',
       'Explained by: Generosity', 'Explained by: Perceptions of corruption',
       'Dystopia + residual', 'GDP_Life_Expectancy', 'Cluster',
       'Logged GDP per capita normalized', 'Social support normalized',
       'GDP Category', 'GDP Bins', 'Weighted Score',
       'Economic and Health Composite', 'Updated Weighted Score',
       'Healthy life expectancy normalized',
       'Freedom to make life choices normalized',
       'Perceptions of corruption normalized',
       'Explained by: Log GDP per capita normalized',
       'Explained by: Social support normalized',
       'Explained by: Healthy life expectancy normalized',
       'Explained by: Freedom to make life choices normalized',
       'Explained by: Perceptions of corruption normalized',
       'Composite Index'],
      dtype='object')
Columns in happiness_report: Index(['Country', 'Region', 'Happiness Rank', 'Happiness Score',
       'Standard Error', 'Economy (GDP per Capita)', 'Family',
       'Health (Life Expectancy)', 'Freedom', 'Trust (Government Corruption)',
       'Generosity', 'Dystopia Residual'],
      dtype='object')
```

Comparing Columns

```python
columns_to_keep_df = [
    'Country name', 'Logged GDP per capita', 'Social support',
    'Healthy life expectancy', 'Freedom to make life choices',
    'Perceptions of corruption', 'Weighted Score', 'Composite Index'
]

# Define columns to keep for happiness_report
columns_to_keep_happiness_report = [
    'Country', 'Region', 'Happiness Rank', 'Happiness Score',
    'Economy (GDP per Capita)', 'Family', 'Health (Life Expectancy)',
    'Freedom', 'Trust (Government Corruption)', 'Generosity'
]

# Filter the datasets
df_filtered = df[columns_to_keep_df]
happiness_report_filtered = happiness_report[columns_to_keep_happiness_report]

# Assuming you want to merge on country name
df_filtered.rename(columns={'Country name': 'Country'}, inplace=True)

# Merge the datasets
merged_data = pd.merge(df_filtered, happiness_report_filtered, on='Country', how='inner')

# Optionally, save the cleaned dataset to a new file
merged_data.to_csv('merged_dataset.csv', index=False)

merged_data.head(10)
```

Merging Data

```python
import pandas as pd
import statsmodels.api as sm

# Assuming your data is loaded into a DataFrame named df
# For example: df = pd.read_csv('your_merged_data.csv')

# Define the dependent variable (composite indicator)
Y = merged_data['Composite Index']

# Define independent variables
X = merged_data[['Economy (GDP per Capita)', 'Health (Life Expectancy)', 'Freedom',  'Trust (Government Corruption)']]
X = sm.add_constant(X)  # Adds a constant term to the predictor

# Fit the regression model
model = sm.OLS(Y, X).fit()

# Print out the statistics
summary = model.summary()
print(summary)
```

OLS Regression

R-squared (0.835): This value indicates that approximately 83.5% of the variability in the Composite Index can be explained by the independent variables included in the model. This is a strong model fit.

Adj. R-squared (0.830): This is adjusted for the number of predictors and is very close to the R-squared, indicating that the model is not overfitted with unnecessary predictors.

F-statistic (157.2): The F-test is highly significant (Prob (F-statistic) = 1.44e-47), suggesting that the model is statistically significant, and the explained variance is not due to random chance.

Economy (GDP per Capita): Coefficient of 0.1309, significant ($p < 0.001$). This suggests a positive relationship with the Composite Index, indicating that as GDP per Capita increases, so does the Composite Index.

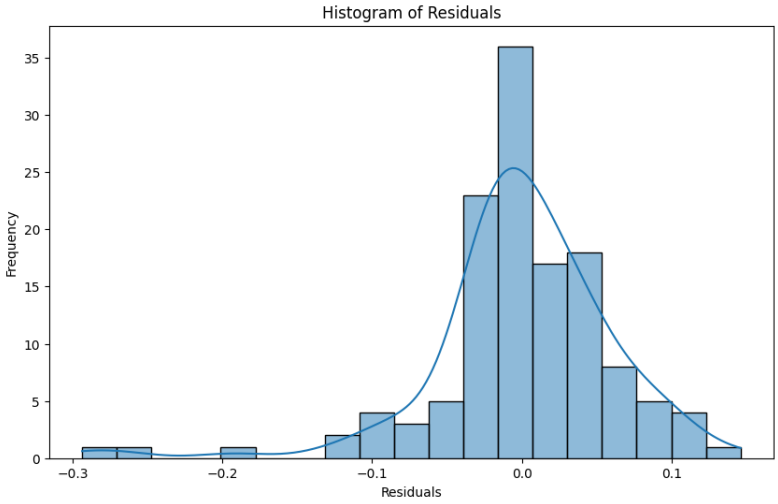
Health (Life Expectancy): Coefficient of 0.3328, also significant ($p < 0.001$). This shows a stronger positive impact on the Composite Index compared to GDP, as might be expected given that health improvements often have a broad impact on well-being.

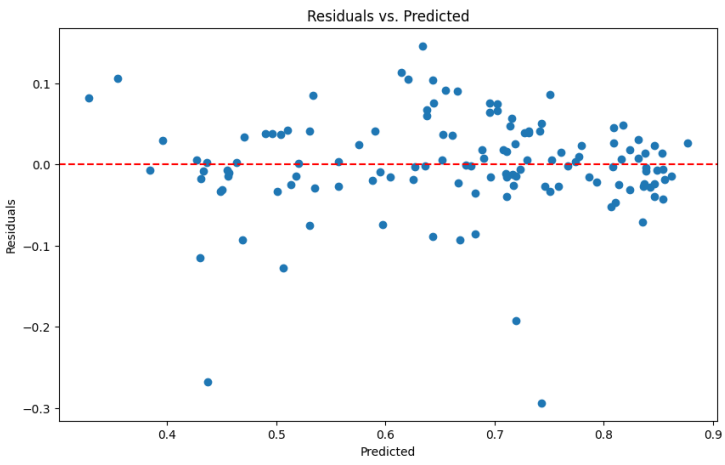
Freedom: Coefficient of -0.2601, significant ($p < 0.001$). Interestingly, this suggests a negative relationship with the Composite Index. This result is counterintuitive and may warrant further investigation.

Trust (Government Corruption): Coefficient of 0.162, significant ($p = 0.012$). Indicates a positive relationship, suggesting that higher trust in government (or lower perceived corruption) correlates with a higher Composite Index.

Ordinary Least Squares Regression Model

Diagnosing assumptions, underlying linear regression analysis, specifically looking at the normality and homogeneity of variance of the residuals.





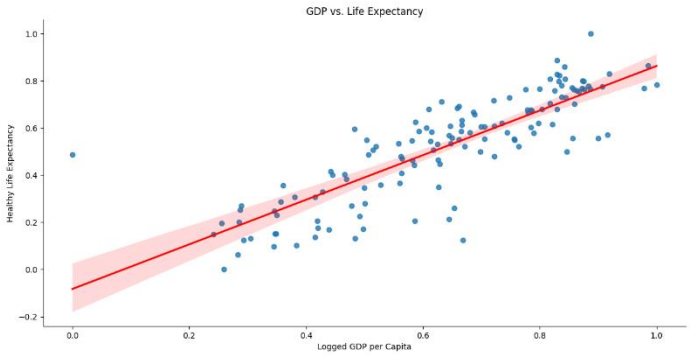
The bell-shaped curve overlay suggests that the residuals are approximately normally distributed, which is a good indicator that the model assumptions for ordinary least squares regression are being met.

The second image assesses whether the variance of the residuals is constant across all levels of predicted values. The red dashed line at y=0 helps visualize if residuals are evenly spread without any clear pattern (like a funnel or curve), which is another assumption of linear regression.

Other Visualisation of Results

Scatter Plot 1:

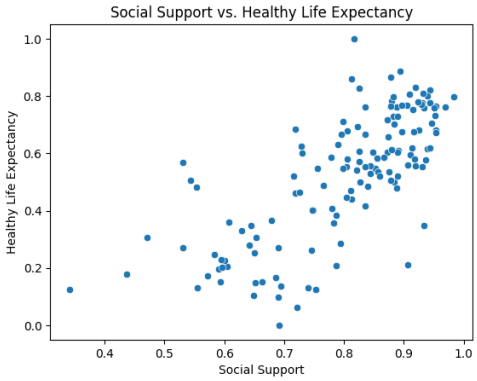
This showing the relationship between "Logged GDP per Capita" and "Healthy Life Expectancy." Each point represents a data observation, and the red line represents the best fit linear relationship, suggesting a positive correlation between higher GDP per capita and longer healthy life expectancy. The shaded area around the line indicates the confidence interval for the regression line, highlighting the potential variability in the regression estimate. This visualization is useful for observing how economic prosperity (as measured by GDP per capita) might influence health outcomes across different populations or countries.



Scatter Plot 1

Scatter Plot 2:

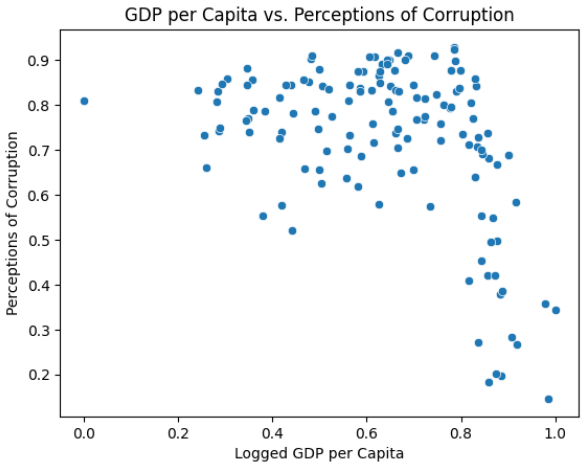
This shows relationship between "Social Support" and "Healthy Life Expectancy." Each blue dot represents a data point correlating these two variables. The plot suggests that there might be a positive correlation, indicating that higher levels of social support are generally associated with higher healthy life expectancy. The graph helps visualize the potential impact of social factors on health outcomes across a dataset.



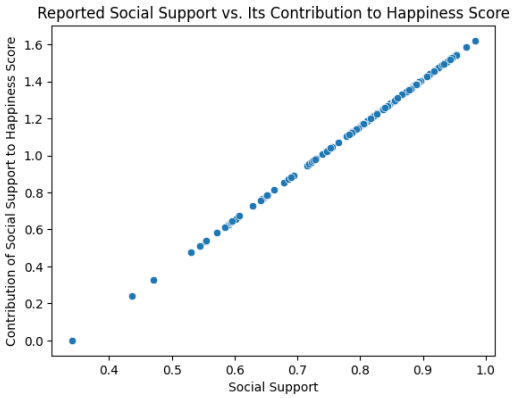
Scatter Plot 2

Scatter Plot 3:

Examining the relationship between "Logged GDP per Capita" and "Perceptions of Corruption." The x-axis represents the logarithmic transformation of GDP per capita, which typically standardizes the scale and spread of economic data, while the y-axis measures perceptions of corruption, presumably on a scale where higher values indicate greater perceptions of corruption. The plot suggests that as GDP per capita increases, the perception of corruption varies, showing no clear trend, indicating a complex or weak relationship between economic prosperity and how corruption is perceived in different contexts.



Scatter Plot 3



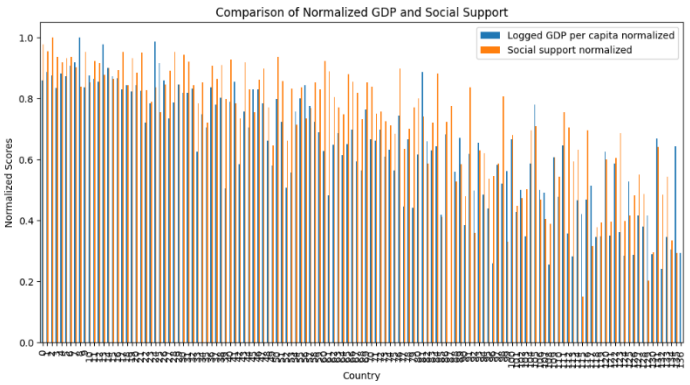
Scatter Plot 4

Scatter plot 4:

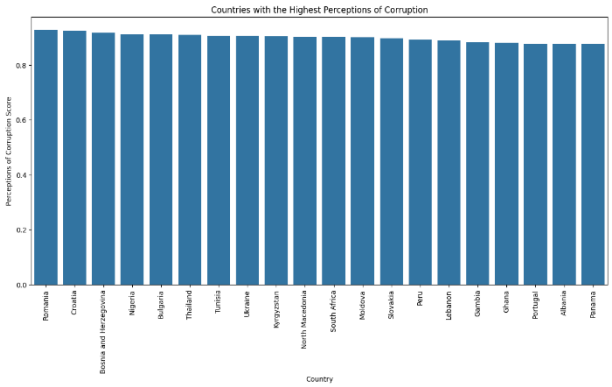
It visualizes the relationship between the level of social support (x-axis) and its contribution to a happiness score (y-axis). The plot exhibits a strong positive correlation, indicating that higher levels of reported social support are closely associated with greater contributions to overall happiness scores. This suggests that social support is a significant factor in determining happiness.

Bar Chart 1:

The graph displays a bar chart comparing normalized scores of "Logged GDP per capita" and "Social support" across various countries. Each country is represented by a pair of bars, with the blue bars indicating normalized GDP per capita and the orange bars representing normalized social support. Normalization scales these indicators to fall between 0 and 1, facilitating direct comparison across countries on a uniform scale. This visualization allows for an easy assessment of how countries compare in terms of economic output and social support within the given dataset.



Bar Chart 1



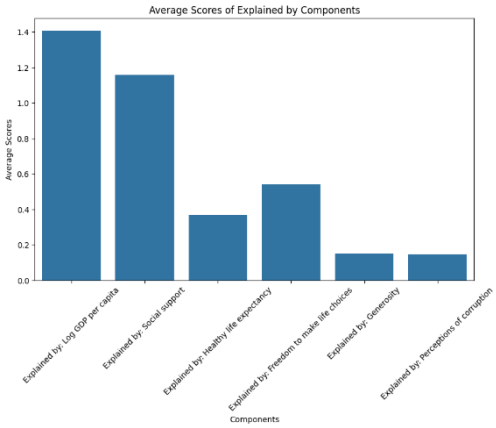
Bar Chart 2

Bar Chart 2:

The graph displays a bar chart of the top countries with the highest perceptions of corruption, based on a dataset. Each bar represents the corruption perception score for a specific country, with scores seemingly normalized or scaled between 0 and 1. Countries are ordered from left to right, with those perceived to have the highest levels of corruption listed first. This visualization is effective in comparing relative corruption perceptions across different countries.

Bar Chart 3:

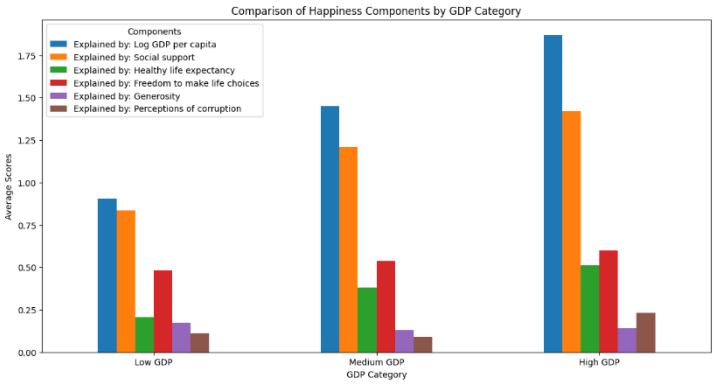
The graph displays the average scores of various components that explain happiness in different countries, as indicated by their respective contributions. These components include "Log GDP per Capita," "Social Support," "Healthy Life Expectancy," "Freedom to Make Life Choices," "Generosity," and "Perceptions of Corruption." The bars represent the mean values calculated for each component, illustrating that economic factors and social support significantly contribute to the explanatory factors of happiness compared to the other components shown.



Bar Chart 3

Bar Chart 4:

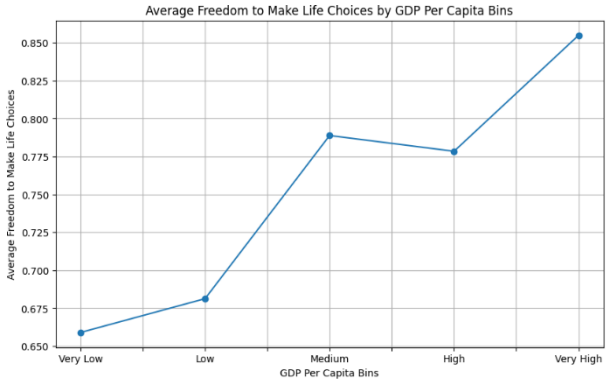
The graph displays the average scores of various components that explain happiness, categorized by three different GDP levels: Low, Medium, and High. These components include "Log GDP per Capita," "Social Support," "Healthy Life Expectancy," "Freedom to Make Life Choices," "Generosity," and "Perceptions of Corruption." Each set of colored bars represents these factors across the GDP categories, showing how the contribution of each component to happiness varies with economic prosperity.



Bar Chart 4

Linear Graph 1:

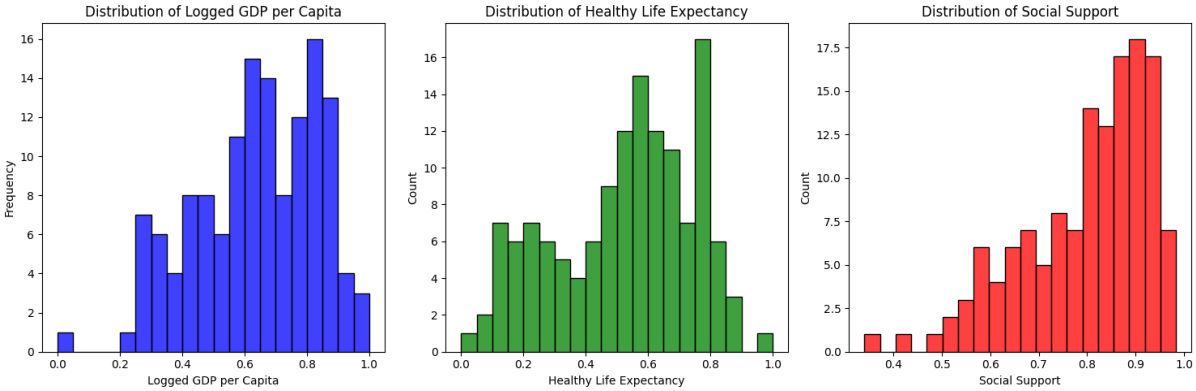
The graph displays the average "Freedom to Make Life Choices" scores across different bins of GDP per capita, ranging from "Very Low" to "Very High." It shows a generally increasing trend in the average freedom scores as GDP per capita increases. The line graph, marked by points at each GDP category, highlights that people in countries with higher GDP per capita tend to have a higher average freedom to make life choices. The increase is particularly notable between the lowest and highest GDP categories.



Linear Graph 1

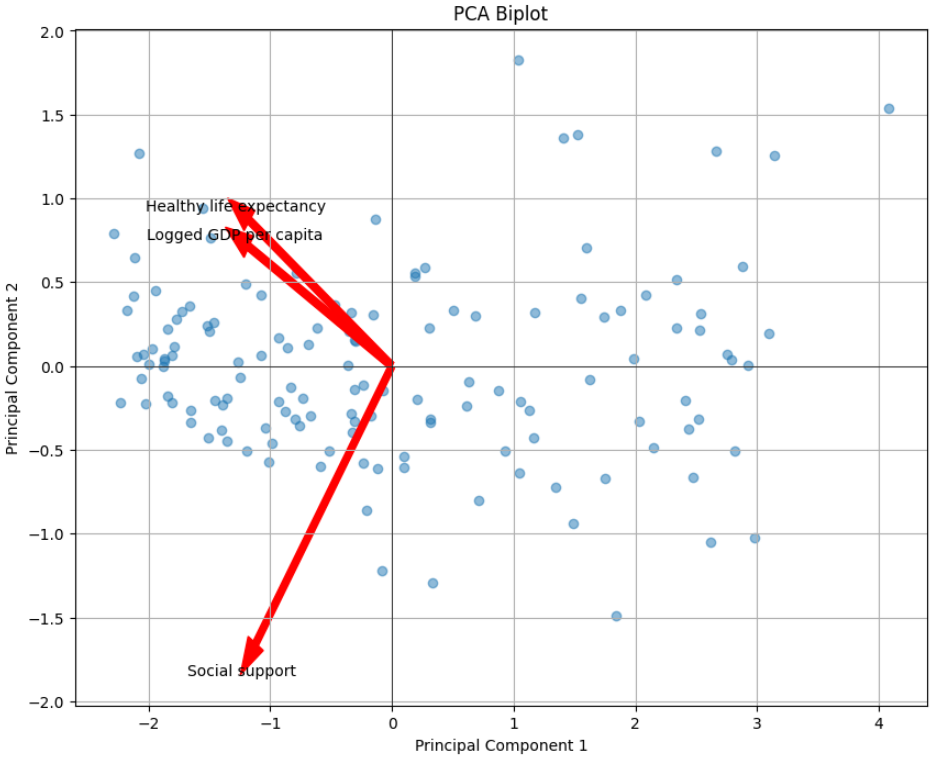
Global Indicator Distributions:

1. Distribution of Logged GDP per Capita: The first histogram (blue) shows the frequency of different ranges of logged GDP per capita values. It has a roughly bell-shaped distribution, though slightly skewed to the right, indicating most countries have a moderate level of GDP per capita, with fewer countries at the very low or very high ends.
2. Distribution of Healthy Life Expectancy: The second histogram (green) depicts the frequency of healthy life expectancy scores. It shows a strong peak in the middle range, suggesting that most countries cluster around a central range of healthy life expectancy values.
3. Distribution of Social Support: The third histogram (red) outlines the distribution of social support scores. This graph is distinctly skewed towards higher values, indicating that a larger number of countries report high levels of social support.



Principal Component Analysis Graph:

The graph is a PCA (Principal Component Analysis) biplot that displays data points for countries or observations reduced to two principal components, which capture the maximum variance in the dataset based on three variables: Logged GDP per capita, Social support, and Healthy life expectancy. The arrows represent the contribution of each variable to the principal components, illustrating how each variable influences the clustering of the data points. This biplot helps in visualizing the relationships between the variables and identifying patterns or groups within the data based on these key indicators.



Conclusion:

The analysis in the notebook effectively utilizes data from the World Happiness Report to explore and establish relationships between various factors such as GDP, life expectancy, and social support through multiple statistical techniques. By employing regression models and factor analysis, it identifies significant contributors to national well-being, providing a quantitative basis to weigh these factors in the composite index. Furthermore, the use of clustering helps to categorize countries into groups based on similar happiness traits, which can be insightful for comparative analysis. This robust approach not only aids in the construction of a composite index but also highlights the most influential factors on overall happiness, fulfilling the objective of enabling clearer, at-a-glance comparisons between countries.