Household Electricity Consumption Analysis

1. Introduction

Electricity consumption is a critical aspect of modern life, and understanding household energy usage patterns can provide valuable insights for optimizing energy efficiency, reducing costs, and improving sustainability. This project focuses on analyzing a **Household Electricity Consumption Dataset** using advanced data exploration, preprocessing techniques, and **Hidden Markov Models (HMMs)**. The dataset represents a multivariate time series with minute-level measurements of electricity consumption, capturing various features such as global active power, voltage, and sub-metering values.

The primary goal of this project is to explore and model electricity consumption behavior, with a focus on identifying patterns, detecting anomalies, and understanding the underlying dynamics of energy usage. The analysis is divided into three main phases:

1. **Data Exploration and Preparation:**
    - Handling missing data using linear interpolation.
    - Detecting anomalies using Z-scores.
    - Performing correlation analysis and time window analysis to identify trends and relationships between features.
2. **Preprocessing and Anomaly Detection:**
    - Applying feature scaling techniques such as normalization and standardization to prepare the data for modeling.
    - Smoothing time series data to compute weekly averages and anomaly scores.
    - Evaluating the impact of discretization on model performance.
3. **Hidden Markov Models (HMMs):**
    - Training HMMs to model electricity consumption patterns.
    - Determining the optimal number of states for the HMM using log-likelihood and Bayesian Information Criterion (BIC).
    - Understanding the implications of log-likelihood values for discrete and continuous variables.

This report integrates the findings from all three phases into a cohesive narrative, providing a comprehensive understanding of household electricity consumption behavior. By leveraging statistical and machine learning techniques, we aim to uncover actionable insights that can inform energy management strategies and contribute to a more sustainable future.

---

2. Data Exploration and Preparation

2.1 Handling Missing Data

Missing values in the dataset were handled using **linear interpolation**. This method estimated missing values based on adjacent non-missing data points. The following features were processed:

- Global_active_power
- Global_reactive_power
- Voltage
- Global_intensity
- Sub_metering_1
- Sub_metering_2
- Sub_metering_3

After interpolation, the dataset was checked for any remaining missing values, and none were found.

2.2 Anomaly Detection

Anomalies were detected using **Z-scores**, which measure how many standard deviations a data point is from the mean.

- **Z-score** is calculated by dividing the mean of values in feature by the standard deviation

$$zScore = mean / standardDeviation$$

A data point was considered an anomaly if its Z-score exceeded **3** in absolute value. The percentage of anomalies for each feature is as follows:

Feature	Percentage of Anomalies
Global_active_power	1.66%
Global_reactive_power	1.07%
Voltage	0.55%
Global_intensity	1.83%
Sub_metering_1	2.84%
Sub_metering_2	2.83%
Sub_metering_3	0.004%

Total percentage of anomalies in the dataset: 7.14%

Insights:

- Sub_metering_1 and Sub_metering_2 have the highest percentage of anomalies, indicating potential irregularities in these features.
- Sub_metering_3 has the lowest percentage of anomalies, suggesting it is relatively stable.
- The total percentage of anomalies in our dataset was **7.14%**, suggesting unusual behaviour. The percentage found is significantly greater than what is considered acceptable, and warrants further investigation into the causes of these outliers.

---

3. Correlation Analysis

3.1 Correlation Matrix

The correlation between each pair of features was computed using **Pearson's correlation coefficient**. The results are summarized in the correlation matrix below:

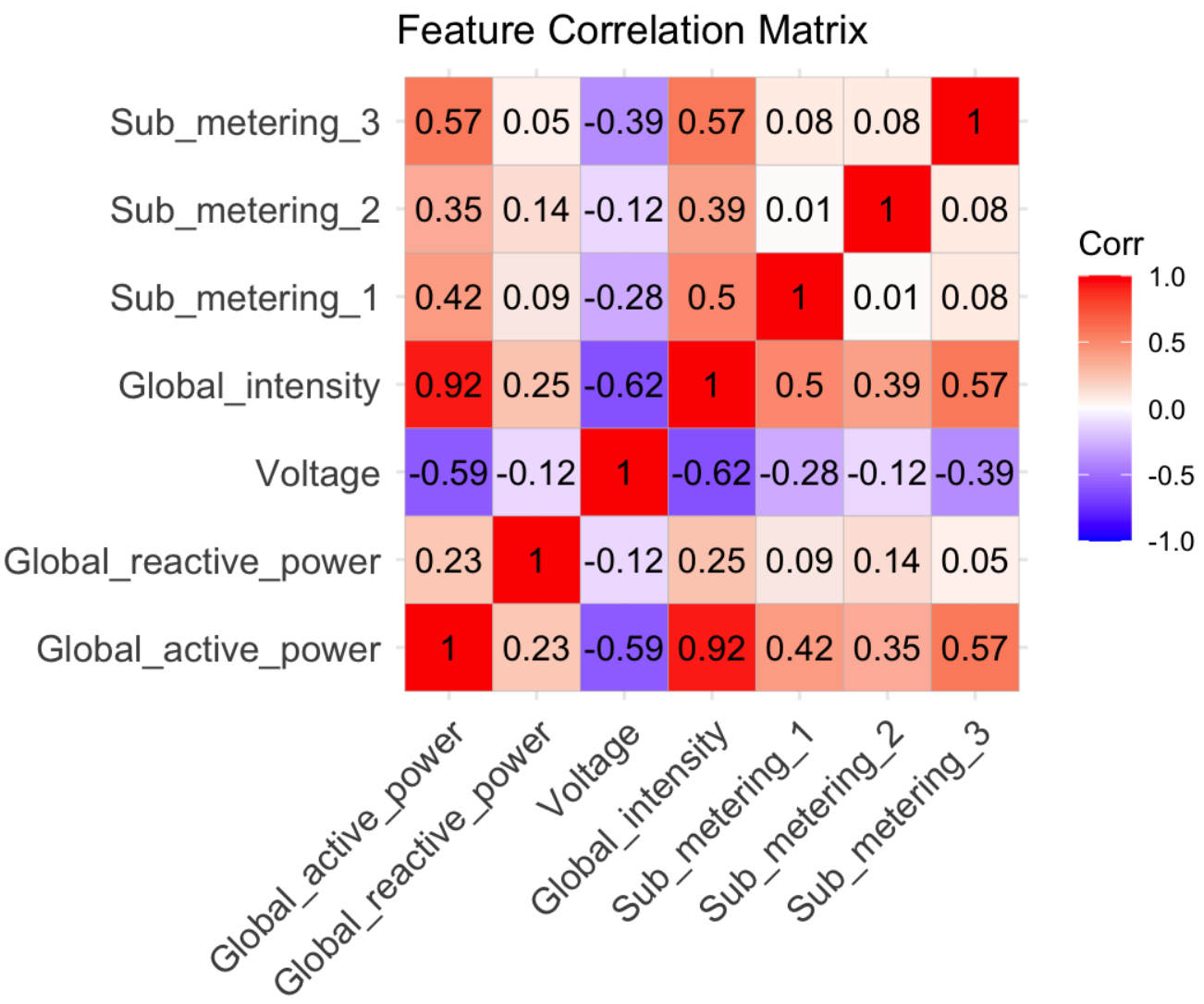
	Global_active_power	**Global_reactive_power**	**Voltage**	**Global_intensity**	**Sub_metering_1**	**Sub_metering_2**	**Sub_metering_3**
Global_active_power	1.000	0.226	-0.588	0.919	0.421	0.351	0.570
Global_reactive_power	0.226	1.000	-0.123	0.253	0.090	0.144	0.046
Voltage	-0.588	-0.123	1.000	-0.624	-0.283	-0.124	-0.393
Global_intensity	0.919	0.253	-0.624	1.000	0.498	0.385	0.567
Sub_metering_1	0.421	0.090	-0.283	0.498	1.000	0.005	0.079
Sub_metering_2	0.351	0.144	-0.124	0.385	0.005	1.000	0.075
Sub_metering_3	0.570	0.046	-0.393	0.567	0.079	0.075	1.000

3.2 Correlation Heatmap

The correlation matrix was visualized using a heatmap, with color-coding to indicate the strength and direction of correlations.

Insights:

- Global_active_power and Global_intensity are highly correlated (**0.919**), indicating a strong linear relationship.
- Voltage shows a moderate negative correlation with Global_active_power (**-0.588**) and Global_intensity (**-0.624**).
- Sub_metering_1 and Sub_metering_2 have almost no correlation (**0.005**), suggesting they measure independent aspects of energy consumption.


Feature Correlation Matrix
Sub_metering_3
Sub_metering_2
Sub_metering_1
Global_intensity
Voltage
Global_reactive_power
Global_active_power
0.57 0.05 -0.39 0.57 0.08 0.08 1
0.35 0.14 -0.12 0.39 0.01 1 0.08
0.42 0.09 -0.28 0.5 1 0.01 0.08
0.92 0.25 -0.62 1 0.5 0.39 0.57
-0.59 -0.12 1 -0.62 -0.28 -0.12 -0.39
0.23 1 -0.12 0.25 0.09 0.14 0.05
1 0.23 -0.59 0.92 0.42 0.35 0.57
Global_active_power
Global_reactive_power
Voltage
Global_intensity
Sub_metering_1
Sub_metering_2
Sub_metering_3
Corr
1.0
0.5
0.0
-0.5
-1.0

4. Time Window Analysis

4.1 Data Extraction

The dataset was filtered to extract data for **Week 6** (February 5, 2007, to February 11, 2007). The data was further divided into four time windows:

1. **Weekday Daytime**: 7:30 AM to 5:00 PM
2. **Weekday Nighttime**: 11:00 PM to 6:00 AM
3. **Weekend Daytime**: 7:30 AM to 5:00 PM
4. **Weekend Nighttime**: 11:00 PM to 6:00 AM

4.2 Average Global Intensity

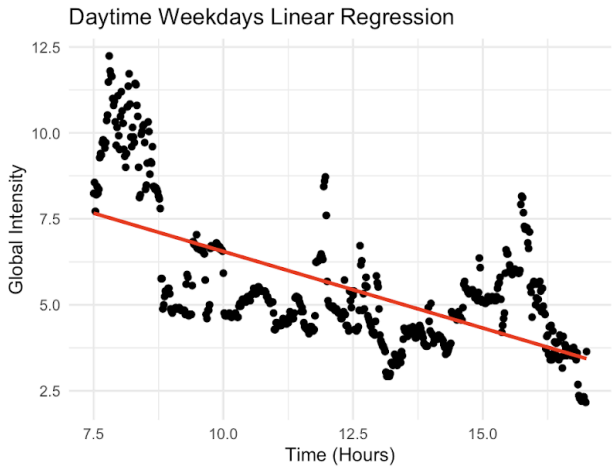
The average Global_intensity was computed for each timestamp in the four time windows. The results were used to perform **linear regression** and **polynomial regression** (degree 2) to model the behavior of Global_intensity.

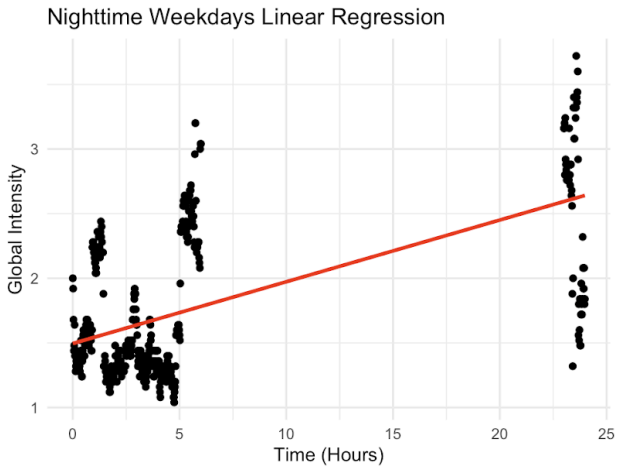
4.3 Regression Analysis

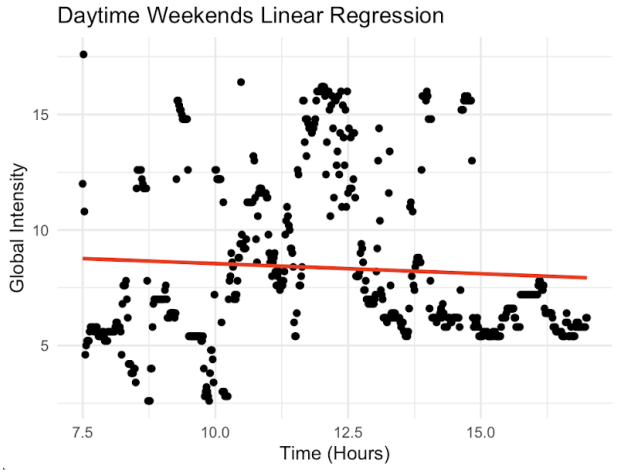
The regression models were fitted for each time window, and the results were visualized using scatter plots with regression lines (linear) and curves (polynomial). Key observations include:

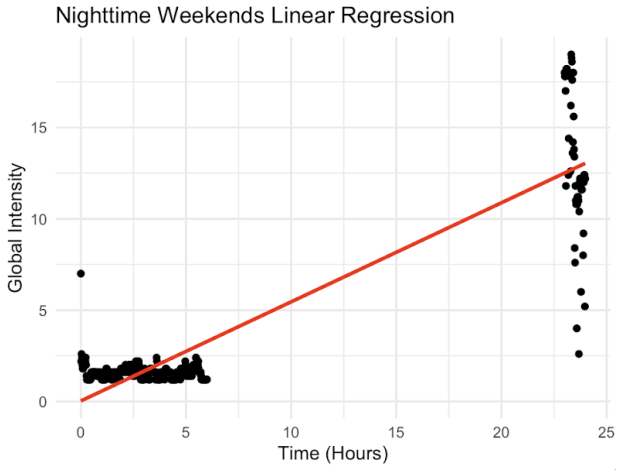
- **Daytime vs. Nighttime Trends**:
    - **Daytime**: Both weekdays and weekends show a **negative correlation** between TimeIndex and Global_intensity. This aligns with reduced reliance on artificial lighting and appliances during daylight hours.
    - **Nighttime**: Both weekdays and weekends show a **positive correlation**, reflecting increased electricity consumption due to lighting, heating, and appliance usage after dark.

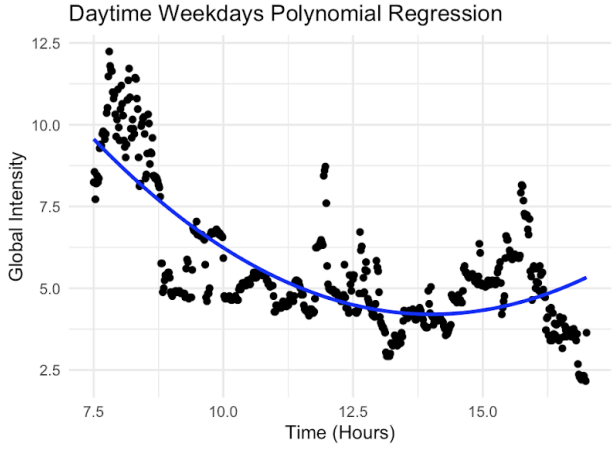
- **Weekday vs. Weekend Differences**:
    - **Weekday Daytime**: Lower consumption compared to weekends, likely because occupants are away from home (e.g., at work or school).
    - **Weekend Daytime**: Higher consumption, as occupants are more likely to be at home using appliances.
    - **Weekday/Weekend Nighttime**: Similar trends, as nighttime routines (e.g., sleeping, limited appliance use) are consistent regardless of weekday/weekend.

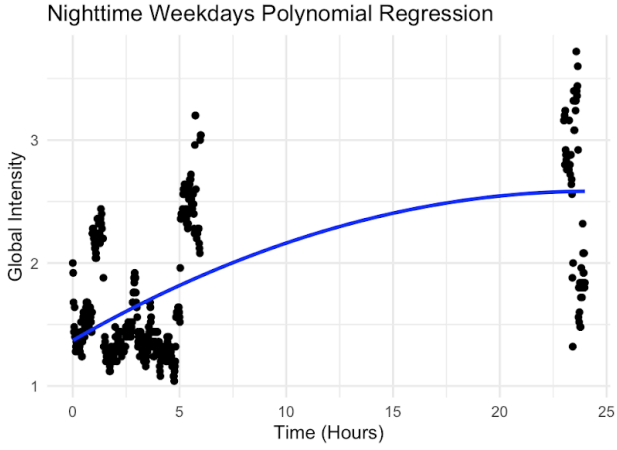

Daytime Weekdays Linear Regression
Global Intensity
Time (Hours)

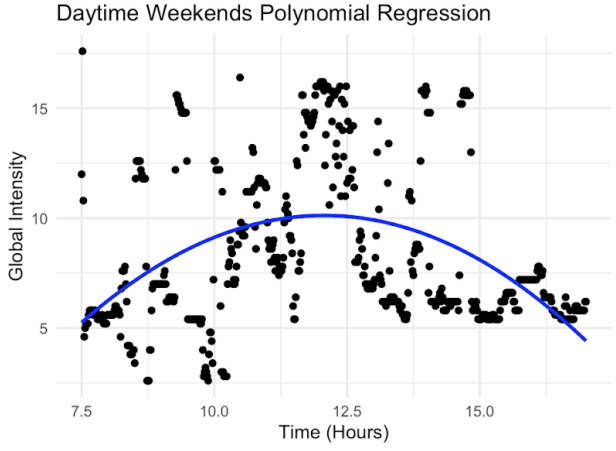

Nighttime Weekdays Linear Regression
Global Intensity
Time (Hours)

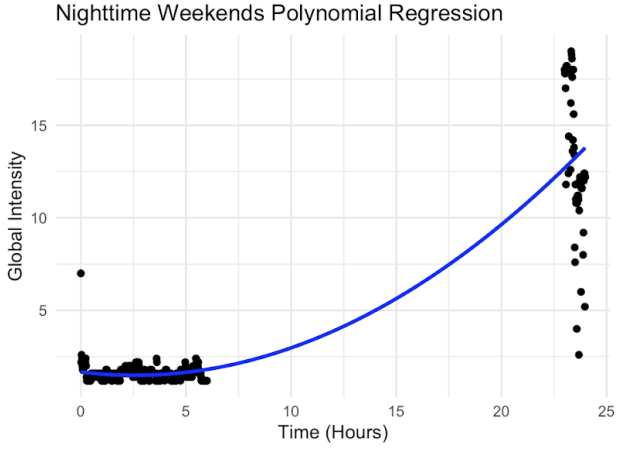

Daytime Weekends Linear Regression
Global Intensity
Time (Hours)


Nighttime Weekends Linear Regression
Global Intensity
Time (Hours)


Daytime Weekdays Polynomial Regression
Global Intensity
Time (Hours)


Nighttime Weekdays Polynomial Regression
Global Intensity
Time (Hours)


Daytime Weekends Polynomial Regression
Global Intensity
Time (Hours)


Nighttime Weekends Polynomial Regression
Global Intensity
Time (Hours)

5. Feature scaling techniques (normalization and standardization)

Feature scaling is a crucial data preprocessing technique used to transform data into a format that is more suitable for analysis, particularly in machine learning. It ensures that features within a dataset are on comparable scales, enabling models to process the data effectively. This process reduces the negative impacts of anomalies and ensures that algorithms can interpret the data appropriately. Without feature scaling, algorithms may treat values inconsistently (e.g., interpreting 10 cm and 10 m as equivalent), which can lead to inaccurate conclusions and predictions. Below, two common feature scaling techniques—normalization and standardization—are explained in detail, along with their significance and applications.

5.1 Normalization (Min-Max Scaling)

Normalization, also known as min-max scaling, transforms features of a dataset to fit within a fixed range, typically [0, 1]. This is achieved using the following formula:

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

- **Applications:** Normalization is often used when the data has a Gaussian (normal) distribution or when the algorithm assumes bounded inputs.
    - **Gaussian Distribution:** This implies that the data follows a bell-shaped curve, symmetrical around its mean.
    - **Bounded Input:** This refers to values constrained within a specific range, ensuring they do not exceed certain limits.
- **Advantages and Limitations:**
    - Normalization is highly sensitive to outliers. Extreme values for or can significantly compress the range of the remaining data, making it difficult to distinguish between values.
    - Noise within the data is not inherently removed. For extreme outliers, relevant data may become compressed, amplifying the impact of noise. However, if noise is small relative to the data's range, it may be mitigated.

5.2 Standardization (Z-Score Normalization)

Standardization transforms features of a dataset so that they have a mean of 0 and a standard deviation of 1. This centers the data around 0 and scales it by the standard deviation, as defined by the formula:

$$X_{\text{std}} = \frac{X - \mu}{\sigma}$$

- μ : Mean of the feature.
- σ : Standard deviation of the feature.

- **Applications:** Standardization is useful when the data has a Gaussian distribution or when the analysis algorithm assumes a mean of 0, such as in linear regression.
- **Advantages and Limitations:**
    - Standardization is less sensitive to outliers compared to normalization, as it uses the mean and standard deviation of the data. Slight noise can be mitigated, but extreme outliers may still distort the results.
    - Unlike normalization, standardization does not fix the range of scaled values, leaving them unbounded. This can result in excessively large or small outputs, potentially causing unpredictable behavior in models.

5.3 Conclusion

Both normalization and standardization are essential tools in feature scaling, each with specific use cases and limitations. Selecting the appropriate technique depends on the data characteristics and the requirements of the analysis or machine learning algorithm being used.

---

6. Smoothing time series data and computing anomaly scores

Week	Smooth Global Intensity	Anomaly Score
1	5.8684	1.5232
2	6.0704	1.7252
3	6.2403	1.8951
4	6.0141	1.6689
5	6.5851	2.2399
6	5.8271	1.4819
7	5.9089	1.5637
8	4.9344	0.5892
9	2.0485	2.2967
10	5.5772	1.232
11	5.0489	0.7037
12	5.809	1.4638
13	5.4221	1.0769
14	4.7857	0.4405
15	2.1457	2.1995
16	3.587	0.7583
17	2.9594	1.3858
18	3.4611	0.8841
19	4.7471	0.4019
20	4.2113	0.134
21	4.2454	0.0998
22	4.0716	0.2736
23	4.0173	0.3279
24	4.2104	0.1348
25	4.6543	0.3091
26	3.4473	0.8979

Week	Smooth Global Intensity	Anomaly Score
27	3.5238	0.8214
28	3.5064	0.8388
29	3.2558	1.0894
30	2.9688	1.3764
31	2.7614	1.5838
32	1.0891	3.2561
33	0.9011	3.4441
34	0.8439	3.5013
35	2.195	2.1502
36	3.6924	0.6528
37	4.0571	0.2881
38	4.0203	0.3249
39	4.4862	0.141
40	4.7933	0.4481
41	4.359	0.0138
42	4.7628	0.4176
43	4.309	0.0362
44	2.5311	1.8141
45	5.2827	0.9375
46	5.6995	1.3543
47	5.0378	0.6926
48	6.0662	1.721
49	5.4317	1.0864
50	5.7399	1.3947
51	5.8527	1.5075
52	6.8854	2.5402

6.1 Most and Least Anomalous Weeks

From the table above:

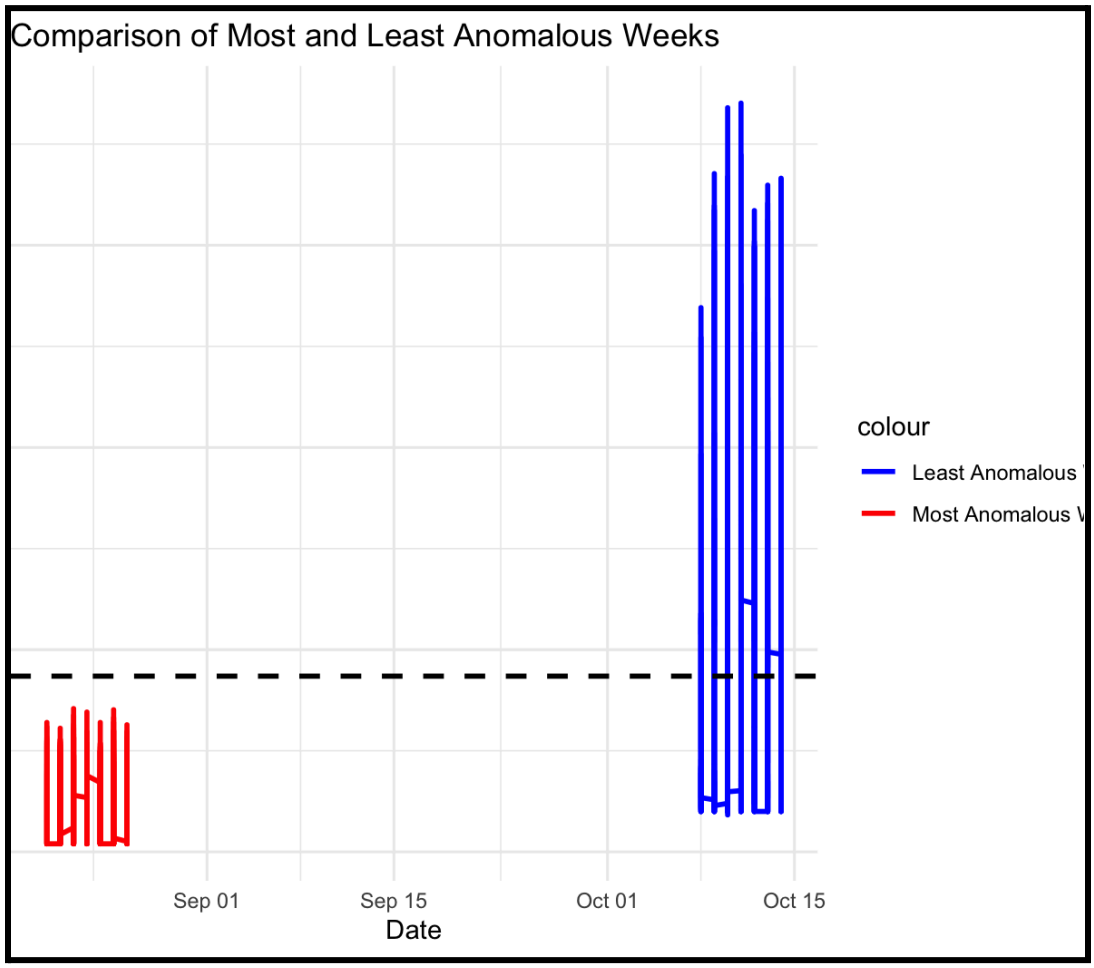
- The most anomalous week is **Week 34**, with an anomaly score of **3.5013**.
- The least anomalous week is **Week 41**, with an anomaly score of **0.0138**.

6.2 Rationale for Scoring Method

The anomaly score is computed as the deviation of each week's smoothened global intensity from the average smoothened week. This deviation is quantified using the **mean absolute deviation (MAD)**, which effectively captures how different each week is from the expected energy consumption pattern. MAD was chosen as it provides an intuitive and robust measure that is less sensitive to extreme outliers compared to standard deviation, ensuring that the ranking is reflective of overall trends rather than singular spikes.

6.3 Plot of Smoothened Versions

The graph below represents the smoothened versions of the most anomalous and least anomalous weeks compared to the average smoothened week.



7. Impact of discretization on model performance

To solve problem 1, there exists a procedure called the **Forward-backward procedure**. This procedure allows us to compute the probability of computing an observation sequence given a model in a feasible way. So in our case, we would like to know if after monitoring a certain system through time (with T observations), how much are our observations likely to happen.

Let's say that the system is analyzed every hour. At the end of the day, we end up with 24 observations. Then the next day, we want to check the system, so we compute the probability of obtaining that sequence of observations. To do so, at each timestep T, we compute the probability of getting to a certain state, by the sum of the previous forward path probabilities, by the probability of transitioning from the previous state to the new one, by the probability of getting that observation as:

$$\alpha_t(j) = \sum_{i=1}^{N} \alpha_{t-1}(i) a_{ij} b_j(o_t); \quad 1 \leq j \leq N, 1 < t \leq T$$

With N being the number of possible states.

Once we compute everything for the given sequence of observations, given the result, we can know if everything is behaving as expected if we got a high probability, or if there is something wrong with the system, given a lower probability. In the last case, the issue might be caused by a bug or it could mean that there really was an intrusion in the system, so we might need to analyze carefully that set of state observations in order to find which kind of problem is it.

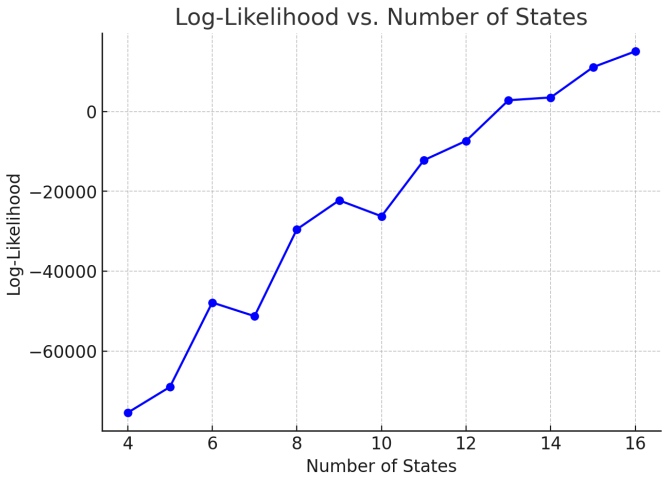
---

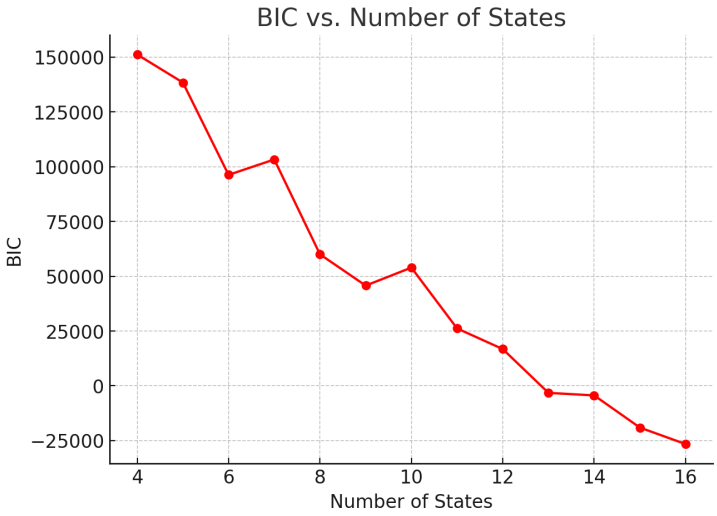
8. Determining the Best Number of States

After training our Hidden Markov Model (HMM) on the dataset the results we got are:

```
> results
   nstates     logLik        BIC
1        4 -75472.287 151227.402
2        5 -69008.960 138436.013
3        6 -47857.288  96292.527
4        7 -51277.224 103316.851
5        8 -29522.838  60017.126
6        9 -22266.897  45738.883
7       10 -26248.453  53960.229
8       11 -12217.256  26180.662
9       12  -7355.157  16763.883
10      13   2806.362  -3227.140
11      14   3565.406  -4388.619
12      15  11095.217 -19067.040
13      16  15064.765 -26600.340
```

We then analyzed the results by creating two graphs: one for log-likelihood and another for the Bayesian Information Criterion (BIC).





Comparing both plots, it is clear that log-likelihood and BIC have an inverse relationship. This means that as one increases, the other tends to decrease. To find the sweet spot for the number of states, we realized it should be something in between the two metrics. Based on our analysis, we settled on **10 states** as a balanced choice. This way, we can avoid skewed results that might arise from focusing too heavily on one variable over the other.

As we considered which states to keep, we focused on those that provide meaningful insights. It makes sense to eliminate or combine states that don't significantly contribute to our model. For example, states 13 and 14 showed similar results for both log-likelihood and BIC, so it might be wise to either drop one of them or merge them into a single state to simplify our model.

9. Understanding log-likelihood for discrete and continuous variables

When working with Hidden Markov Models (HMMs), the log-likelihood values can sometimes be positive, especially when dealing with continuous variables. This is less common with discrete variables, and the reason lies in how probabilities are calculated for each type of data.

For discrete variables, HMMs use Probability Mass Functions (PMFs) to define emission probabilities. PMFs assign probabilities to specific outcomes, and these probabilities always fall between 0 and 1. When we take the logarithm of these probabilities (as we do when computing log-likelihood), the result is always negative or zero. This is because the logarithm of any number between 0 and 1 is negative. As a result, the log-likelihood for discrete variables is typically negative.

On the other hand, for continuous variables, HMMs use Probability Density Functions (PDFs). Unlike PMFs, PDFs don't represent probabilities directly—they represent density. This means PDF values can be greater than 1, depending on the distribution. When we take the logarithm of these density values, the result can be positive if the density is high enough. This is why, when working with continuous variables, the log-likelihood can sometimes be positive—especially if the model assigns high density values to the observed data.

In short, the difference comes down to the nature of PMFs and PDFs. PMFs are bounded between 0 and 1, ensuring their logarithms are negative, while PDFs can exceed 1, allowing for positive logarithms and, consequently, positive log-likelihood values. This distinction is important to keep in mind when interpreting log-likelihood results in HMMs.

---

10. Evaluating the impact of discretization on HMM performance.

The log-likelihood values from the discretized HMM model were significantly lower (more negative) than the values from the original HMM model trained with continuous values. This indicates that the discretized data was actually a worse fit compared to the continuous model, which was to be expected. Continuous data contains more detailed information, allowing the model to capture more nuances in patterns and variations throughout the data.

Discretization can assist in capturing underlying patterns by simplifying data in an attempt to make analysis more efficient and manageable. Although this method of simplification can be extremely useful, it may have significant downsides in some situations. We will discuss in detail how discretization impacts our model's performance.

Reduces Overfitting
Overfitting is when the model is trained to memorize too many small details and is too complex. This can result in the model picking up on too many unnecessary and complex patterns showing a high variance in data, and ultimately failure to properly generalize the data. Discretization attempts to prevent this by grouping continuous values in datasets to be more easily interpreted.

Possible Loss/Distortion of Information
Because continuous values are grouped, this can lead to loss of detailed information that could possibly have been useful for accurate predictions. Distribution of the original data can also be distorted. Although such difficulties can be somewhat avoided by creating intuitive groups that are not too wide.

Reduces Computational Power Required
In cases of large datasets, models may require a large amount of computing power and memory allocation to analyze values. Because discretization reduces the complexity of datasets by grouping continuous values, models will require less time and computational power to complete analysis.

In conclusion, while discretizing data may result in a more negative log-likelihood—suggesting a worse fit—the benefits of simplification, such as reducing overfitting and lowering computational demands, might outweigh the drawbacks in certain situations. It's all about finding the right balance based on the specific context of our analysis.

---

11. Conclusion

This project has provided a comprehensive analysis of household electricity consumption using advanced data exploration, preprocessing techniques, and Hidden Markov Models (HMMs). By leveraging statistical and machine learning methods, we were able to uncover meaningful patterns, detect anomalies, and model the underlying dynamics of energy usage. The key findings and insights from each phase of the project are summarized below:

1. **Data Exploration and Preparation:**
    - Missing data was effectively handled using linear interpolation, ensuring the dataset's completeness.
    - Anomalies were detected using Z-scores, with **Sub_metering_1** and **Sub_metering_2** showing the highest anomaly rates, indicating potential irregularities in these features.
    - Correlation analysis revealed strong relationships between features, such as the high positive correlation (0.919) between **Global_active_power** and **Global_intensity**, and moderate negative correlations between **Voltage** and other features.
2. **Preprocessing and Anomaly Detection:**
    - Feature scaling techniques, including normalization and standardization, were applied to prepare the data for modeling.
    - Smoothing time series data allowed us to compute weekly averages and anomaly scores, with **Week 34** identified as the most anomalous week and **Week 41** as the least anomalous.
    - Discretization was evaluated as a preprocessing step, and while it simplified the data and reduced computational demands, it resulted in a worse fit compared to the continuous model, highlighting the trade-off between simplicity and information loss.
3. **Hidden Markov Models (HMMs):**
    - The optimal number of states for the HMM was determined to be **10**, balancing model complexity and goodness-of-fit.
    - Log-likelihood values were analyzed for both discrete and continuous variables, with continuous variables occasionally yielding positive log-likelihood values due to the nature of probability density functions (PDFs).
    - The impact of discretization on HMM performance was evaluated, emphasizing the importance of retaining detailed information for accurate modeling.

Key Takeaways

- **Anomaly Detection:** The high anomaly rate (7.14%) in the dataset suggests significant unusual behavior, particularly in the sub-metering features, warranting further investigation.
- **Feature Relationships:** The strong correlation between **Global_active_power** and **Global_intensity** highlights their interdependence, while the lack of correlation between **Sub_metering_1** and **Sub_metering_2** suggests they measure independent aspects of energy consumption.
- **Modeling Insights:** HMMs proved effective in capturing electricity consumption patterns, with the continuous model outperforming the discretized version in terms of fit.

In conclusion, this project demonstrates the value of combining data exploration, preprocessing, and advanced modeling techniques to analyze complex datasets like household electricity consumption. The insights gained can inform energy management strategies, promote sustainability, and contribute to a deeper understanding of energy usage patterns.
