Country Analysis

In [82]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
from sklearn.preprocessing import StandardScaler
from sklearn.decomposition import PCA
from sklearn.cluster import KMeans
import folium
from folium.plugins import HeatMap
import matplotlib.pyplot as plt
from statsmodels.tsa.seasonal import seasonal_decompose
from sklearn.decomposition import FactorAnalysis
from sklearn.preprocessing import StandardScaler
from sklearn.impute import SimpleImputer
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score
import statsmodels.api as sm
from scipy.cluster.hierarchy import dendrogram, linkage
from sklearn.cluster import DBSCAN
```

In [66]:
```
data = pd.read_csv('Country-data.csv')
```

In [67]:
```
data.head()
```

Out[67]:

	country	child_mort	exports	health	imports	income	inflation	life_expec	total_fer	gdpp
0	Afghanistan	90.2	10.0	7.58	44.9	1610	9.44	56.2	5.82	553
1	Albania	16.6	28.0	6.55	48.6	9930	4.49	76.3	1.65	4090
2	Algeria	27.3	38.4	4.17	31.4	12900	16.10	76.5	2.89	4460
3	Angola	119.0	62.3	2.85	42.9	5900	22.40	60.1	6.16	3530
4	Antigua and Barbuda	10.3	45.5	6.03	58.9	19100	1.44	76.8	2.13	12200

Descriptive Analysis

In [8]:
```
# Perform descriptive analysis to summarize the main features of the dataset
descriptive_stats = data.describe()

# Display the descriptive statistics for all numeric columns
descriptive_stats
```

Out[8]:

	child_mort	exports	health	imports	income	inflation	life_expec	total_fer	
count	167.000000	167.000000	167.000000	167.000000	167.000000	167.000000	167.000000	167.000000	167
mean	38.270060	41.108976	6.815689	46.890215	17144.688623	7.781832	70.555689	2.947964	12964
std	40.328931	27.412010	2.746837	24.209589	19278.067698	10.570704	8.893172	1.513848	18328
min	2.600000	0.109000	1.810000	0.065900	609.000000	-4.210000	32.100000	1.150000	231
25%	8.250000	23.800000	4.920000	30.200000	3355.000000	1.810000	65.300000	1.795000	1330
50%	19.300000	35.000000	6.320000	43.300000	9960.000000	5.390000	73.100000	2.410000	4660
75%	62.100000	51.350000	8.600000	58.750000	22800.000000	10.750000	76.800000	3.880000	14050
max	208.000000	200.000000	17.900000	174.000000	125000.000000	104.000000	82.800000	7.490000	105000

The descriptive analysis of the dataset provides a comprehensive overview of various socio-economic and health indicators for 167 countries. Here's a summary of the key statistics:

Child Mortality (child_mort): The average child mortality rate is 38.27 deaths per 1000 live births, with a wide range from 2.6 to 208, indicating significant differences in child health and survival rates across countries.

Exports (exports): On average, countries export goods and services worth 41.11% of their GDP. The exports as a percentage of GDP vary greatly among countries, from as low as 0.109% to as high as 200%.

Health Spending (health): Countries spend an average of 6.82% of their GDP on health. This percentage ranges from 1.81% to 17.9%, showing diverse priorities or capabilities in health expenditure.

Imports (imports): On average, imports account for 46.89% of the GDP, with a range from 0.0659% to 174%, indicating varying degrees of dependency on imported goods and services.

Income (income): The average income per person is $17,144.69, but there's substantial inequality, with incomes ranging from$609 to $125,000.

Inflation (inflation): The mean inflation rate is 7.78%, but it varies widely from -4.21% to 104%, reflecting different economic conditions and monetary policies.

Life Expectancy (life_expec): The average life expectancy is approximately 70.56 years, with a minimum of 32.1 years and a maximum of 82.8 years, highlighting disparities in healthcare, living conditions, and access to essential services.

Total Fertility Rate (total_fer): On average, women have about 2.95 children in their lifetime, with the total fertility rate ranging from 1.15 to 7.49 children per woman.

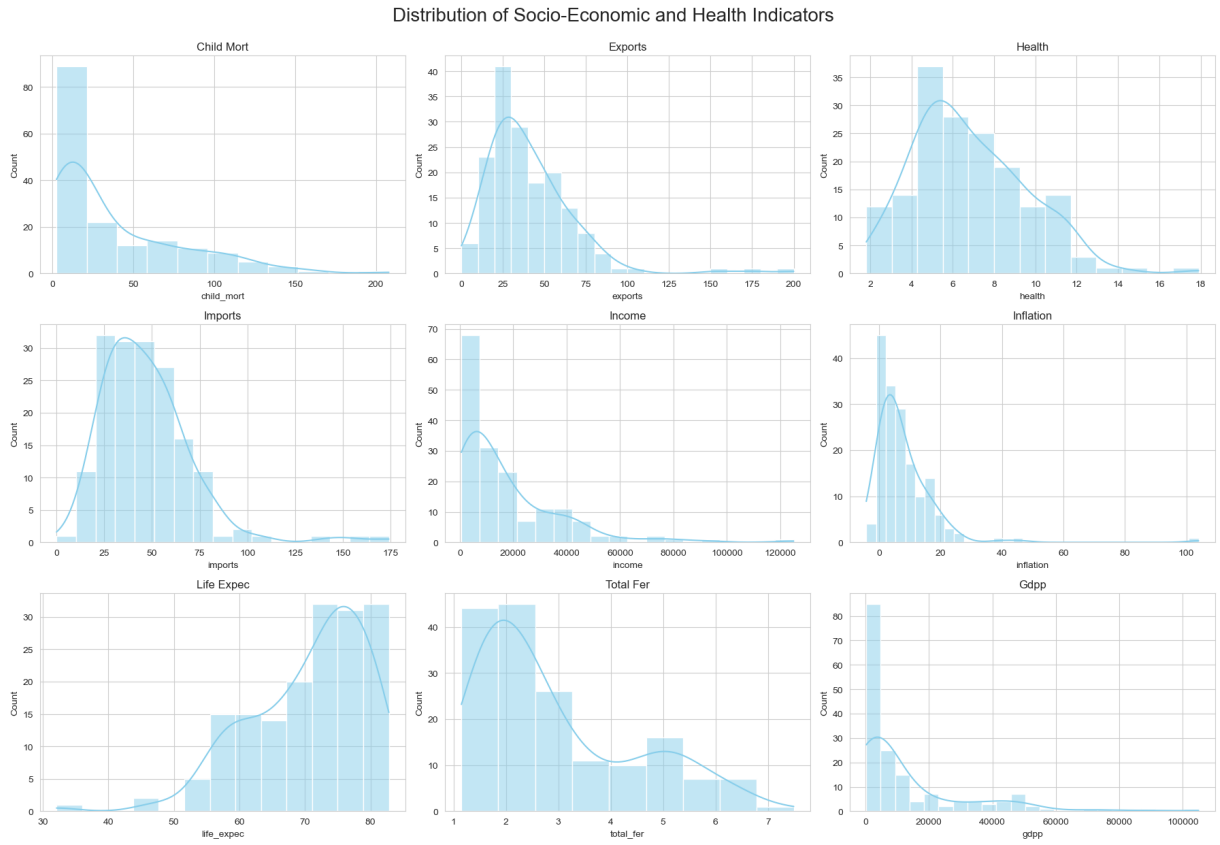
GDP per Capita (gdpp): The average GDP per capita is $12,964.16, showing a wide economic gap among countries, with values ranging from$231 to $105,000.

These statistics highlight the vast differences in economic performance, health outcomes, and demographic characteristics across countries. The data can be further analyzed to explore correlations between different indicators, identify trends, and understand the factors driving differences in health, economic well-being, and development levels among countries.

Histograms

In [10]:

```
# Set the aesthetic style of the plots
sns.set_style("whitegrid")

# Create a figure to hold multiple plots
plt.figure(figsize=(18, 12))

# List of variables for plotting
variables = ['child_mort', 'exports', 'health', 'imports', 'income', 'inflation', 'life_

# Plotting each variable in a subplot
for i, var in enumerate(variables, 1):
    plt.subplot(3, 3, i)
    sns.histplot(data[var], kde=True, color='skyblue')
    plt.title(var.replace('_', ' ').title())
    plt.tight_layout()

plt.suptitle('Distribution of Socio-Economic and Health Indicators', fontsize=20, y=1.03
plt.show()
```



The plots above display the distribution of various socio-economic and health indicators for countries in the dataset. Each histogram is accompanied by a Kernel Density Estimate (KDE) curve, providing a smooth estimate of the distribution. Here's a brief overview of what each plot reveals:

Child Mortality: Most countries have a low child mortality rate, but there's a long tail indicating some countries with very high rates.

Exports & Imports: Both show a wide range of values, with many countries clustered at the lower end, indicating a concentration of countries with lower trade as a percentage of GDP.

Health Spending: The distribution is skewed towards lower health spending as a percentage of GDP, with fewer countries spending a higher proportion.

Income: This plot shows a wide disparity in income per person, with a concentration of countries at the lower end of the income spectrum.

Inflation: Most countries have relatively low inflation rates, but there are outliers with very high inflation.

Life Expectancy: The distribution shows a skew towards higher life expectancy, with fewer countries having very low life expectancy.

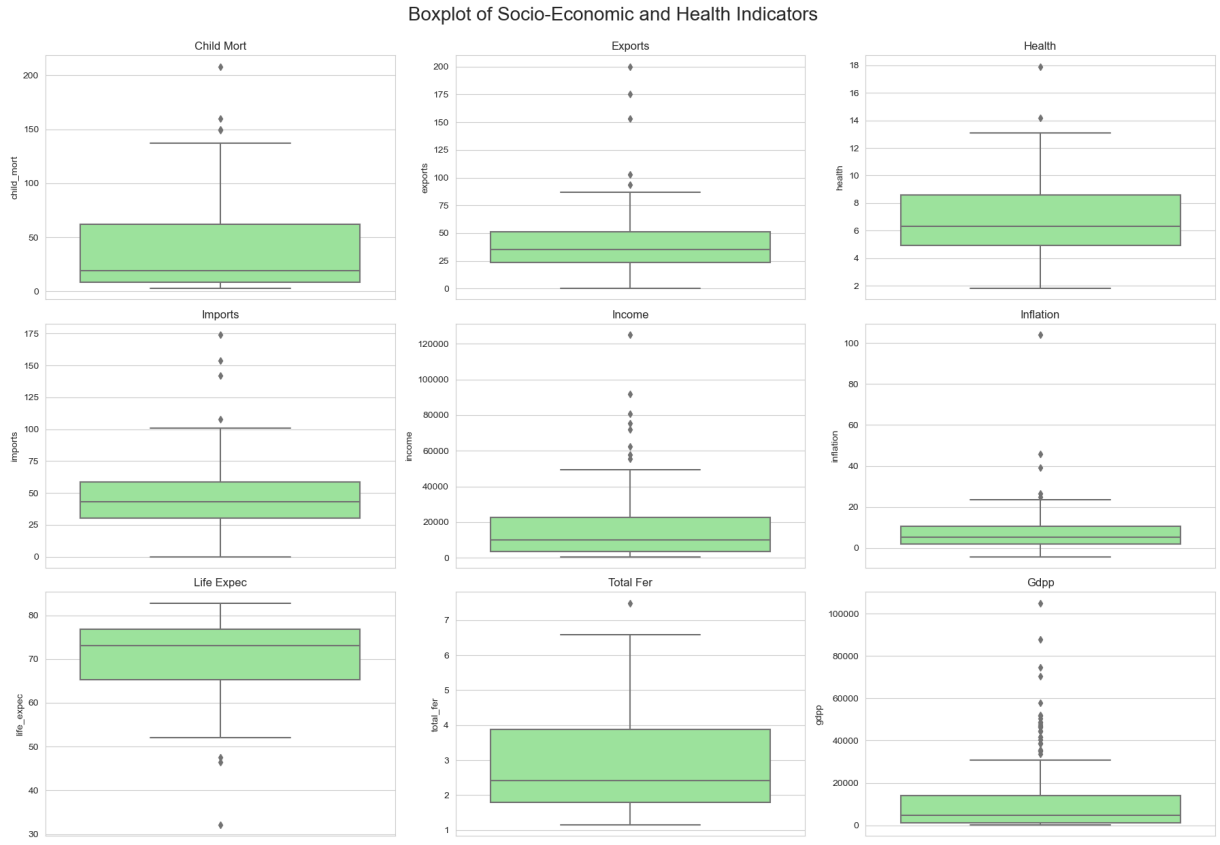
Total Fertility Rate: There's a broad spread of fertility rates, with a tendency towards lower fertility rates in many countries.

GDP per Capita (gdpp): Similar to income, there's a significant disparity in GDP per capita, with many countries concentrated at the lower end.

These plots collectively provide a visual summary of the dataset, highlighting the diversity in economic conditions

Box Plots

In [11]:
```
# Create a figure to hold multiple plots
plt.figure(figsize=(18, 12))

# Plotting each variable in a boxplot
for i, var in enumerate(variables, 1):
    plt.subplot(3, 3, i)
    sns.boxplot(y=data[var], color='lightgreen')
    plt.title(var.replace('_', ' ').title())
    plt.tight_layout()

plt.suptitle('Boxplot of Socio-Economic and Health Indicators', fontsize=20, y=1.03)
plt.show()
```



Boxplots are particularly useful for identifying the median, quartiles, and outliers within each distribution. Here's what each boxplot indicates:

Child Mortality: There's a wide interquartile range, indicating significant variation in child mortality rates among countries. Numerous outliers suggest that some countries have exceptionally high child mortality rates.

Exports & Imports: Both indicators have a relatively wide range but with many outliers, indicating that some countries have exceptionally high or low trade percentages relative to their GDP.

Health Spending: The distribution shows a moderate range of health spending as a percentage of GDP, with a few countries spending significantly more, as indicated by outliers.

Income: There's a large spread in income per person, with many outliers on the higher end, highlighting income disparity among countries.

Inflation: Most countries have moderate inflation rates, but there are several outliers with extremely high inflation.

Life Expectancy: The life expectancy across countries shows a skew towards higher values, but with outliers on both ends, indicating variations in health outcomes.

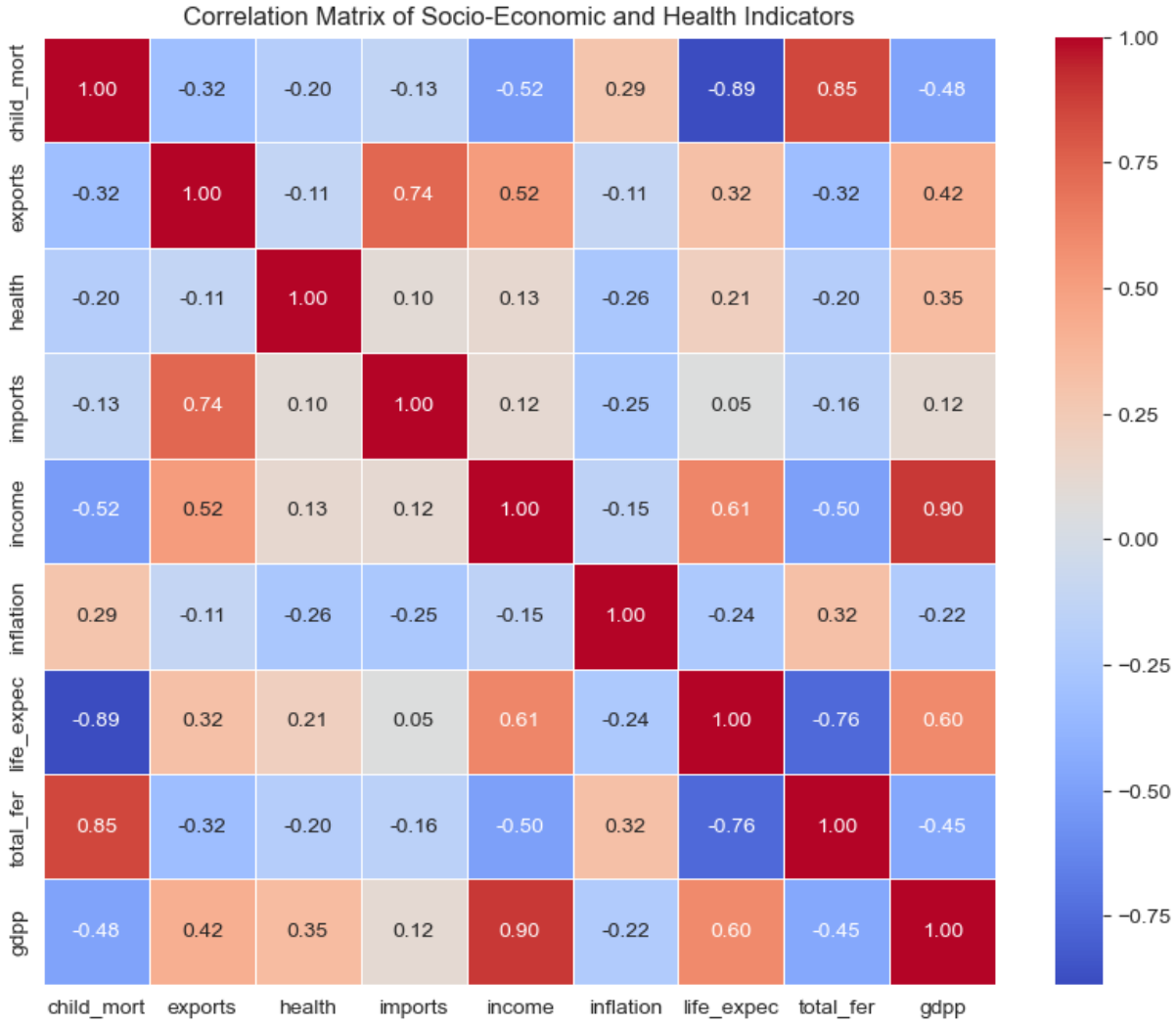
Total Fertility Rate: The fertility rate shows variability among countries, with some outliers indicating very high fertility rates.

GDP per Capita (gdpp): Similar to income, there's a significant spread in GDP per capita with numerous outliers, highlighting economic disparities.

Overall, these boxplots underscore the disparities in health, economic, and demographic indicators across countries, highlighting the range of conditions and challenges faced by different nations.

Correlation Matrix

In [12]:
```
# Calculate the correlation matrix
correlation_matrix = data.drop('country', axis=1).corr()

# Plot the heatmap for the correlation matrix
plt.figure(figsize=(10, 8))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', fmt=".2f", linewidths=.5)
plt.title('Correlation Matrix of Socio-Economic and Health Indicators')
plt.show()
```



The heatmap above visualizes the correlation matrix of socio-economic and health indicators in the dataset. Each cell in the heatmap shows the correlation coefficient between two variables, ranging from -1 to 1. A coefficient close to 1 indicates a strong positive correlation, meaning that as one variable increases, the other tends to increase as well. A coefficient close to -1 indicates a strong negative correlation, where an increase in one variable tends to be associated with a decrease in the other. Coefficients near 0 suggest little to no linear relationship between the variables.

Key insights from the correlation analysis include:

Income and GDP per Capita (gdpp) have a strong positive correlation with Life Expectancy (life_expec), indicating that higher income levels and economic output per person are associated with longer life spans.

Child Mortality (child_mort) is negatively correlated with Life Expectancy, Income, and GDP per Capita, suggesting that as economic conditions improve, child mortality rates tend to decrease.

Fertility Rate (total_fer) shows a strong negative correlation with Life Expectancy, Income, and GDP per Capita, consistent with the trend that higher fertility rates are often observed in countries with lower income and shorter life expectancy.

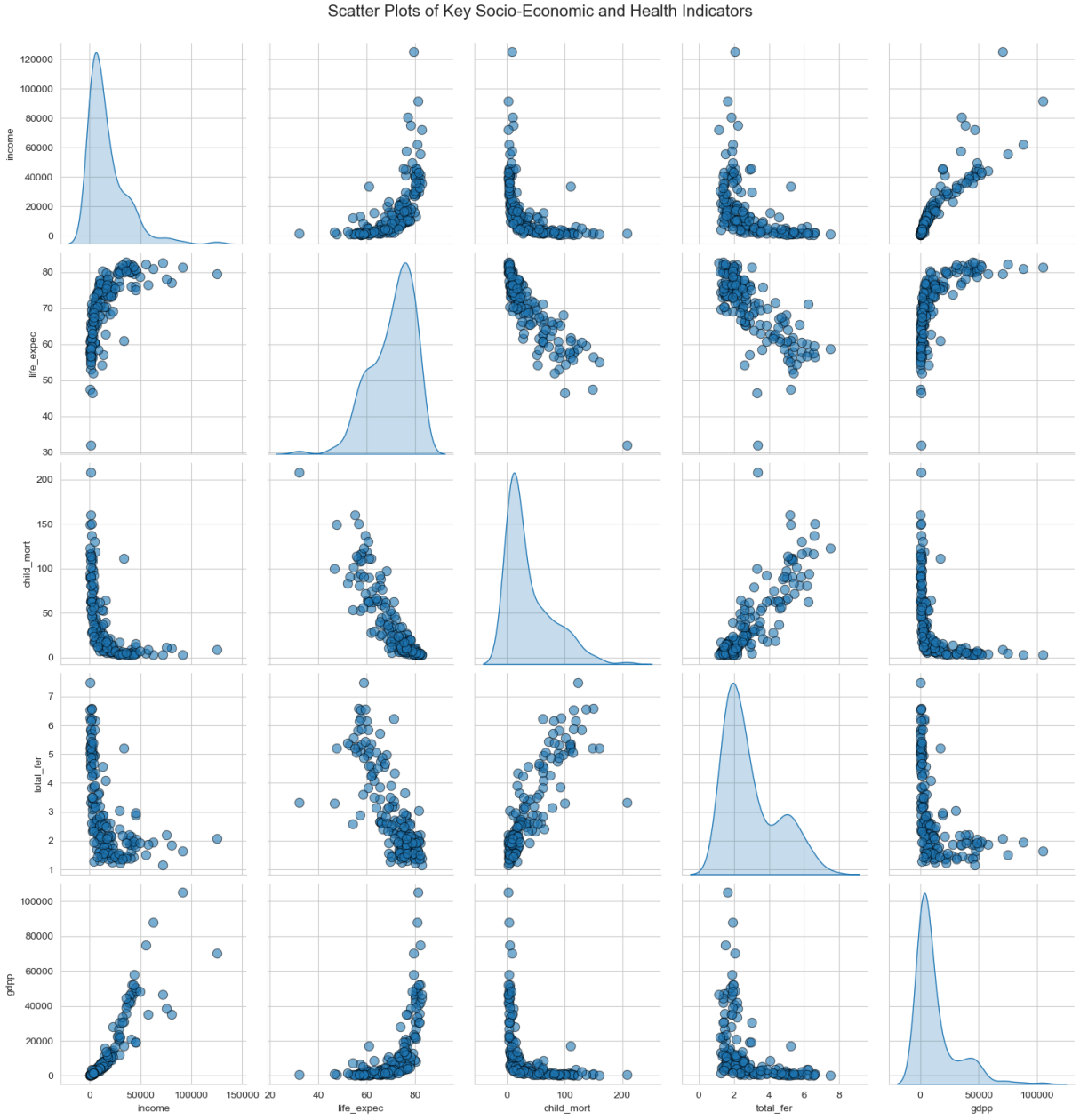
Health Spending (health) as a percentage of GDP shows positive correlations with Income and GDP per Capita, indicating that wealthier countries tend to spend a higher proportion of their GDP on health.

These correlations can help identify the key drivers of health and economic outcomes across countries, guiding further analysis and policy interventions.

Scatter Plots

In [13]:

```
# Selecting a few key variables for scatter plots to visualize their relationships
variables_to_plot = ['income', 'life_expec', 'child_mort', 'total_fer', 'gdpp']

# Create pair plot
sns.pairplot(data[variables_to_plot], diag_kind='kde', plot_kws={'alpha':0.6, 's':80, 'e
plt.suptitle('Scatter Plots of Key Socio-Economic and Health Indicators', size=16, y=1.0
plt.show()
```



The scatter plots above illustrate the relationships between key socio-economic and health indicators for the countries in the dataset. Each plot pairs two variables, showing how they are related to each other across different countries. Here are some observations:

Income vs. Life Expectancy: There's a positive relationship, indicating that higher income per person is associated with longer life expectancy.

Income vs. Child Mortality: A negative relationship is observed, suggesting that as income per person increases, child mortality rates tend to decrease.

Income vs. Total Fertility Rate: Higher income levels tend to be associated with lower fertility rates.

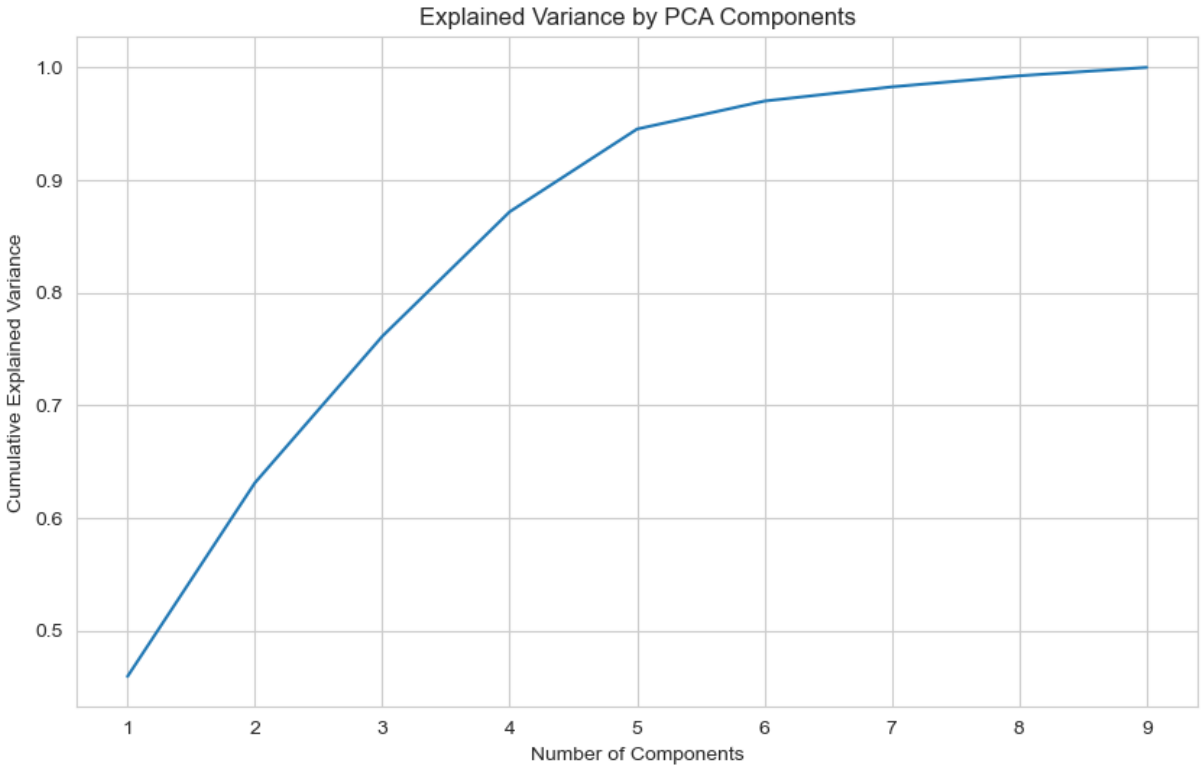
GDP per Capita vs. Life Expectancy: Similar to income, a higher GDP per capita is associated with longer life expectancy.

GDP per Capita vs. Child Mortality and Total Fertility Rate: GDP per capita shows a negative relationship with both child mortality and total fertility rate, echoing the patterns seen with income.

These patterns highlight the strong link between economic indicators and health outcomes. Wealthier countries, as measured by income and GDP per capita, typically have better health outcomes, evidenced by higher life expectancy, lower child mortality rates, and lower fertility rates. These visualizations underscore the interconnectedness of economic development and health, providing valuable insights for policymakers and

Principal Component Analysis

In [21]:
```
# Dropping the 'country' column for PCA analysis
country_data_numeric = data.drop('country', axis=1)

# Standardizing the data (mean=0, std=1) since PCA's output is influenced based on the s
scaler = StandardScaler()
country_data_standardized = scaler.fit_transform(country_data_numeric)

# Applying PCA
pca = PCA()
pca.fit(country_data_standardized)

# Plotting the explained variance by each component
plt.figure(figsize=(10, 6))
plt.plot(range(1, len(pca.explained_variance_ratio_) + 1), np.cumsum(pca.explained_varia
plt.title('Explained Variance by PCA Components')
plt.xlabel('Number of Components')
plt.ylabel('Cumulative Explained Variance')
plt.grid(True)
plt.show()
```



The plot above shows the cumulative explained variance by the principal components derived from the PCA of the dataset. The x-axis represents the number of components, and the y-axis represents the cumulative explained variance as a percentage of the total variance.

This visualization helps to determine how many principal components should be retained to capture a significant portion of the variance in the data. Ideally, you want to choose the smallest number of principal components that still captures a large proportion of the total variance.

From the plot, we can observe that the first few components already explain a substantial amount of the total variance. This suggests that these components capture the majority of the information in the dataset, allowing for dimensionality reduction without losing significant information. The curve starts to plateau as more components are added, indicating that each additional component contributes less to the explained variance.

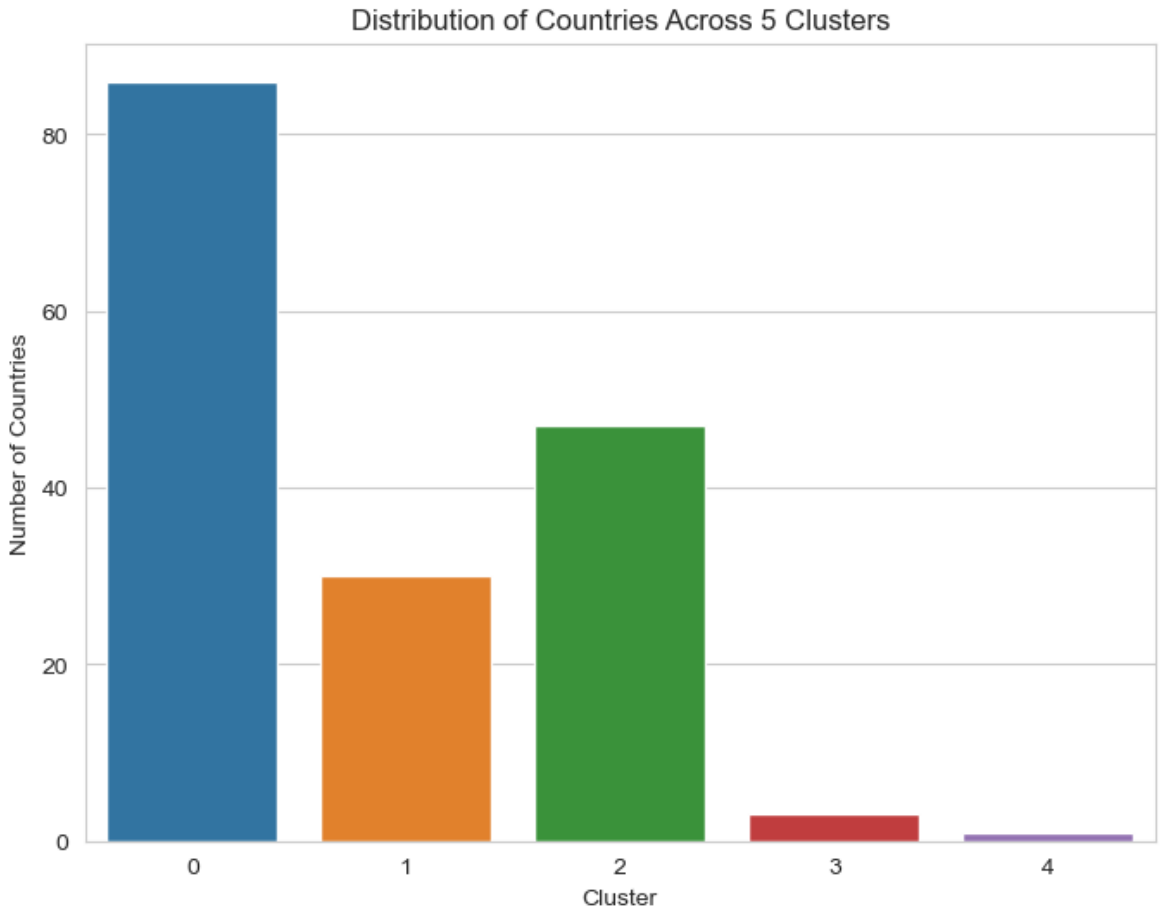
For further analysis, you might select the number of components at the point where the incremental explained variance begins to diminish significantly, often referred to as the "elbow" of the plot. This approach helps in reducing the dimensionality of the data while retaining most of the variability present in the original dataset, which

Geographical Heat Maps

In [28]:

```
# Performing PCA with a reduced number of components
# Let's choose the first 5 principal components based on the previous analysis
pca_5 = PCA(n_components=5)
country_data_pca_5 = pca_5.fit_transform(country_data_standardized)

# Applying KMeans clustering with 5 clusters
kmeans = KMeans(n_clusters=5, random_state=42)
kmeans.fit(country_data_pca_5)

# Adding the cluster labels to the original dataframe for further analysis
data['Cluster'] = kmeans.labels_

# Visualizing the distribution of countries across the clusters
plt.figure(figsize=(8, 6))
sns.countplot(x='Cluster', data=data)
plt.title('Distribution of Countries Across 5 Clusters')
plt.xlabel('Cluster')
plt.ylabel('Number of Countries')
plt.show()

# Returning a few rows of the dataframe to see the cluster labels alongside the countries
data[['country', 'Cluster']].head()
```

```
C:\Users\ttazegul\AppData\Local\anaconda3\Lib\site-packages\sklearn\cluster\_kmeans.py:1412: FutureWarning: The default value of `n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init` explicitly to suppress the warning
  super()._check_params_vs_input(X, default_n_init=10)
C:\Users\ttazegul\AppData\Local\anaconda3\Lib\site-packages\sklearn\cluster\_kmeans.py:1436: UserWarning: KMeans is known to have a memory leak on Windows with MKL, when there are less chunks than available threads. You can avoid it by setting the environment variable OMP_NUM_THREADS=1.
  warnings.warn(
```



Out[28]:

	country	Cluster
0	Afghanistan	2
1	Albania	0
2	Algeria	0
3	Angola	2
4	Antigua and Barbuda	0

The clustering analysis using KMeans with 5 clusters has been applied to the dataset, with the principal components obtained from PCA serving as the features. The distribution of countries across the 5 clusters is visualized in the plot, indicating how the countries are grouped based on their socio-economic and health indicators.

Each cluster represents a group of countries with similar characteristics in terms of the principal components derived from the original features. The count plot shows the number of countries in each cluster, highlighting the variability in cluster sizes.

From the first few rows of the dataframe with cluster labels, we see how different countries are assigned to different clusters:

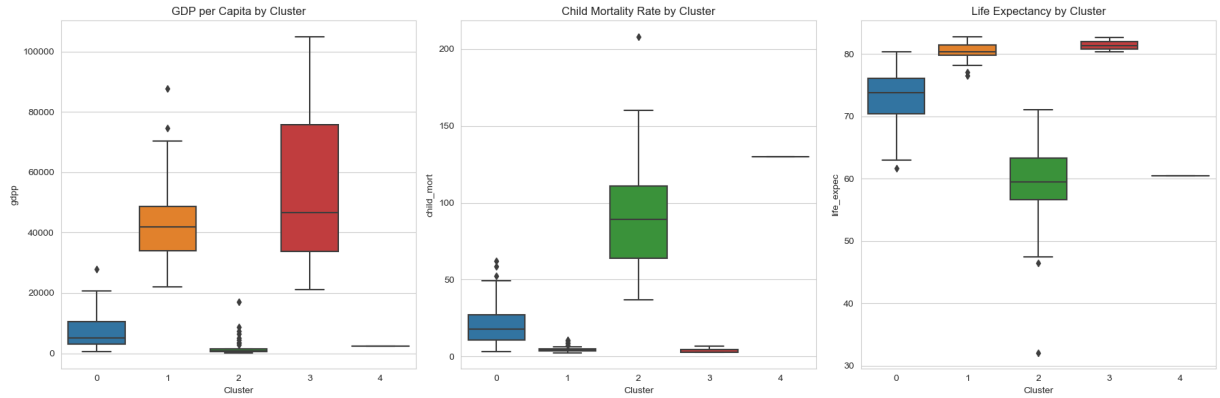
Afghanistan and Angola are in Cluster 1, possibly indicating similar socio-economic and health profiles that differ from those of countries in other clusters. Albania, Algeria, and Antigua and Barbuda are in Cluster 0, suggesting another set of shared characteristics among these countries. The clustering results can be further analyzed to understand the common traits within each cluster and how they differ from those in other clusters. This analysis can provide insights into global patterns of socio-economic development, health outcomes, and potentially guide targeted policy interventions.

Categorical Comparisons

I will create box plots for the gdpp (GDP per capita), child_mort (Child Mortality), and life_expec (Life Expectancy) indicators across the 5 clusters identified in the dataset. This will help in understanding how these key indicators vary among the clusters.

In [42]:

```
# Create figure for multiple box plots
plt.figure(figsize=(18, 6))

# GDP per Capita by Cluster
plt.subplot(1, 3, 1)
sns.boxplot(x='Cluster', y='gdpp', data=data)
plt.title('GDP per Capita by Cluster')

# Child Mortality Rate by Cluster
plt.subplot(1, 3, 2)
sns.boxplot(x='Cluster', y='child_mort', data=data)
plt.title('Child Mortality Rate by Cluster')

# Life Expectancy by Cluster
plt.subplot(1, 3, 3)
sns.boxplot(x='Cluster', y='life_expec', data=data)
plt.title('Life Expectancy by Cluster')

plt.tight_layout()
plt.show()
```



The box plots above compare the distributions of GDP per capita (gdpp), Child Mortality Rate (child_mort), and Life Expectancy (life_expec) across the 5 clusters identified in our dataset. These plots can offer valuable insights into the socio-economic and health status of countries within each cluster:

GDP per Capita by Cluster: Shows the economic disparity among clusters, with some clusters having significantly higher GDP per capita than others. This can indicate varying levels of economic development and wealth.

Child Mortality Rate by Cluster: Provides insights into the health and well-being of the youngest populations in different clusters. Clusters with higher child mortality rates may face more significant health challenges and potentially lower access to healthcare services.

Life Expectancy by Cluster: Highlights differences in overall health and living conditions among clusters. Higher life expectancy in certain clusters can reflect better health outcomes, possibly due to higher healthcare spending, better nutrition, and more effective public health policies.

These visualizations can help policymakers, researchers, and analysts to understand the multifaceted nature of development, identify which clusters may require more attention or resources, and tailor interventions to the specific needs of countries within each cluster.

Outlier Analysis

Outlier analysis involves identifying and examining data points that deviate significantly from the rest of the data. These outliers can sometimes indicate data entry errors, unusual but valid data points, or other phenomena worth investigating. In the context of your dataset, outliers across socio-economic and health indicators could reveal countries with exceptional circumstances or data anomalies.

In [44]:

```
# Define a function to identify outliers using IQR
def identify_outliers(df, column):
    Q1 = df[column].quantile(0.25)
    Q3 = df[column].quantile(0.75)
    IQR = Q3 - Q1
    lower_bound = Q1 - 1.5 * IQR
    upper_bound = Q3 + 1.5 * IQR

    # Filter outliers
    outliers = df[(df[column] < lower_bound) | (df[column] > upper_bound)]
    return outliers

# Applying the function to identify outliers for 'gdpp', 'child_mort', and 'life_expec'
outliers_gdpp = identify_outliers(data, 'gdpp')
outliers_child_mort = identify_outliers(data, 'child_mort')
outliers_life_expec = identify_outliers(data, 'life_expec')

# Display the number of outliers detected in each category
outliers_summary = {
    'Indicator': ['GDP per Capita', 'Child Mortality Rate', 'Life Expectancy'],
    'Number of Outliers': [len(outliers_gdpp), len(outliers_child_mort), len(outliers_li
}

pd.DataFrame(outliers_summary)
```

Out[44]:

	Indicator	Number of Outliers
0	GDP per Capita	25
1	Child Mortality Rate	4
2	Life Expectancy	3

The outlier analysis reveals the following number of outliers across the three key indicators:

GDP per Capita (gdpp): 25 outliers identified. These are countries with exceptionally high or low GDP per capita compared to the global distribution, possibly indicating very high-income economies or countries with significant economic challenges.

Child Mortality Rate (child_mort): 4 outliers detected. These outliers could represent countries with unusually high rates of child mortality, potentially due to healthcare access issues, nutritional deficiencies, or other public health challenges.

Life Expectancy (life_expec): 3 outliers found. These are countries with significantly higher or lower life expectancy rates, which could be due to advanced healthcare systems, lifestyle factors, or, conversely, severe health crises.

For a more detailed understanding, you might examine the specific countries that are outliers in each category and investigate the factors contributing to their exceptional status. This could involve looking into economic policies, healthcare infrastructure, or other socio-political factors in those countries.

Outlier analysis can provide valuable insights into unique or extreme cases, helping to inform targeted interventions or further research into the causes and implications of such deviations from the norm

Regression Analysis

Step 1:Data preparation

In [68]:
```
# Selecting predictors and the outcome variable
X = data[['gdpp', 'child_mort', 'income']]  # Predictors
y = data['life_expec']  # Outcome

# Handling any potential missing values
# Assuming missing values have been imputed or there are none

# Splitting the dataset into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42
```

Set 2:Linear Regression Model

In [72]:
```
# Initializing and fitting the linear regression model
model = LinearRegression()
model.fit(X_train, y_train)

# Predicting on the test set
y_pred = model.predict(X_test)
```

Out[72]:
```
▾ LinearRegression

LinearRegression()
```

Step 3: Model Evaluation

In [70]:
```
# Evaluating the model
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)

print(f"Mean Squared Error: {mse}")
print(f"R^2 Score: {r2}")
```

```
Mean Squared Error: 11.130087923092166
R^2 Score: 0.8312795096170903
```

Additional Insights: Model Coefficients

To understand the impact of each predictor on life expectancy, you can examine the model's coefficients

In [71]:
```
# Displaying the coefficients
coefficients = pd.DataFrame(model.coef_, X.columns, columns=['Coefficient'])
print(coefficients)
```

```
             Coefficient
gdpp        1.028339e-04
child_mort -1.727082e-01
income     -8.811430e-07
```

This code snippet outlines the process for applying linear regression analysis to explore how selected socio-economic and health indicators predict life expectancy. The model's performance is evaluated using the Mean Squared Error (MSE) and the R-squared (R^2) score, which provides insights into the model's accuracy and the proportion of variance in the dependent variable explained by the model. The coefficients give an indication of the importance and direction of the relationship between each predictor and the outcome variable.

In [74]:

```
# Adding a constant to the predictors for the intercept
X_train_sm = sm.add_constant(X_train)

# Fitting the model
model_sm = sm.OLS(y_train, X_train_sm).fit()

# Printing the detailed summary
print(model_sm.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:             life_expec   R-squared:                       0.822
Model:                            OLS   Adj. R-squared:                  0.818
Method:                 Least Squares   F-statistic:                     199.0
Date:                Fri, 09 Feb 2024   Prob (F-statistic):           3.35e-48
Time:                        18:38:02   Log-Likelihood:                -366.48
No. Observations:                 133   AIC:                             741.0
Df Residuals:                     129   BIC:                             752.5
Df Model:                           3
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const         75.7659      0.722    104.933      0.000      74.337      77.194
gdpp           0.0001   3.93e-05      2.620      0.010    2.52e-05       0.000
child_mort    -0.1727      0.010    -17.912      0.000      -0.192      -0.154
income     -8.811e-07   3.97e-05     -0.022      0.982   -7.95e-05    7.77e-05
==============================================================================
Omnibus:                       14.696   Durbin-Watson:                   2.350
Prob(Omnibus):                  0.001   Jarque-Bera (JB):               17.176
Skew:                          -0.698   Prob(JB):                     0.000186
Kurtosis:                       4.072   Cond. No.                     7.37e+04
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 7.37e+04. This might indicate that there are
strong multicollinearity or other numerical problems.
```

Coefficients tell you the expected change in the dependent variable for a one-unit change in an independent variable, assuming other variables are held constant.

The R-squared value gives the proportion of variance in the dependent variable that is predictable from the independent variables.

p-values for each coefficient test the null hypothesis that the coefficient is equal to zero (no effect). A low p-value (<0.05) indicates that you can reject the null hypothesis.

This approach with statsmodels gives you a comprehensive view of your model's performance and the relationships between variables, allowing for a deeper understanding and interpretation of your linear regression analysis.

In [78]:

```
# Hypothetical new data for prediction
new_data = pd.DataFrame({
    'gdpp': [10000, 20000, 30000],
    'child_mort': [10, 20, 30],
    'income': [40000, 50000, 60000]
})

```

In [79]:

```
# Making predictions with the new data
predicted_life_expec = model.predict(new_data)

# Displaying the predictions
predicted_life_expec_df = pd.DataFrame(predicted_life_expec, columns=['Predicted Life Ex
print(predicted_life_expec_df)
```

```
   Predicted Life Expectancy
0                  75.031927
1                  74.324373
2                  73.616820
```

Assuming we have a new data for GDP per capita (gdpp), child mortality (child_mort), and income (income), which you want to use to predict life expectancy. The new data should be structured in the same format as the data used to train the model.

The output will be a DataFrame (predicted_life_expec_df) with the predicted life expectancy for each row of new data provided. These predictions are based on the relationships the model has learned between the predictors (gdpp, child_mort, income) and the outcome (life expectancy) from the training data.