You've uploaded a file named steel+industry+energy+consumption.zip. I'll start by examining its contents to identify the dataset and understand its structure. Then, I'll provide detailed steps for data filtering, outlier detection, and the requested interpolation and smoothing techniques, along with explanations and code.

Let me extract and inspect the contents of the uploaded file.

The uploaded ZIP file contains a dataset named Steel\_industry\_data.csv. I'll load this file to inspect its structure and contents before proceeding with the requested analyses.

The dataset contains 35,040 rows and 11 columns. Here's a summary of its columns:

1. **date**: Timestamp.
2. **Usage\_kWh**: Energy usage in kWh.
3. **Lagging\_Current\_Reactive.Power\_kVarh**: Lagging reactive power in kVarh.
4. **Leading\_Current\_Reactive\_Power\_kVarh**: Leading reactive power in kVarh.
5. **CO2(tCO2)**: CO₂ emissions in tons.
6. **Lagging\_Current\_Power\_Factor**: Lagging power factor.
7. **Leading\_Current\_Power\_Factor**: Leading power factor.
8. **NSM**: Number of seconds from midnight.
9. **WeekStatus**: Whether the day is a weekday or weekend.
10. **Day\_of\_week**: Day of the week.
11. **Load\_Type**: Type of load (e.g., Light\_Load).

**Approach**

We'll conduct the following steps:

1. **Data Filtering**: Identify and handle missing values and anomalies.
2. **Outlier Detection**:
   * Z-score method
   * Interquartile Range (IQR)
3. **Interpolation**:
   * Kriging
   * Spline interpolation
4. **Smoothing**:
   * Moving average
   * Exponential smoothing

Let's start with basic data filtering.

**Observations:**

1. **No Missing Values**: All columns are complete.
2. **Numerical Distribution**:
   * The Usage\_kWh column has values ranging from 0 to 157.18, with a mean of 27.39 and a high standard deviation, indicating potential outliers.
   * Other columns like Lagging\_Current\_Reactive.Power\_kVarh and Leading\_Current\_Reactive\_Power\_kVarh also show significant variability.

**Next Steps:**

We will identify and handle outliers using the **Z-score** and **IQR** methods for the Usage\_kWh column as an example, then extend it to others if needed.

**Step 1: Outlier Detection Using Z-Score**

The Z-score indicates how many standard deviations a data point is from the mean. Data points with Z-scores beyond a threshold (commonly ±3) are considered outliers. Let's calculate it.

It seems like I can’t do more advanced data analysis right now. Please try again later. If you want, I can provide you with Python code to run locally for the analysis, or we can revisit this once the issue is resolved. Let me know how you'd like to proceed!

import zipfile

import os

# Define the file path and extraction directory

zip\_file\_path = '/mnt/data/steel+industry+energy+consumption.zip'

extraction\_dir = '/mnt/data/steel\_industry\_energy\_consumption'

# Extract the zip file

with zipfile.ZipFile(zip\_file\_path, 'r') as zip\_ref:

zip\_ref.extractall(extraction\_dir)

# List the extracted files

os.listdir(extraction\_dir)

import pandas as pd

# Load the dataset

file\_path = os.path.join(extraction\_dir, 'Steel\_industry\_data.csv')

data = pd.read\_csv(file\_path)

# Display basic information about the dataset and the first few rows

data.info(), data.head()

# Check for missing values

missing\_values = data.isnull().sum()

# Identify numerical columns for further analysis

numerical\_cols = data.select\_dtypes(include=['float64', 'int64']).columns

# Basic statistics for numerical columns to understand data distribution

basic\_stats = data[numerical\_cols].describe()

missing\_values, basic\_stats

from scipy.stats import zscore

# Calculate Z-scores for the `Usage\_kWh` column

data['Z\_Score\_Usage\_kWh'] = zscore(data['Usage\_kWh'])

# Identify outliers where Z-score > 3 or < -3

z\_score\_outliers = data[(data['Z\_Score\_Usage\_kWh'] > 3) | (data['Z\_Score\_Usage\_kWh'] < -3)]

# Percentage of outliers

z\_score\_outlier\_percentage = len(z\_score\_outliers) / len(data) \* 100

z\_score\_outlier\_percentage, z\_score\_outliers[['date', 'Usage\_kWh', 'Z\_Score\_Usage\_kWh']].head()