inflation

January 12, 2025

1 This notebook aims to analyze and visualize inflation-related data, including the Consumer Price Index(CPI) from 2000-most recent data.

```
[1]: import pandas as pd
     import matplotlib.pyplot as plt
```

```
[2]: file_path = (r"C:\Users\lamarwells\OneDrive - Rasmussen,␣
      ↪Inc\AnacondaProjects\CPI2000_2020.csv")
```

```
[ ]: df = pd.read_csv(file_path)
```

1.1 This will provide some summary statistics/high level overview

```
[10]: df.describe()
```

```
[10]:              Year        Jan        Feb        Mar        Apr        May  \
      count    25.000000  25.000000  25.000000  25.000000  25.000000  25.000000
      mean   2012.000000   2.368000   2.376000   2.392000   2.376000   2.380000
      std       7.359801   1.188388   1.227219   1.226757   1.182962   1.160101
      min    2000.000000   1.000000   1.100000   1.100000   0.900000   0.900000
      25%    2006.000000   1.700000   1.700000   1.800000   1.800000   1.700000
      50%    2012.000000   2.200000   2.200000   2.100000   2.100000   2.200000
      75%    2018.000000   2.500000   2.400000   2.400000   2.300000   2.400000
      max    2024.000000   6.000000   6.400000   6.500000   6.200000   6.000000

                   Jun        Jul        Aug        Sep        Oct        Nov  \
      count  25.000000  24.000000  24.000000  24.000000  24.000000  24.000000
      mean    2.392000   2.370833   2.358333   2.358333   2.350000   2.354167
      std     1.143431   1.112959   1.130954   1.165786   1.152313   1.139405
      min     0.900000   0.900000   0.900000   0.800000   0.600000   0.800000
      25%     1.700000   1.775000   1.700000   1.700000   1.800000   1.700000
      50%     2.200000   2.150000   2.100000   2.050000   2.100000   2.100000
      75%     2.500000   2.500000   2.525000   2.525000   2.350000   2.375000
      max     5.900000   5.900000   6.300000   6.600000   6.300000   6.000000

                   Dec      HALF1      HALF2
```

```
count  24.000000  25.000000  24.000000
mean    2.366667   2.380000   2.358333
std     1.157083   1.144552   1.122465
min     0.800000   1.100000   0.800000
25%     1.800000   1.800000   1.700000
50%     2.200000   2.200000   2.150000
75%     2.450000   2.400000   2.350000
max     5.700000   6.200000   6.100000
```

[11]:

```
months = df[['Jan', 'Feb', 'Mar', 'Apr']]
```

[12]:

```
print(type(months))
```

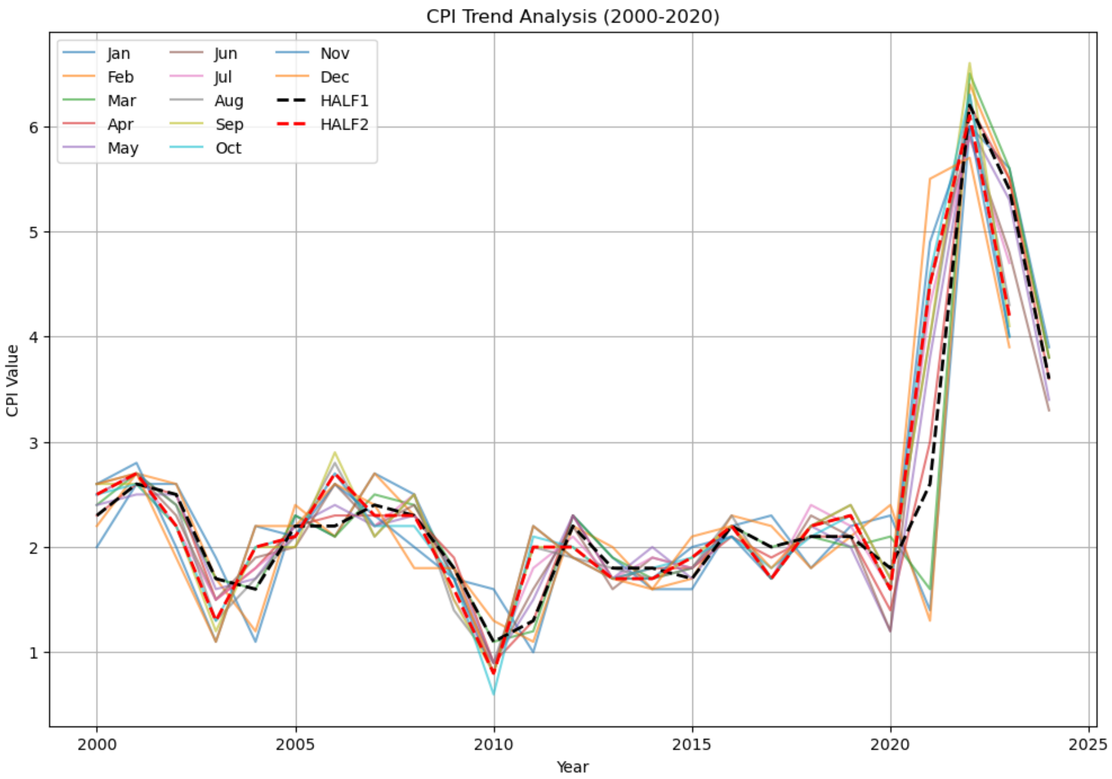
```
<class 'pandas.core.frame.DataFrame'>
```

[13]:

```
years = df['Year']
months = df[['Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun', 'Jul', 'Aug', 'Sep',␣
 ↪'Oct', 'Nov', 'Dec']]
half1 = df['HALF1']
half2 = df['HALF2']
```

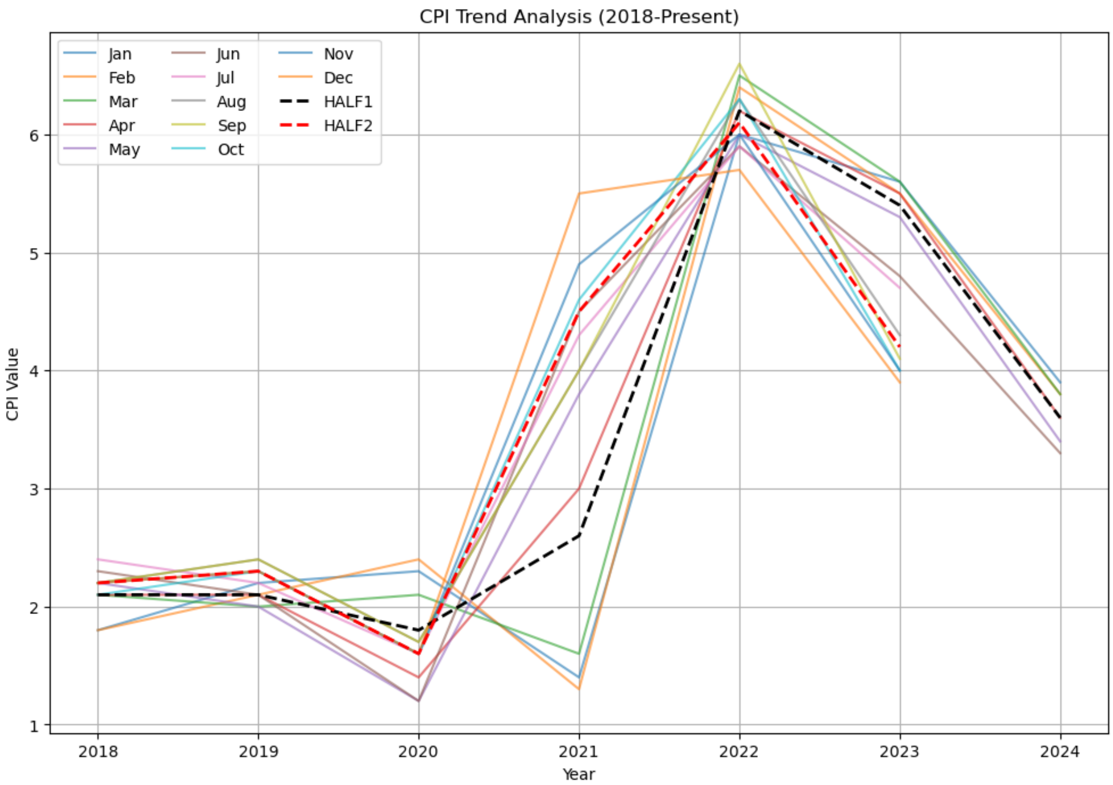
[14]:

```
plt.figure(figsize=(12, 8))
for month in months.columns:
    plt.plot(years, df[month], label=month, alpha=0.6)

# Adding half-yearly averages to the plot
plt.plot(years, half1, label='HALF1', color='black', linestyle='--',␣
 ↪linewidth=2)
plt.plot(years, half2, label='HALF2', color='red', linestyle='--', linewidth=2)

plt.title('CPI Trend Analysis (2000-2020)')
plt.xlabel('Year')
plt.ylabel('CPI Value')
plt.legend(loc='upper left', ncol=3)
plt.grid(True)
plt.show()
```

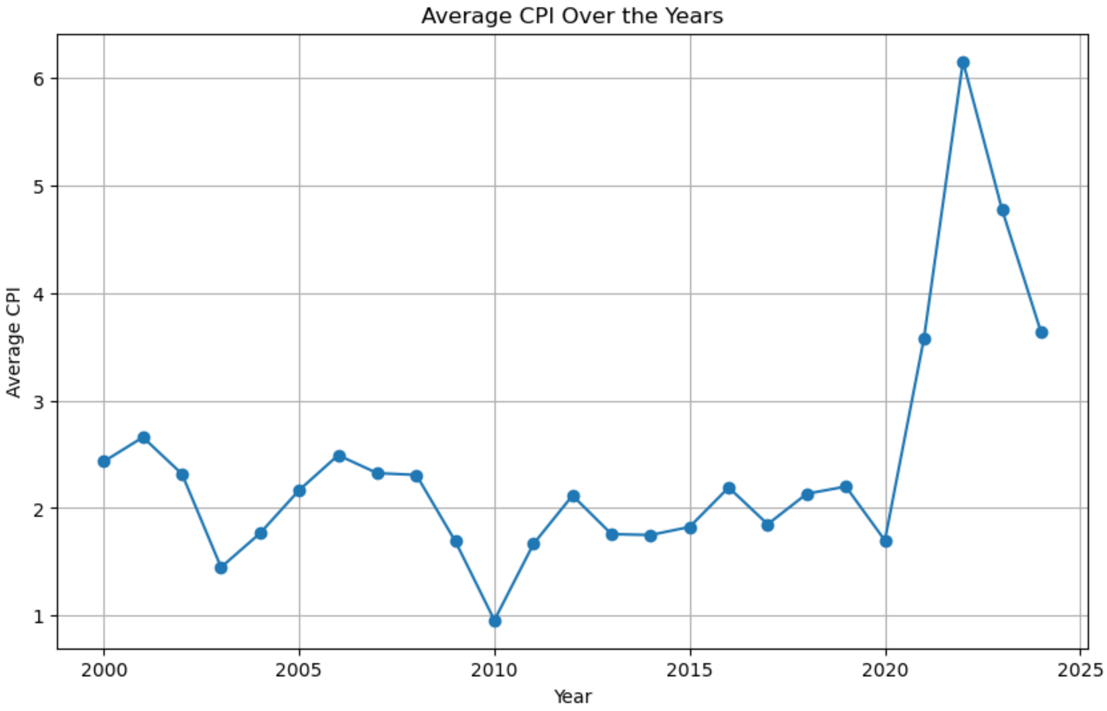


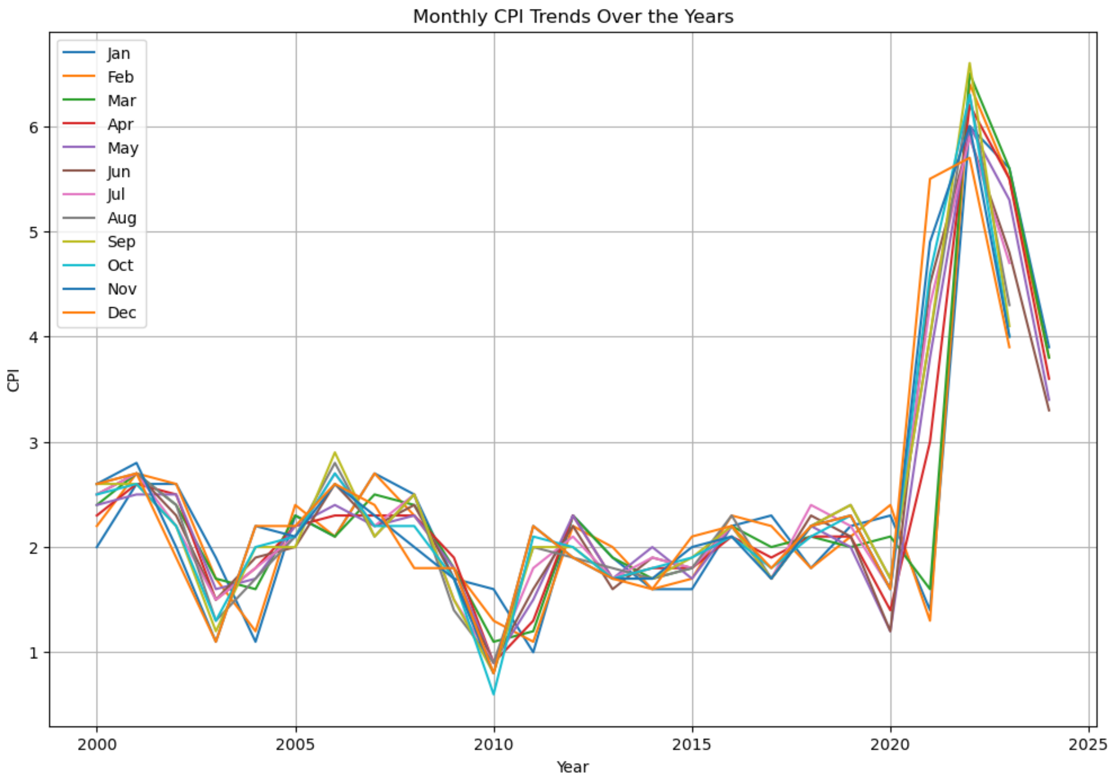
```
[15]: df_recent = df[df['Year'] >= 2018]
      years_recent = df_recent['Year']
      months_recent = df_recent[['Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun', 'Jul',␣
        ↪'Aug', 'Sep', 'Oct', 'Nov', 'Dec']]
      half1_recent = df_recent['HALF1']
      half2_recent = df_recent['HALF2']
```

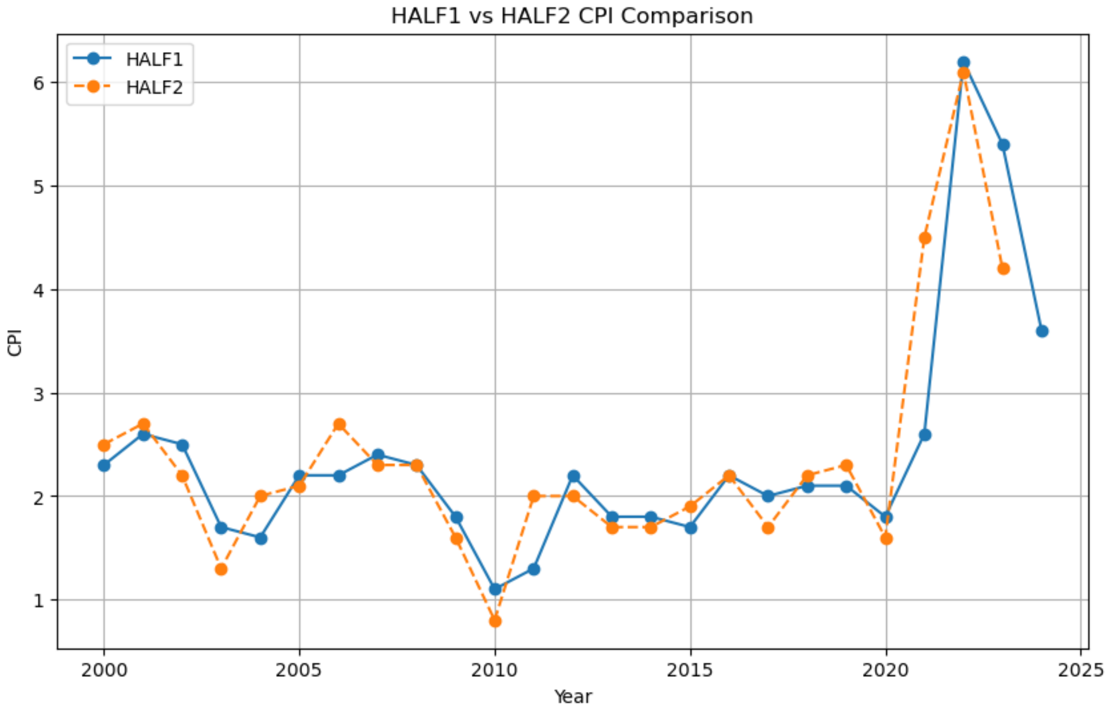
```
[16]: plt.figure(figsize=(12, 8))
      for month in months_recent.columns:
          plt.plot(years_recent, df_recent[month], label=month, alpha=0.6)
      plt.plot(years_recent, half1_recent, label='HALF1', color='black',␣
        ↪linestyle='--', linewidth=2)
      plt.plot(years_recent, half2_recent, label='HALF2', color='red',␣
        ↪linestyle='--', linewidth=2)

      plt.title('CPI Trend Analysis (2018-Present)')
      plt.xlabel('Year')
      plt.ylabel('CPI Value')
      plt.legend(loc='upper left', ncol=3)
      plt.grid(True)
      plt.show()
```



[18]:
```
import matplotlib.pyplot as plt

# Summary statistics
summary_stats = df.describe()

# Plot CPI values over the years (using the average of all months)
df['Average_CPI'] = df[['Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun', 'Jul', 'Aug',␣
 ↪'Sep', 'Oct', 'Nov', 'Dec']].mean(axis=1)
plt.figure(figsize=(10, 6))
plt.plot(df['Year'], df['Average_CPI'], marker='o')
plt.xlabel('Year')
plt.ylabel('Average CPI')
plt.title('Average CPI Over the Years')
plt.grid(True)
plt.show()

# Plot CPI trends for each month
plt.figure(figsize=(12, 8))
for month in df.columns[1:13]:
    plt.plot(df['Year'], df[month], label=month)
plt.xlabel('Year')
plt.ylabel('CPI')
plt.title('Monthly CPI Trends Over the Years')
plt.legend()
plt.grid(True)
plt.show()

# Compare HALF1 and HALF2
plt.figure(figsize=(10, 6))
plt.plot(df['Year'], df['HALF1'], marker='o', label='HALF1')
plt.plot(df['Year'], df['HALF2'], marker='o', label='HALF2', linestyle='--')
plt.xlabel('Year')
plt.ylabel('CPI')
plt.title('HALF1 vs HALF2 CPI Comparison')
plt.legend()
plt.grid(True)
plt.show()

summary_stats
```




Monthly CPI Trends Over the Years
Jan
Feb
Mar
Apr
May
Jun
Jul
Aug
Sep
Oct
Nov
Dec
CPI
1
2
3
4
5
6
Year
2000
2005
2010
2015
2020
2025

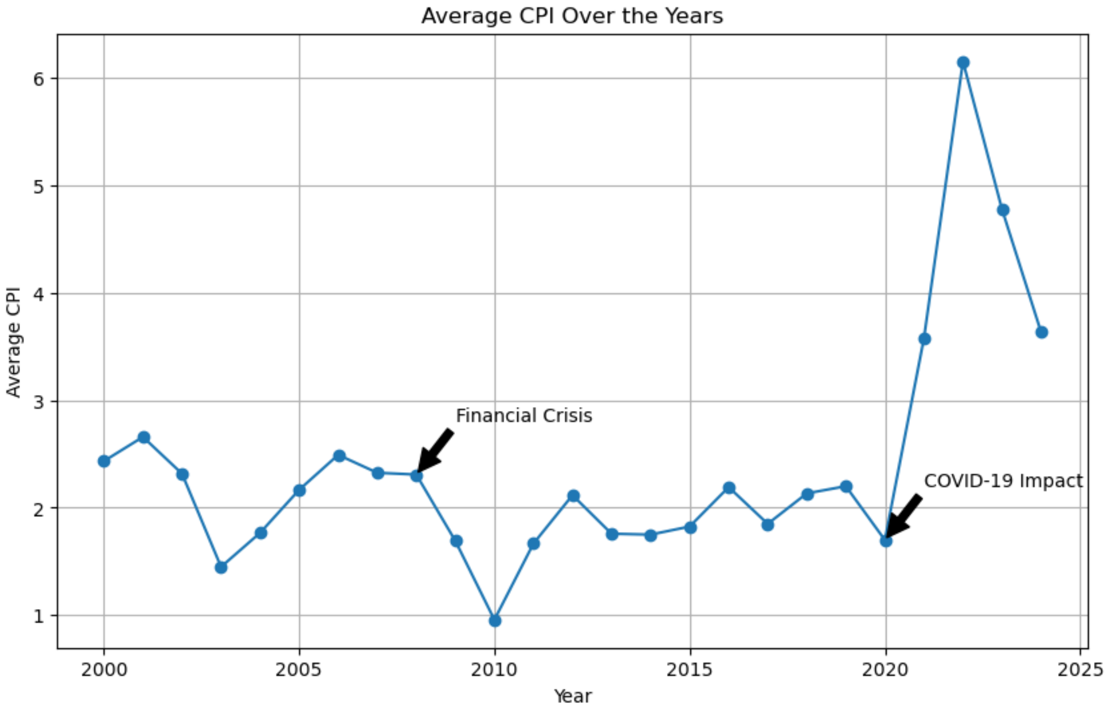

HALF1 vs HALF2 CPI Comparison
HALF1
HALF2
CPI
1
2
3
4
5
6
Year
2000
2005
2010
2015
2020
2025

[18]:

```
              Year        Jan        Feb        Mar        Apr        May  \
count    25.000000  25.000000  25.000000  25.000000  25.000000  25.000000
mean   2012.000000   2.368000   2.376000   2.392000   2.376000   2.380000
std       7.359801   1.188388   1.227219   1.226757   1.182962   1.160101
min    2000.000000   1.000000   1.100000   1.100000   0.900000   0.900000
25%    2006.000000   1.700000   1.700000   1.800000   1.800000   1.700000
50%    2012.000000   2.200000   2.200000   2.100000   2.100000   2.200000
75%    2018.000000   2.500000   2.400000   2.400000   2.300000   2.400000
max    2024.000000   6.000000   6.400000   6.500000   6.200000   6.000000

             Jun        Jul        Aug        Sep        Oct        Nov  \
count  25.000000  24.000000  24.000000  24.000000  24.000000  24.000000
mean    2.392000   2.370833   2.358333   2.358333   2.350000   2.354167
std     1.143431   1.112959   1.130954   1.165786   1.152313   1.139405
min     0.900000   0.900000   0.900000   0.800000   0.600000   0.800000
25%     1.700000   1.775000   1.700000   1.700000   1.800000   1.700000
50%     2.200000   2.150000   2.100000   2.050000   2.100000   2.100000
75%     2.500000   2.500000   2.525000   2.525000   2.350000   2.375000
max     5.900000   5.900000   6.300000   6.600000   6.300000   6.000000

             Dec      HALF1      HALF2  Average_CPI
count  24.000000  25.000000  24.000000    25.000000
mean    2.366667   2.380000   2.358333     2.395667
std     1.157083   1.144552   1.122465     1.106227
min     0.800000   1.100000   0.800000     0.958333
25%     1.800000   1.800000   1.700000     1.758333
50%     2.200000   2.200000   2.150000     2.166667
75%     2.450000   2.400000   2.350000     2.433333
max     5.700000   6.200000   6.100000     6.150000
```

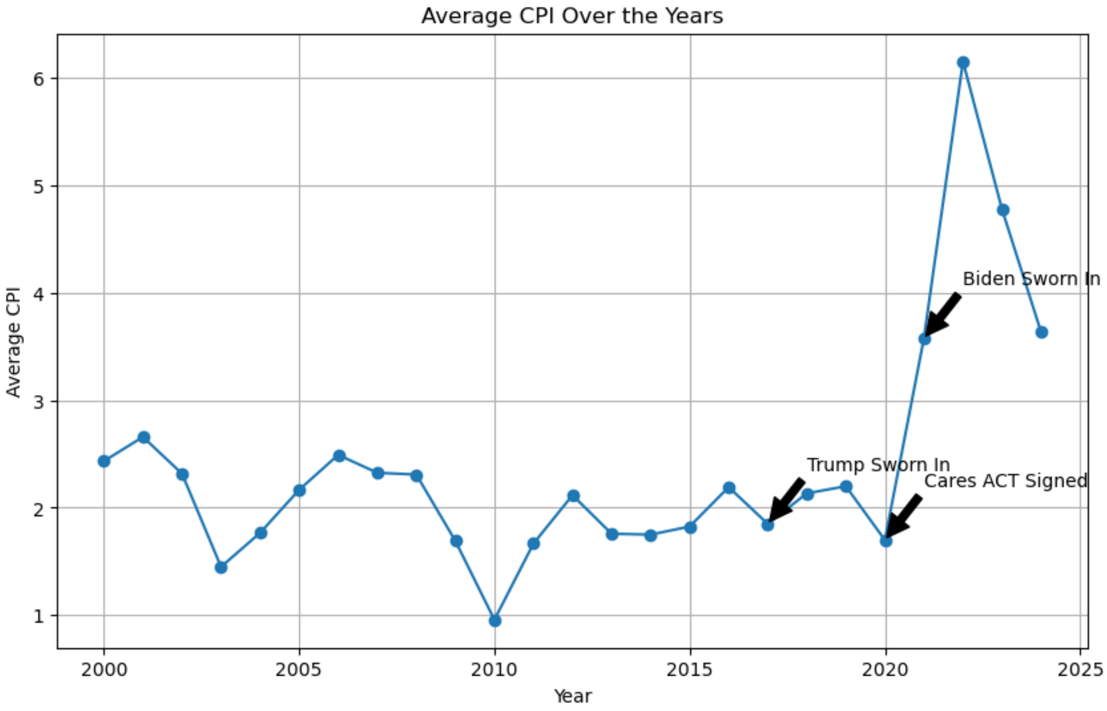
[19]:

```python
plt.figure(figsize=(10, 6))
plt.plot(df['Year'], df['Average_CPI'], marker='o')
plt.xlabel('Year')
plt.ylabel('Average CPI')
plt.title('Average CPI Over the Years')
plt.grid(True)

# Annotate multiple years
for year, label in [(2008, 'Financial Crisis'), (2020, 'COVID-19 Impact')]:
    cpi_value = df[df['Year'] == year]['Average_CPI'].values[0]
    plt.annotate(label, xy=(year, cpi_value), xytext=(year + 1, cpi_value + 0.
  ↪5),
                 arrowprops=dict(facecolor='black', shrink=0.05))

plt.show()
```



[24]:

```
plt.figure(figsize=(10, 6))
plt.plot(df['Year'], df['Average_CPI'], marker='o')
plt.xlabel('Year')
plt.ylabel('Average CPI')
plt.title('Average CPI Over the Years')
plt.grid(True)

# Annotate multiple years
for year, label in [(2017, 'Trump Sworn In'), (2020, 'Cares ACT Signed'),(2021,␣
 ↪'Biden Sworn In')]:
    cpi_value = df[df['Year'] == year]['Average_CPI'].values[0]
    plt.annotate(label, xy=(year, cpi_value), xytext=(year + 1, cpi_value + 0.
 ↪5),
                 arrowprops=dict(facecolor='black', shrink=0.05))

plt.show()
```



```
[ ]: file_pathPPI = r"C:\Users\lamarwells\OneDrive - Rasmussen,␣
      ↪Inc\AnacondaProjects\PPIManufactring.csv"
```

```
[ ]: ppi = pd.read_csv(file_pathPPI)
```

```
[ ]: ppi.head(5)
```

```
[ ]: print(ppi.columns)
     print(df.columns)
```

```
[ ]:
```