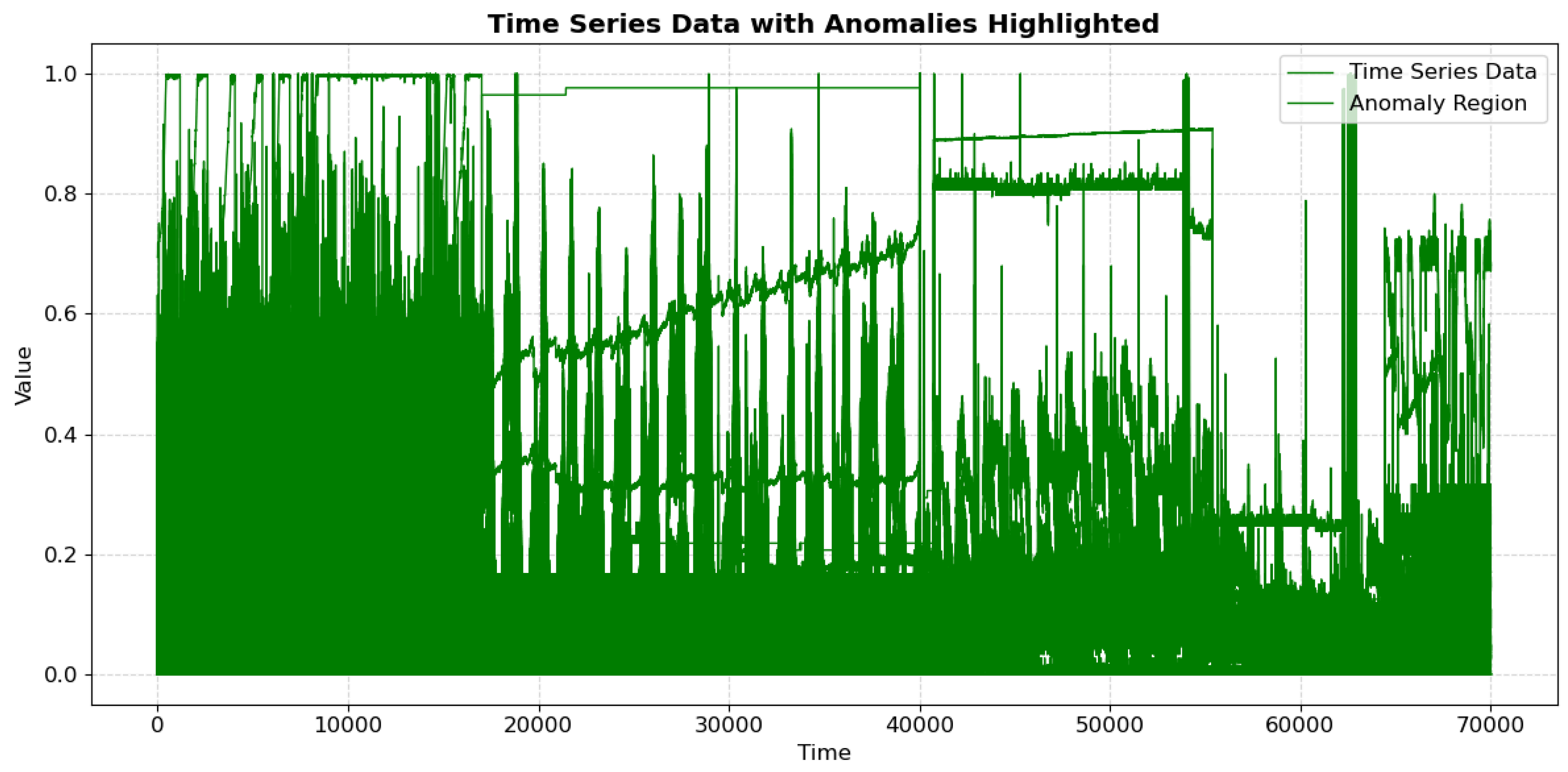
In [32]:

```python
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.ensemble import RandomForestClassifier
from tabulate import tabulate

# Step 1: Read the time series data and anomaly labels from CSV files
time_series_data = pd.read_csv("test.csv")
anomaly_labels = pd.read_csv("test_label.csv")

# Step 2: Plot the time series data with anomalies highlighted
def plot_with_anomalies(data, labels):
    # Create a plot
    plt.figure(figsize=(12, 6))

    # Plot the time series data
    plt.plot(data.index, data.values, color='green', label='Time Series Data', linewidth=1)

    # Check for anomalies and highlight them
    if 'Label' in labels.columns:
        anomalies = labels[labels['Label'] == 1]
        for _, row in anomalies.iterrows():
            plt.axvspan(row['Start'], row['End'], color='red', alpha=0.3)  # Change color to orange

    # Add legend and labels
    plt.legend(['Time Series Data', 'Anomaly Region'], loc="upper right", fontsize='large')
    plt.xlabel('Time', fontsize='large')
    plt.ylabel('Value', fontsize='large')
    plt.title('Time Series Data with Anomalies Highlighted', fontsize='x-large', fontweight='bold')
    plt.tick_params(axis='both', which='major', labelsize='large')
    plt.grid(True, linestyle='--', alpha=0.5)

    # Show plot
    plt.tight_layout()
    plt.show()

# Call the function to plot the time series data with anomalies
plot_with_anomalies(time_series_data, anomaly_labels)

# Step 3: Perform Exploratory Data Analysis (EDA)
print("Basic Statistical Information:")
statistical_info_table = time_series_data.describe().transpose()

# Display the Basic Statistical Information in a table format
print(tabulate(statistical_info_table.reset_index().rename(columns={'index': 'Serial No.'}), headers='keys', tablefmt='fancy_grid', showindex=False))

# Step 4: Find potential root causes using feature importance analysis
random_forest = RandomForestClassifier(n_estimators=100, random_state=42)
random_forest.fit(time_series_data, anomaly_labels.values.ravel())  # Train the model
feature_importances = random_forest.feature_importances_
feature_importance_df = pd.DataFrame({'Feature': time_series_data.columns, 'Importance': feature_importances})
feature_importance_df = feature_importance_df.sort_values(by='Importance', ascending=False)

# Display the top features contributing to anomalies in a table format
print("\nTop 5 features contributing to anomalies:")
top_features_table = feature_importance_df.head(5)
print(tabulate(top_features_table.reset_index(drop=True), headers='keys', tablefmt='fancy_grid', showindex=False))
```



Basic Statistical Information:

Serial No.	count	mean	std	min	25%	50%	75%	max
0	70001	0.125281	0.14853	0	0.010309	0.070707	0.191919	1
1	70001	0.0254237	0.0723878	0	0.001495	0.004785	0.020927	1
2	70001	0.0344154	0.0886287	0	0.001873	0.005242	0.02927	1
3	70001	0.0374622	0.0935202	0	0.002546	0.006364	0.031797	1
4	70001	0.322057	0.456736	0	0	0	0.976471	0.976471
5	70001	0.459721	0.347015	0	0.068121	0.534963	0.685973	1
6	70001	0.336344	0.312518	0	0.060259	0.309052	0.346061	1
7	70001	0	0	0	0	0	0	0
8	70001	0.0112258	0.056158	0	5.8e-05	0.001867	0.006221	1
9	70001	0.000749263	0.0151993	0	0	0	0	1
10	70001	0.0564138	0.0985104	0	0	0	0.102237	1
11	70001	0.0616159	0.0471572	0	0.028846	0.056485	0.083333	1
12	70001	0.0255832	0.0461581	0	0	0	0.04	1
13	70001	0.0804306	0.0871121	0	0.010805	0.064179	0.107667	0.938554
14	70001	0.0474983	0.081024	0	0.002456	0.016933	0.073489	1
15	70001	0.0833762	0.124425	0	0.000948	0.042659	0.117584	1
16	70001	0.000145712	0.00932189	0	0	0	0	1
17	70001	7.42417e-05	0.00536997	0	0	0	0	1
18	70001	0.112591	0.166105	0	0.015493	0.040706	0.108727	1
19	70001	0.10716	0.124291	0	0.021188	0.062946	0.134593	1
20	70001	0.142572	0.162469	0	0.033279	0.08323	0.156046	1
21	70001	0.159021	0.157019	0	0.039345	0.113603	0.222237	1
22	70001	0.147304	0.164831	0	0.028409	0.085668	0.195522	1
23	70001	0.267206	0.286852	0	0.024874	0.171194	0.256881	1
24	70001	0.165341	0.162178	0	0.036304	0.130055	0.227723	1
25	70001	0.263033	0.289581	0	0.024194	0.165146	0.252336	1
26	70001	0	0	0	0	0	0	0
27	70001	0.155052	0.163294	0	0.045776	0.091743	0.208649	1
28	70001	7.50406e-06	0.0017396	0	0	0	0	0.455867
29	70001	0.16905	0.0755812	0	0.125	0.149425	0.218391	0.875
30	70001	0.191246	0.170423	0	0.06152	0.137931	0.254096	1
31	70001	0.104893	0.187695	0	0	0.001101	0.114316	1
32	70001	0.0187077	0.0546645	0	0	0	0	1
33	70001	0.0515991	0.0474198	0	0.013514	0.028623	0.089041	1
34	70001	0.269691	0.21386	0	0.099265	0.224793	0.396694	1
35	70001	0.229657	0.230937	0	0.042573	0.137634	0.391304	1
36	70001	0	0	0	0	0	0	0
37	70001	0	0	0	0	0	0	0

Top 5 features contributing to anomalies:

Feature	Importance
5	0.129956
2	0.0647632
3	0.064378
1	0.05955

35	0.0510017

In [ ]: