ANOMALY DETECTION FOR ELECTRIC ENERGY CONSUMPTION USING PCA AND HMM

CMPT 318 D100 Fall 2024
Group 8:

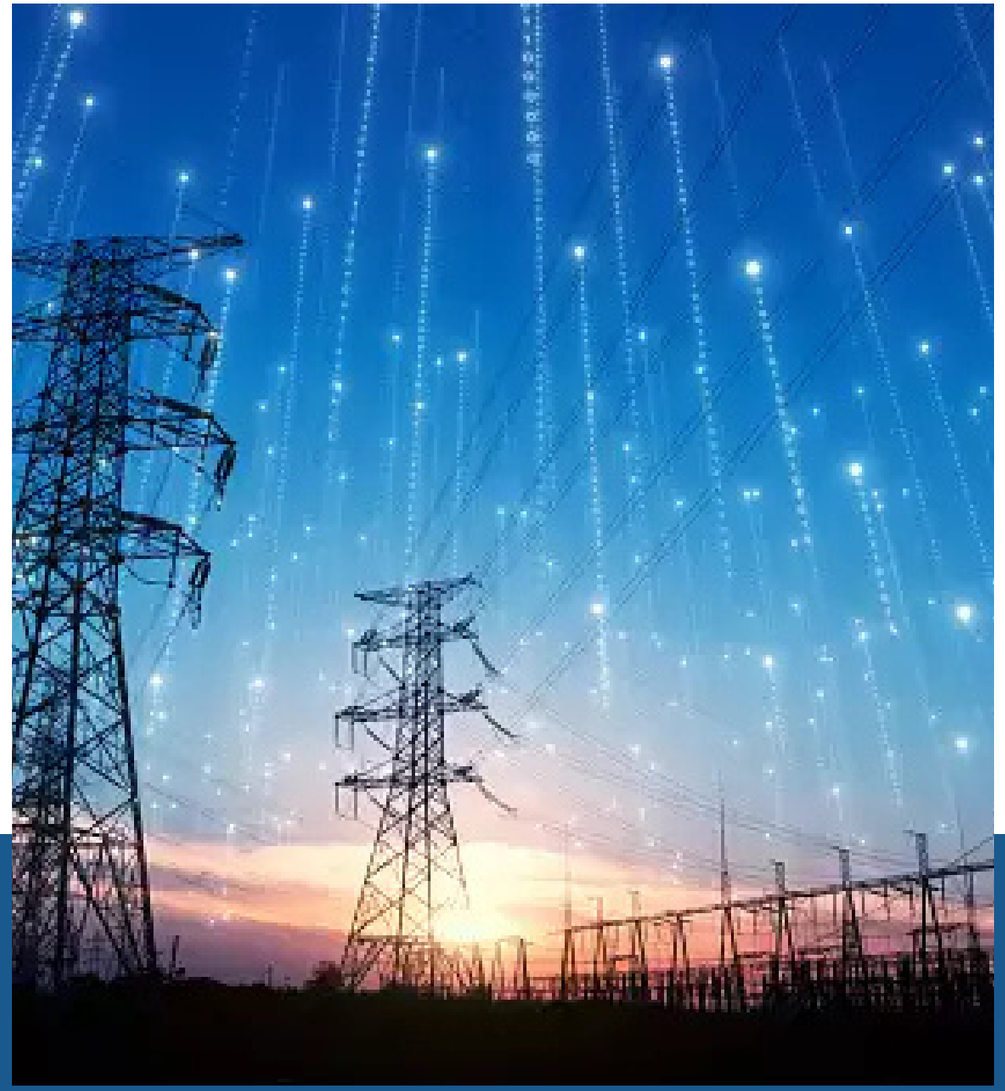
- Simon Yu, 301451144
- Calvin Weng, 301556001
- Tin Liang, 301565565
- Nazanin Pouria Mehr: 301442860

Table of contents

- Overview
- Problem Scope
- Feature Scaling
- Feature Engineering
- HMM Training
- Anomaly Detection
- Conclusion
- References

Overview

This project applies anomaly detection techniques to electric energy consumption data using PCA and HMM. It identifies patterns and anomalies in supervisory control systems, leveraging PCA for variable selection and HMM for state analysis. Key tasks include log-likelihood evaluations, BIC comparisons, and testing frameworks.



Problem Statement

Electric power grids rely on supervisory control systems for real-time management. Anomalies in this data can signal cyber threats or faults. This project explores detecting these anomalies using time-series data analysis, employing PCA for dimensionality reduction and HMM for temporal pattern modeling.

Feature Scaling

Scaling is crucial for ensuring all features contribute equally to model training, preventing bias from features with larger ranges. For HMMs, scaling improves the reliability of likelihood calculations and distance measurements, enhancing anomaly detection (Analytics Vidhya, 2020)..

Methods of Scaling:

- **Normalization:** Scales data to a fixed range, preserving the shape of the distribution; best for data without outliers.
- **Standardization:** Centers data to have a mean of 0 and a standard deviation of 1; ideal for handling outliers and varied feature ranges.

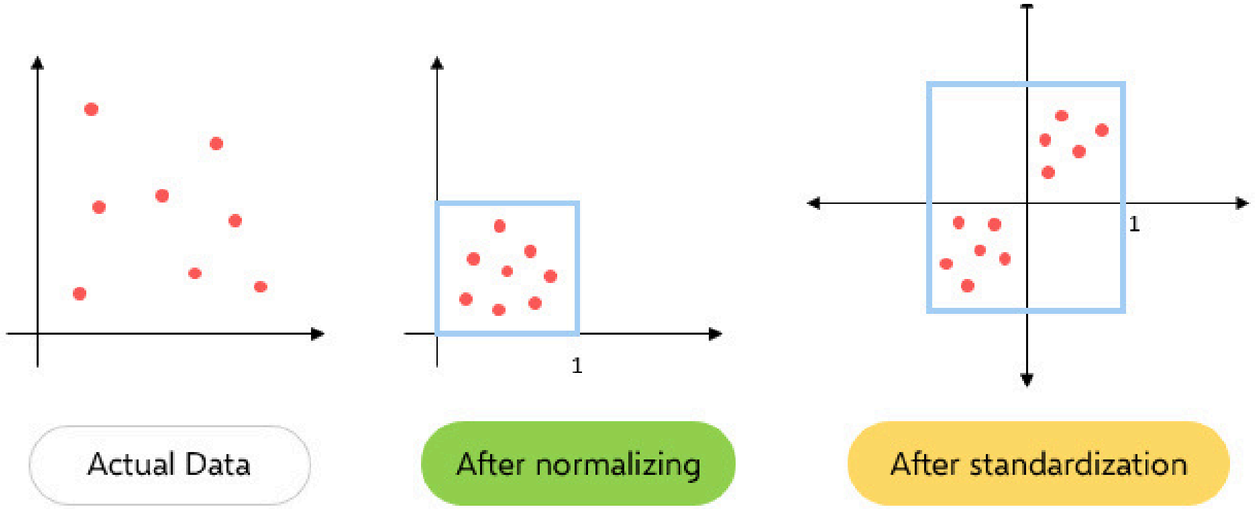
Standardisation	Normalisation
$x_{\text{stand}} = \frac{x - \text{mean}(x)}{\text{standard deviation } (x)}$	$x_{\text{norm}} = \frac{x - \min(x)}{\max(x) - \min(x)}$

Feature Scaling

For anomaly detection in electricity consumption data using Hidden Markov Models (HMMs), standardization is chosen as the preferred scaling method.

Why we chose standardization for HMMs:

- **Robustness to Outliers:** Handles extreme values like faulty meter readings without distorting data.
- **Alignment with Normal Distribution:** Matches HMM assumptions of data being centered around the mean, aiding in probabilistic modeling.



Feature Engineering

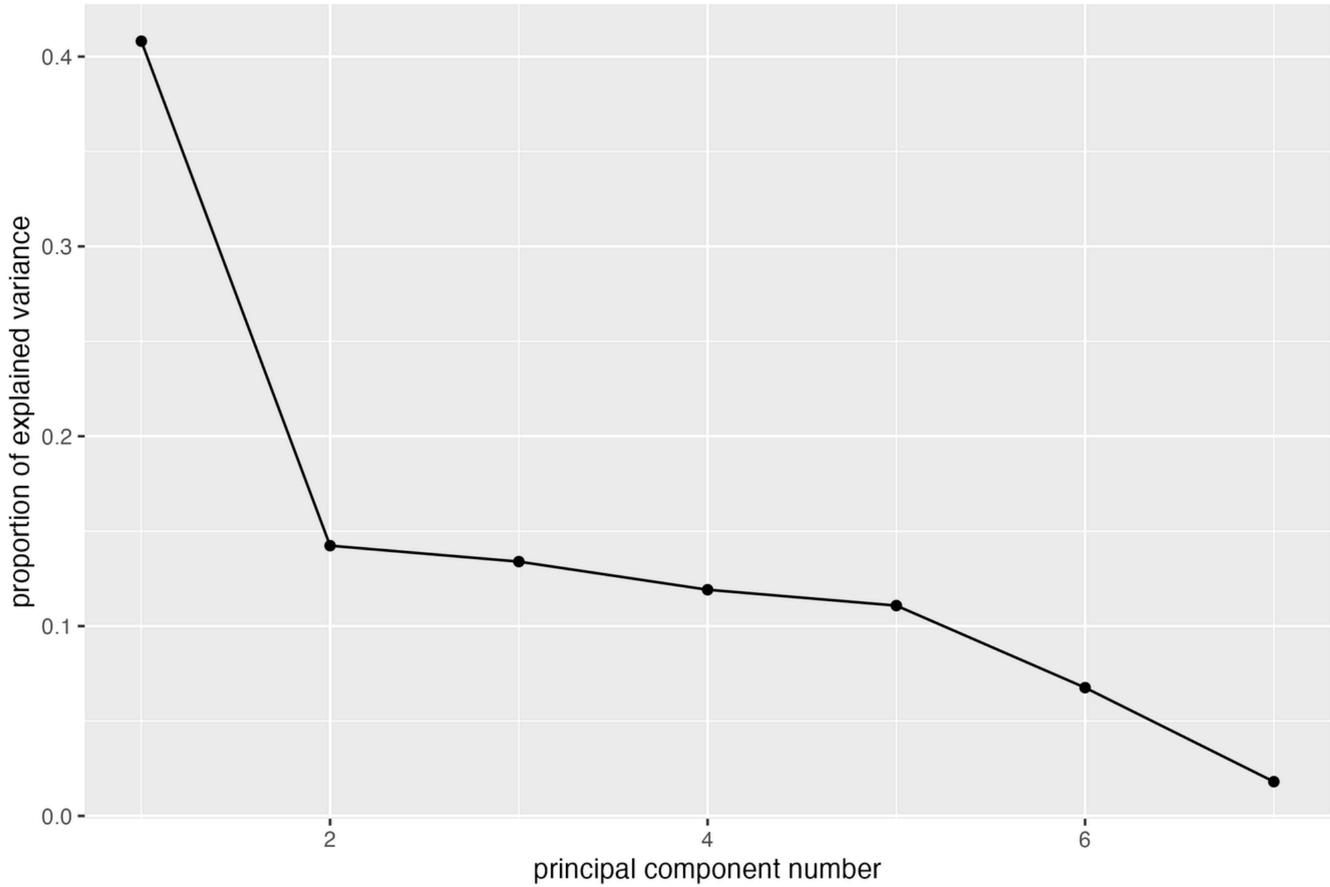
After preprocessing the dataset to ensure it was clean and standardized, we moved on to the Feature Engineering step, focusing on selecting a subset of variables for training multivariate Hidden Markov Models (HMMs) on normal electricity consumption data we used Principal Component Analysis (PCA) to select key variables for training multivariate HMMs on normal electricity consumption data.

	PC1	PC2	PC3	PC4	PC5	PC6	PC7
Global_active_power	-0.47939	-0.10351	0.20960	-0.68158	-0.48242	-0.13091	0.03315
Global_reactive_power	-0.18338	0.945602	0.164547	0.144018	-0.15577	0.010711	-0.00319
Voltage	0.393444	-0.05855	0.910659	-0.05774	0.091745	0.026718	0.002288
Global_intensity	-0.54619	-0.00499	0.147372	-0.13598	0.674889	0.413784	-0.18638
Sub_metering_1	-0.04200	0.017046	-0.00227	-0.03989	0.107182	0.195860	0.972881
Sub_metering_2	-0.07946	0.087788	0.010348	-0.11559	0.469075	-0.86021	0.111814
Sub_metering_3	-0.52472	-0.28925	0.279030	0.691320	-0.21814	-0.18021	0.071729

Feature Engineering

After preprocessing the dataset to ensure it was clean and standardized, we moved on to the Feature Engineering step, focusing on selecting a subset of variables for training multivariate Hidden Markov Models (HMMs) on normal electricity consumption datawe used Principal Component Analysis (PCA) to select key variables for training multivariate HMMs on normal electricity consumption data.

From the PCA results, we selected Global_reactive_power and Global_intensity as key variables, as they contributed significantly to the variance in the first few PCs. A scree plot was used to visualize and confirm the variance distribution, guiding the selection of important components.



HMM Training And Testing

The electricity consumption dataset spans nearly three years. The first two years were used for training, with the rest reserved for testing. A proper time window is critical for capturing consistent patterns during training.

The time window was chosen based on the following assumptions:

- Weekday electricity usage is more consistent, as most people follow regular work or school schedules.
- Midnight to early morning consumption is steady, as most people are typically asleep.

HMM

- The time window Wednesday 5 AM to 7 AM was selected.
- This period minimizes variability caused by weekend activities and captures the gradual increase in electricity consumption as people wake up and start their day.
- Training data for the selected variables was plotted, revealing clear, consistent patterns during this time window, validating its suitability for model training.

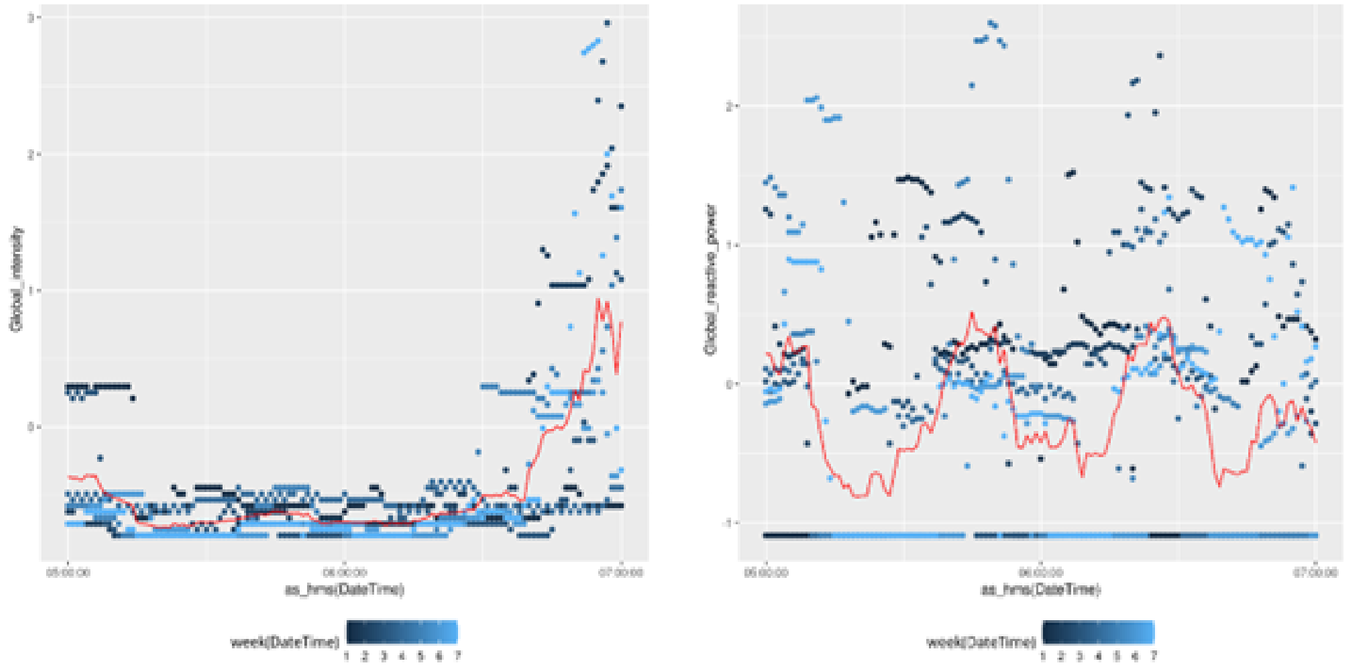


Figure 1

HMM

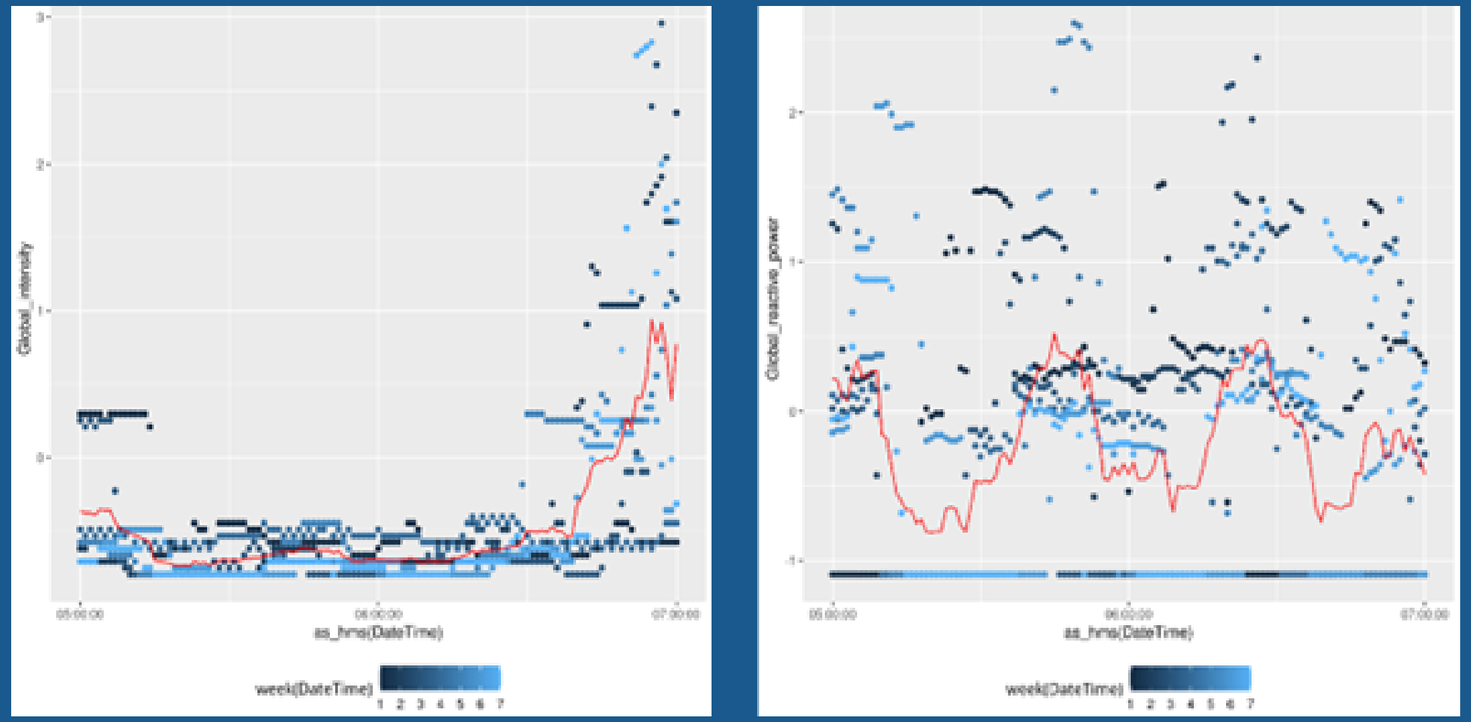


Figure 1

As shown in Figure 1, Global_intensity stays low for most of the time, sharply increasing as it approaches 7 AM, aligning with our assumptions. Global_reactive_power remains near -1.1 (zero before scaling) and increases periodically before returning to -1.1. This behavior is expected, as power plants typically keep reactive power near zero by adjusting for inductive/capacitive loads on the grid through reactive power compensation (Dixon et al., 2005). These patterns in the response variables suggest that the selected time window contains detectable trends, making it suitable for analysis using an HMM.

HMM

The training set, filtered by the chosen time window, consisted of 12,584 datapoints, divided into 104 time sequences of 121 points each. Hidden Markov Models (HMMs) were trained with varying nstates (4, 5, 6, 7, 8, 10, 12, 14, 16, 18, 20) using the depmixS4 package, and their log-likelihoods and BICs were compared (Figure 2).
Key observations from Figure 2:

- Nstates 4 to 10: Significant improvements in log-likelihood and BIC.
- Nstates 12 to 14: Marginal improvement, suggesting diminishing returns.
- Nstates 16 to 20: Slight gains, but the improvements are minimal.

The results indicate that while models with more states capture finer variations, the improvements are constrained by limited training data, leading to underfitting. This highlights the need for additional time-series data to enhance model performance effectively.

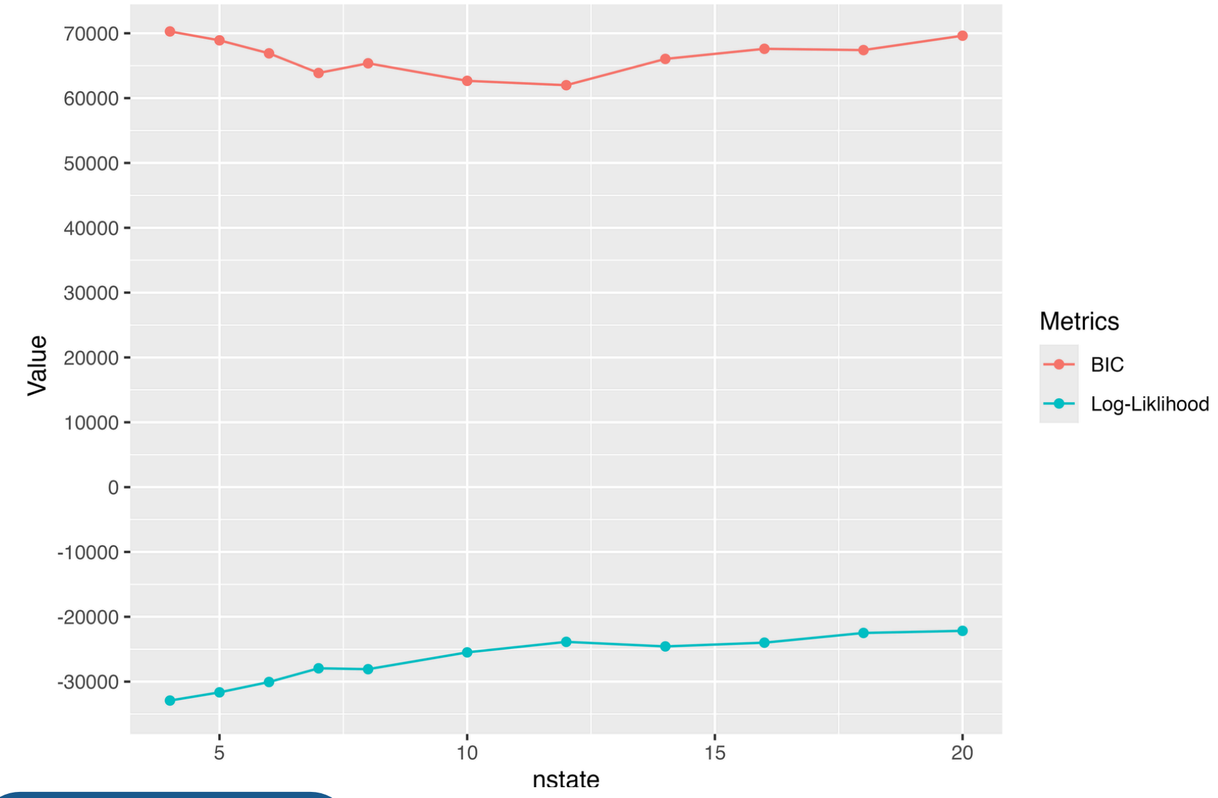


Figure 2

The forward-backward algorithm, essential for HMM training, estimates the probabilities of state sequences at each timestep, incorporating uncertainty into the process and enabling model training with incomplete labeling (Baum et al., 1970; Rabiner, 1989).

Using the depmixS4 forwardbackward() function, we computed the posterior probabilities and normalized log-likelihoods for testing data, based on the fitted models from training. Normalization ensured fair comparison across models, eliminating biases from dataset size.

Figure 3 highlights that the 10-state model achieved the highest normalized negative log-likelihood on the test data, identifying it as the least overfit model.

The 10-state model's superior performance on both training and test sets confirms its ability to generalize to unseen data, making it the most practical choice for real-world applications.

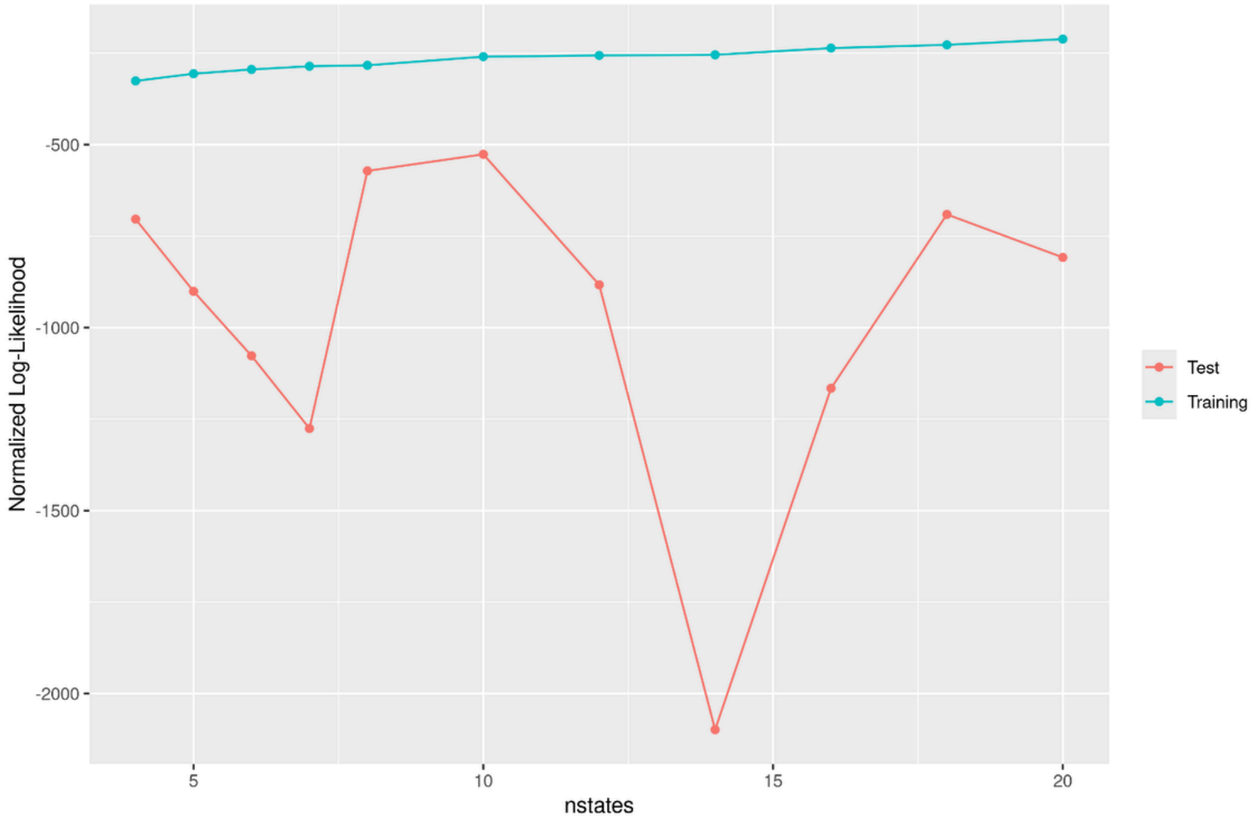


Figure 3

Log-Likelihood Analysis and Comparison

```
> print(log_likelihoods)
         1          2         3          4          5          6         7          8          9         10
-10.0533201 -11.6137236 0.7571379 -2.1409780 -2.6572122 -3.6187775 2.8469595 -3.2768775 -7.7300518 -3.6750779
```

Figure 4

To evaluate the performance of the Hidden Markov Model (HMM) and its ability to generalize to unseen data, the normalized training log-likelihood (-3.15) was compared to the log-likelihoods of 10 test subsets. These subsets were created by dividing the test data into 10 equal-sized, chronologically ordered parts, ensuring a balanced and systematic analysis.

Figure 4 illustrates the log-likelihoods for the test subsets:

- Subset 7, with a log-likelihood of 2.85, closely aligned with the training data, indicating strong consistency.
- Subset 2, with a log-likelihood of -11.61, showed greater deviation, suggesting less alignment with the training patterns.
-

To quantify these deviations, the maximum difference between the test subset log-likelihoods and the training log-likelihood was calculated, resulting in a threshold of 8.46. This threshold defines the acceptable range of normal behavior, where:

- Values exceeding this range indicate potential anomalies, either due to unusual deviations (too low) or overly close matches (too high).

Anomaly Detection

The analysis showed that all test subsets fell within the threshold, indicating a good fit to the test data with no apparent anomalies. The log-likelihood graph visualizes this, with subset log-likelihoods plotted against the upper and lower bounds. Subset 7 aligned most closely with the training data, while Subset 2 neared the lower threshold, suggesting a slight deviation nearing anomaly. These results confirm that the model effectively captures patterns in the training data and applies them consistently to the test data.

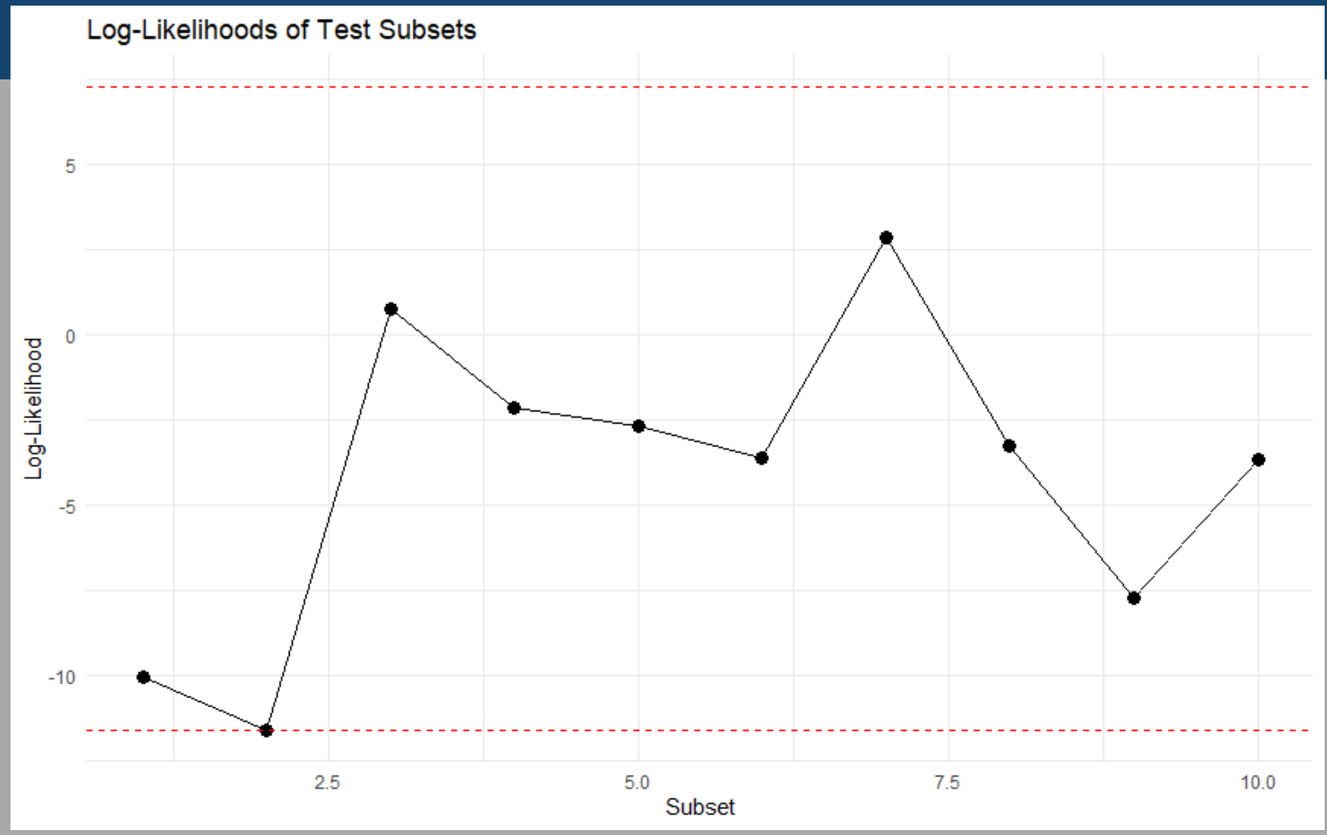
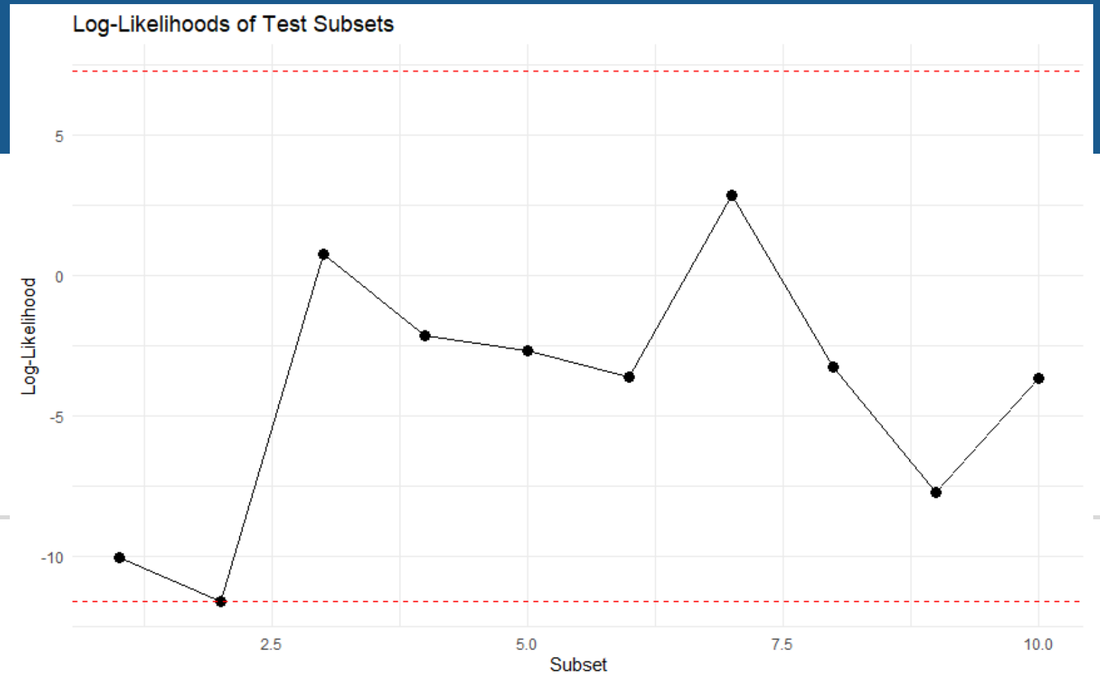


Figure 5

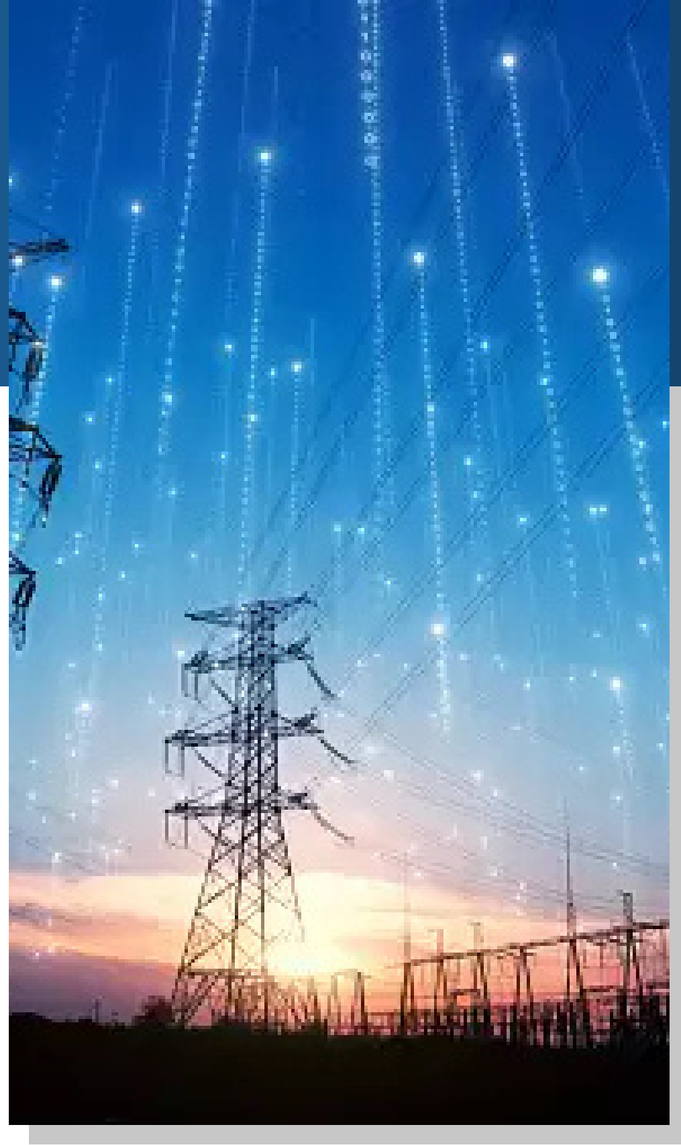
Anomaly Injection

Injecting anomalies involves modifying data to simulate abnormal behaviors, testing the model's ability to detect irregularities. In this case, anomalies were added by creating spikes in global_active_power, missing voltage readings, and increasing global_intensity values. These changes disrupted normal patterns, causing deviations from the model's expected behavior. The anomalies showed as significant deviations from the training data's log-likelihood threshold, demonstrating the model's potential for real-world anomaly detection.



Conclusion

This project explored anomaly detection using Hidden Markov Models (HMM) and developed a method for identifying unusual patterns in time-series data. By analyzing log-likelihoods from both training and test data, we were able to detect deviations that suggest anomalies. The project helped build key skills in data cleaning, feature engineering, and machine learning. It also highlighted the importance of interdisciplinary knowledge in understanding the dataset. Overall, the project laid a strong foundation for real-time anomaly detection, providing a framework for further development and practical use.



Citations

1- Shaibu, S. (2024, October 15). Normalization vs. Standardization: Key Differences Explained. DataCamp. https://www.datacamp.com/tutorial/normalization-vs-standardization

2- Dixon, J., Moran, L., Rodriguez, J., & Domke, R. (2005, December). Reactive Power Compensation Technologies: State-of-the-Art Review. Proceedings of the IEEE, 93(12), 2144–2164. https://doi.org/10.1109/JPROC.2005.859937

3- Baum, L. E., Petrie, T., Soules, G., & Weiss, N. (1970). A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. Annals of Mathematical Statistics, 41(1), 164–171. https://doi.org/10.1214/aoms/1177697196

4- Rabiner, L. R. (1989). A tutorial on hidden Markov models and selected applications in speech recognition. Proceedings of the IEEE, 77(2), 257–286. https://doi.org/10.1109/5.18626

Thank You

CMPT 318 D100 Fall 2024

Professor: Uwe Glaesser

Group 8:

- Simon Yu, 301451144
- Calvin Weng, 301556001
- Tin Liang, 301565565
- Nazanin Pouria Mehr, 301442860