Step 1: Import Required Libraries

In [1]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split, GridSearchCV
from sklearn.preprocessing import StandardScaler
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import mean_squared_error, r2_score
import joblib
```

Step 2: Load Dataset

In [2]:
```
excel_file = 'SupplyChainEmissionFactorsforUSIndustriesCommodities.xlsx'
years = range(2010, 2017)
```

In [3]:
```
years[2]
```

Out[3]:
```
2012
```

In [4]:
```
df_1 = pd.read_excel(excel_file, sheet_name=f'{years[0]}_Detail_Commodity')
df_1.head()
```

Out[4]:

	Commodity Code	Commodity Name	Substance	Unit	Supply Chain Emission Factors without Margins	Margins of Supply Chain Emission Factors	Supply Chain Emission Factors with Margins	Unnamed 7
0	1111A0	Fresh soybeans, canola, flaxseeds, and other o...	carbon dioxide	kg/2018 USD, purchaser price	0.398	0.073	0.470	NaN
1	1111A0	Fresh soybeans, canola, flaxseeds, and other o...	methane	kg/2018 USD, purchaser price	0.001	0.001	0.002	NaN
2	1111A0	Fresh soybeans, canola, flaxseeds, and other o...	nitrous oxide	kg/2018 USD, purchaser price	0.002	0.000	0.002	NaN
3	1111A0	Fresh soybeans, canola, flaxseeds, and other o...	other GHGs	kg CO2e/2018 USD, purchaser price	0.002	0.000	0.002	NaN
4	1111B0	Fresh wheat, corn, rice, and other grains	carbon dioxide	kg/2018 USD, purchaser price	0.659	0.081	0.740	NaN

In [5]:
```
df_2 = pd.read_excel(excel_file, sheet_name=f'{years[0]}_Detail_Industry')
df_2.head()
```

Out[5]:

	Industry Code	Industry Name	Substance	Unit	Supply Chain Emission Factors without Margins	Margins of Supply Chain Emission Factors	Supply Chain Emission Factors with Margins	Unnamed: 7	Reli
0	1111A0	Oilseed farming	carbon dioxide	kg/2018 USD, purchaser price	0.414	0.073	0.487	NaN	
1	1111A0	Oilseed farming	methane	kg/2018 USD, purchaser price	0.001	0.001	0.002	NaN	
2	1111A0	Oilseed farming	nitrous oxide	kg/2018 USD, purchaser price	0.002	0.000	0.002	NaN	
3	1111A0	Oilseed farming	other GHGs	kg CO2e/2018 USD, purchaser price	0.002	0.000	0.002	NaN	
4	1111B0	Grain farming	carbon dioxide	kg/2018 USD, purchaser price	0.680	0.082	0.762	NaN	

In [13]:

```
all_data = []
for year in years:
    try:
        df_com = pd.read_excel(excel_file, sheet_name=f'{year}_Detail_Commo
        df_ind = pd.read_excel(excel_file, sheet_name=f'{year}_Detail_Indus

        df_com['Source'] = 'Commodity'
        df_ind['Source'] = 'Industry'
        df_com['Year'] = df_ind['Year'] = year

        df_com.columns = df_com.columns.str.strip()
        df_ind.columns = df_ind.columns.str.strip()

        df_com.rename(columns={
            'Commodity Code': 'Code',
            'Commodity Name': 'Name'
        }, inplace=True)

        df_ind.rename(columns={
            'Industry Code': 'Code',
            'Industry Name': 'Name'
        }, inplace=True)

        all_data.append(pd.concat([df_com, df_ind], ignore_index=True))

    except Exception as e:
        print(f"Error processing year {year}: {e}")
```

In [14]:
```
all_data[3]
```

Out[14]:

	Code	Name	Substance	Unit	Supply Chain Emission Factors without Margins	Margins of Supply Chain Emission Factors	Supply Chain Emission Factors with Margins	Unnamed: 7
0	1111A0	Fresh soybeans, canola, flaxseeds, and other o...	carbon dioxide	kg/2018 USD, purchaser price	0.373	0.072	0.444	NaN
1	1111A0	Fresh soybeans, canola, flaxseeds, and other o...	methane	kg/2018 USD, purchaser price	0.001	0.001	0.002	NaN
2	1111A0	Fresh soybeans, canola, flaxseeds, and other o...	nitrous oxide	kg/2018 USD, purchaser price	0.002	0.000	0.002	NaN
3	1111A0	Fresh soybeans, canola, flaxseeds, and other o...	other GHGs	kg CO2e/2018 USD, purchaser price	0.002	0.000	0.002	NaN
4	1111B0	Fresh wheat, corn, rice, and other grains	carbon dioxide	kg/2018 USD, purchaser price	0.722	0.079	0.801	NaN
...	...	...	...	...	...	...	...	...
3151	813B00	Civic, social, professional, and similar organ...	other GHGs	kg CO2e/2018 USD, purchaser price	0.008	0.000	0.008	NaN
3152	814000	Private households	carbon dioxide	kg/2018 USD, purchaser price	0.000	0.000	0.000	NaN
3153	814000	Private households	methane	kg/2018 USD, purchaser price	0.000	0.000	0.000	NaN
3154	814000	Private households	nitrous oxide	kg/2018 USD, purchaser price	0.000	0.000	0.000	NaN
3155	814000	Private households	other GHGs	kg CO2e/2018 USD, purchaser price	0.000	0.000	0.000	NaN

3156 rows × 15 columns

In [15]:
```
len(all_data)
```

Out[15]:
```
7
```

In [16]:
```
df = pd.concat(all_data, ignore_index=True)
df.head(10)
```

Out[16]:

	Code	Name	Substance	Unit	Supply Chain Emission Factors without Margins	Margins of Supply Chain Emission Factors	Supply Chain Emission Factors with Margins	Unnamed: 7	Rel
0	1111A0	Fresh soybeans, canola, flaxseeds, and other o...	carbon dioxide	kg/2018 USD, purchaser price	0.398	0.073	0.470	NaN	
1	1111A0	Fresh soybeans, canola, flaxseeds, and other o...	methane	kg/2018 USD, purchaser price	0.001	0.001	0.002	NaN	
2	1111A0	Fresh soybeans, canola, flaxseeds, and other o...	nitrous oxide	kg/2018 USD, purchaser price	0.002	0.000	0.002	NaN	
3	1111A0	Fresh soybeans, canola, flaxseeds, and other o...	other GHGs	kg CO2e/2018 USD, purchaser price	0.002	0.000	0.002	NaN	
4	1111B0	Fresh wheat, corn, rice, and other grains	carbon dioxide	kg/2018 USD, purchaser price	0.659	0.081	0.740	NaN	
5	1111B0	Fresh wheat, corn, rice, and other grains	methane	kg/2018 USD, purchaser price	0.008	0.001	0.009	NaN	
6	1111B0	Fresh wheat, corn, rice, and other grains	nitrous oxide	kg/2018 USD, purchaser price	0.004	0.000	0.004	NaN	
7	1111B0	Fresh wheat, corn, rice, and other grains	other GHGs	kg CO2e/2018 USD, purchaser price	0.004	0.000	0.004	NaN	
8	111200	Fresh vegetables, melons, and potatoes	carbon dioxide	kg/2018 USD, purchaser price	0.183	0.132	0.315	NaN	
9	111200	Fresh vegetables, melons, and potatoes	methane	kg/2018 USD, purchaser price	0.001	0.001	0.002	NaN	

In [17]:
```
len(df)
```

Out[17]:
```
22092
```

Step 3: Data Preprocessing

In [18]:
```
df.columns
```

Out[18]:
```
Index(['Code', 'Name', 'Substance', 'Unit',
       'Supply Chain Emission Factors without Margins',
       'Margins of Supply Chain Emission Factors',
       'Supply Chain Emission Factors with Margins', 'Unnamed: 7',
       'DQ ReliabilityScore of Factors without Margins',
       'DQ TemporalCorrelation of Factors without Margins',
       'DQ GeographicalCorrelation of Factors without Margins',
       'DQ TechnologicalCorrelation of Factors without Margins',
       'DQ DataCollection of Factors without Margins', 'Source', 'Year'],
      dtype='object')
```

In [19]:
```
df.isnull().sum()
```

Out[19]:
```
Code                                                    0
Name                                                    0
Substance                                               0
Unit                                                    0
Supply Chain Emission Factors without Margins           0
Margins of Supply Chain Emission Factors                0
Supply Chain Emission Factors with Margins              0
Unnamed: 7                                          22092
DQ ReliabilityScore of Factors without Margins          0
DQ TemporalCorrelation of Factors without Margins       0
DQ GeographicalCorrelation of Factors without Margins   0
DQ TechnologicalCorrelation of Factors without Margins  0
DQ DataCollection of Factors without Margins            0
Source                                                  0
Year                                                    0
dtype: int64
```

In [ ]:
```

```