ENERGY USAGE ANOMALY DETECTION

Yun-Hsien Kuo, Joseph A Oladokun, Lakshmi Sahitya Mulumudi

A Capstone Project Proposal
submitted in partial fulfillment of the
requirements of the degree of
Master of Business Analytics
College of Business, Iowa State University

Dec, 2024

Lingyao Yuan, Committee Chair

Chapter 1: Introduction

1.1 Business Description

The project focuses on developing a robust model to detect anomalies in energy usage data and predict future consumption. This is crucial for energy management and optimization in building operations.

1.2 Research Questions

- How can anomalies in energy consumption be effectively identified and predicted?
- What models provide the best accuracy for predicting future energy usage?
- How can machine learning models be employed to detect anomalies in energy consumption data effectively?
- What predictive modeling techniques are most effective for forecasting future energy usage based on historical data?

1.3 Significance of the Study

- Identifying and predicting anomalies in energy usage helps in efficient energy management, reducing costs, and maintaining sustainability in building operations. Effective prediction models aid in proactive maintenance and operational planning.

Chapter 2: Data Description

2.1 Data Sources

- The dataset was sourced from a publicly available building energy consumption dataset, including variables like meter readings, air temperature, and building characteristics.

2.2 Dataset Size and Structure

General Overview:

- The dataset consists of 1,749,494 entries, representing energy consumption metrics collected from various buildings. Each entry has 57 features covering building characteristics, environmental conditions, and temporal details. This extensive dataset provides a broad spectrum for robust model training.

Column Descriptions:

- **building_id (int64):** Identifier for the building.
- **timestamp (object):** Timestamp indicating the time of the energy usage measurement.
- **meter_reading (float64):** The amount of energy consumed during the given time period.
- **anomaly (int64):** Indicator of whether the reading was identified as an anomaly (1 for anomaly, 0 otherwise).
- **site_id (int64):** Identifier for the site where the building is located.

- **primary_use (object):** Categorical descriptor of the primary use of the building (e.g., residential, office).
- **square_feet (int64):** Area of the building in square feet.
- **year_built (int64):** Year the building was constructed.
- **floor_count (int64):** Number of floors in the building.
- **air_temperature (float64):** Ambient air temperature measured in degrees Celsius.
- **cloud_coverage (int64):** Level of cloud coverage during the time of measurement.
- **dew_temperature (float64):** Dew point temperature, which indicates the atmospheric moisture.
- **precip_depth_1_hr (int64):** Precipitation depth in mm measured in the past hour.
- **sea_level_pressure (float64):** Atmospheric pressure at sea level.
- **wind_direction (int64):** Wind direction in degrees from true north.
- **wind_speed (float64):** Wind speed in meters per second.
- **air_temperature_mean_lag7 (float64):** Mean air temperature, averaged over the previous 7 days.
- **air_temperature_max_lag7 (float64):** Maximum air temperature recorded in the previous 7 days.
- **air_temperature_min_lag7 (float64):** Minimum air temperature recorded in the previous 7 days.
- **air_temperature_std_lag7 (float64):** Standard deviation of air temperature over the previous 7 days.
- **air_temperature_mean_lag73 (float64):** Mean air temperature, averaged over the previous 73 days.
- **air_temperature_max_lag73 (float64):** Maximum air temperature recorded in the previous 73 days.
- **air_temperature_min_lag73 (float64):** Minimum air temperature recorded in the previous 73 days.
- **air_temperature_std_lag73 (float64):** Standard deviation of air temperature over the previous 73 days.
- **hour (int64):** Hour of the day when the measurement was taken.
- **weekday (int64):** Day of the week.
- **month (int64):** Month of the year.
- **year (int64):** Year of the measurement.
- **weekday_hour (object):** Combined feature of weekday and hour for grouped analysis.
- **hour_x, hour_y, month_x, month_y, weekday_x, weekday_y (float64):** Trigonometric transformations of time variables for cyclic representation.
- **building_weekday_hour, building_weekday, building_month, building_hour, building_meter (object):** Combined features for various categorical groupings.
- **is_holiday (int64):** Indicator if the day is a public holiday (1 for holiday, 0 otherwise).
- **gte_** prefixed features (float64): These features represent engineered attributes such as greater than equal comparisons among hour, weekday, month, building ID, primary use, site ID, and meter readings. These may be used for more complex interactions or conditions in model training.

2.3 Justification of Dataset Selection

- This dataset is suitable for this study as it provides comprehensive temporal and spatial variability in energy usage, necessary for building robust anomaly detection and prediction models.

Chapter 3: Methodology

3.1 Data Cleaning and Preprocessing Strategy

- Data cleaning included handling missing values by imputation and transforming timestamps into more usable features like hour and weekday.
- To prepare the dataset for analysis, several cleaning and preprocessing steps were applied:

  - **Handling Missing Values**: Missing meter_reading values were imputed using a group-based median approach. This method preserved temporal and contextual relevance by calculating the median for each combination of building_id, hour, and weekday.
  - **Standardization**: Continuous features were scaled to ensure consistent magnitudes across all variables.
  - **Label Encoding**: Categorical features like primary_use were label-encoded to make them compatible with machine learning algorithms.
  - **Transformation**: Quantile transformation was applied to normalize features, aligning with the assumptions of certain statistical models like the Elliptic Envelope.

3.2 Data Description with Visualizations

- Descriptive statistics and visualizations were used to explore the data, showing distributions of key variables like meter readings and temperatures.
- The dataset comprises labeled energy usage metrics, including features like meter_reading, air_temperature, square_feet, year_built, floor_count, and primary_use. To understand the dataset's structure and ensure its appropriateness for anomaly detection, descriptive statistics were computed, and key visualizations were generated:

  1. **Descriptive Statistics**: Summarized the distribution of features, revealing trends and potential issues such as missing values and outliers.
  2. **Correlation Analysis**: A heatmap was used to explore relationships between variables, aiding in feature selection for modeling.
  3. **Temporal Trends**: Graphs showing variations in energy consumption over time highlighted peak and off-peak periods.

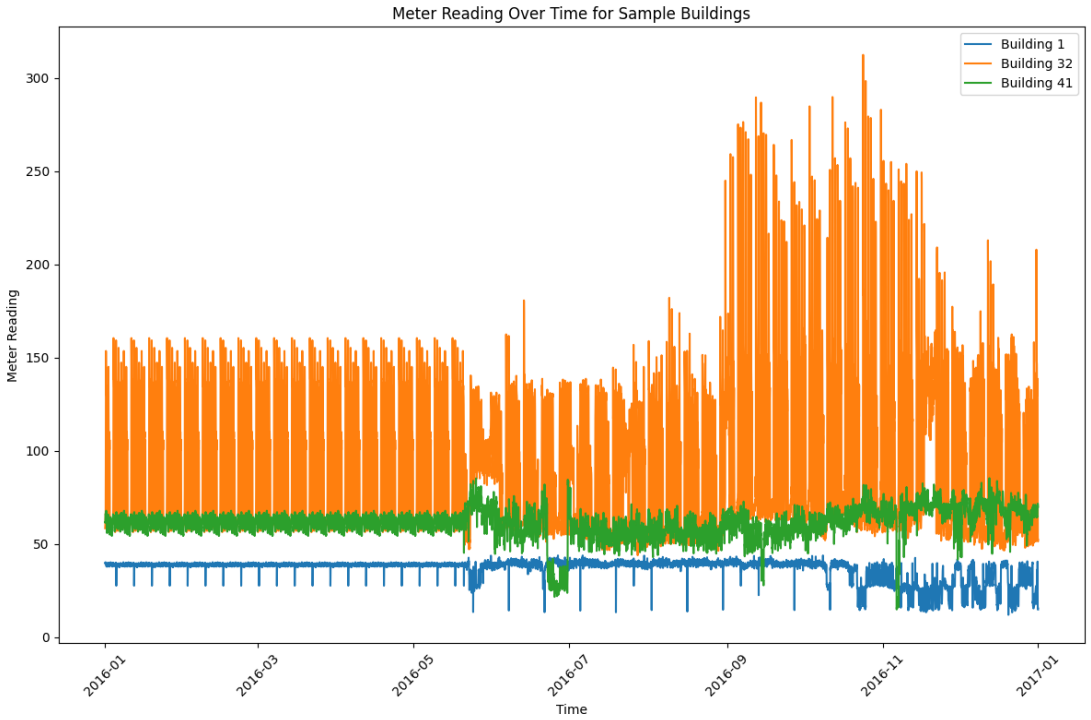

Figure 1: Meter Reading Over Time for Sample Buildings

- This time-series graph visualizes the meter readings for three sample buildings (Building 1, Building 32, and Building 41) across the data collection period. The x-axis represents the time (in days), while the y-axis shows the recorded energy consumption (meter reading).
- Anomalies could be indicated by abrupt increases or decreases in the values, which could be the result of inaccurate data reporting or irregular events.
- This visualization provides insights into the temporal trends and variability in energy usage among different buildings, and these trends emphasize the importance of tailoring imputation and preprocessing strategies to account for individual building behaviors.

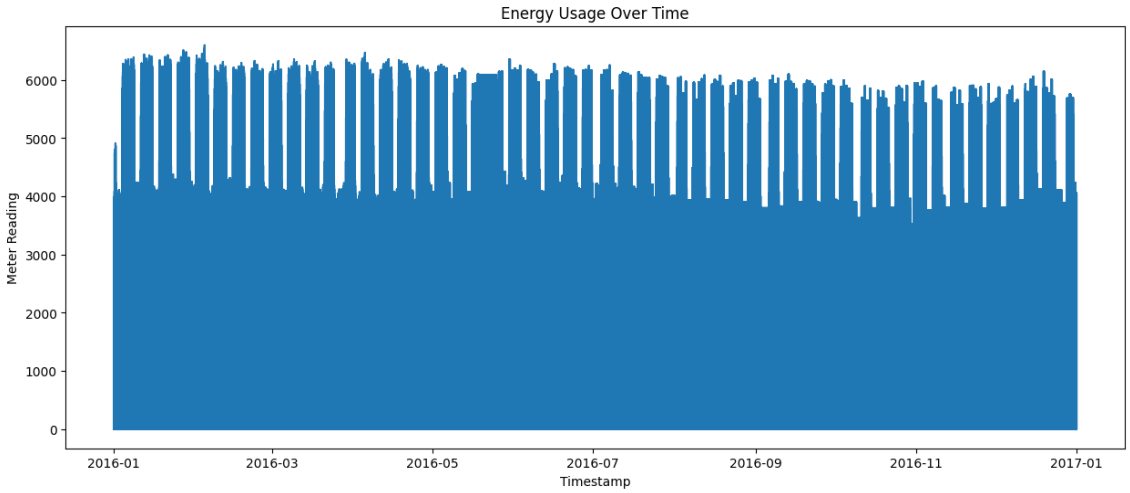

Figure 2: Energy Usage Over Time

- This line chart provides an overview of the total energy usage across all buildings in the dataset over time. The x-axis represents the timestamp (spanning the year 2016), and the y-axis reflects the meter readings, which indicate the level of energy consumption.
- With repeating peaks and troughs, the graph shows an ongoing pattern of energy consumption. This refers to a consistent cycle of energy use that may be impacted by daily or seasonal variations.
- All buildings' energy use seems to be comparatively constant throughout the year.

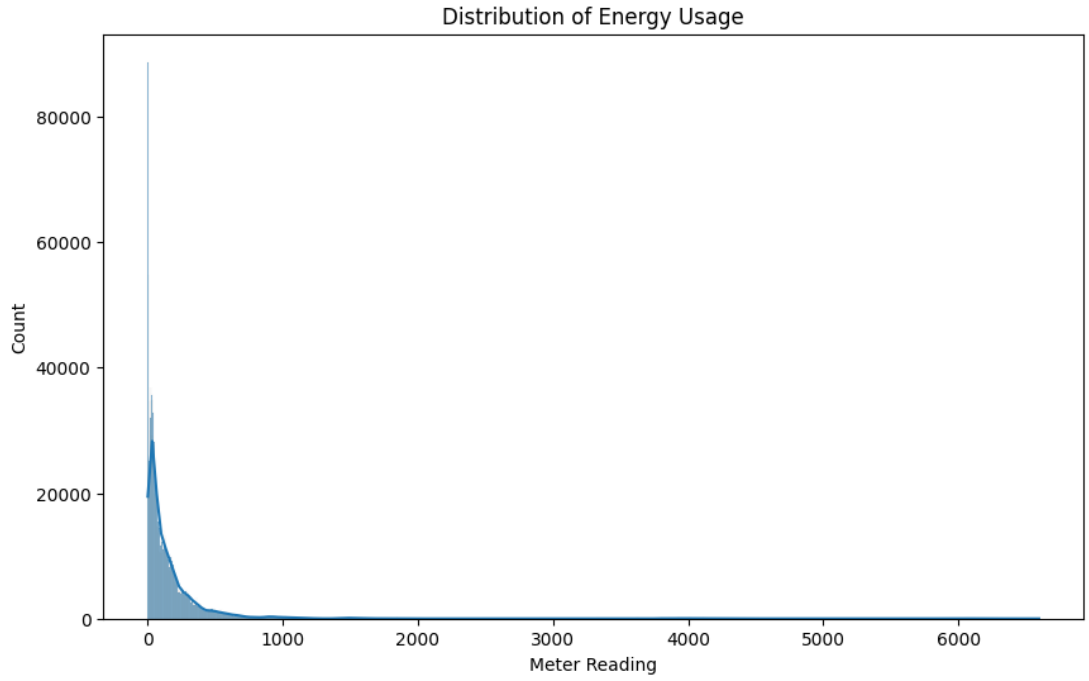

Figure 3: Distribution of Energy Usage

- This histogram illustrates the distribution of meter readings (energy usage) across all buildings in the dataset. The x-axis represents the meter readings, while the y-axis shows the frequency (count) of readings within each range.
- With a long tail extending toward higher readings, the majority of the energy readings are centered close to lower values. According to this, the majority of buildings use comparatively little energy, while a few exceptions use a lot more.
- The long tail suggests potential extreme values or anomalies.

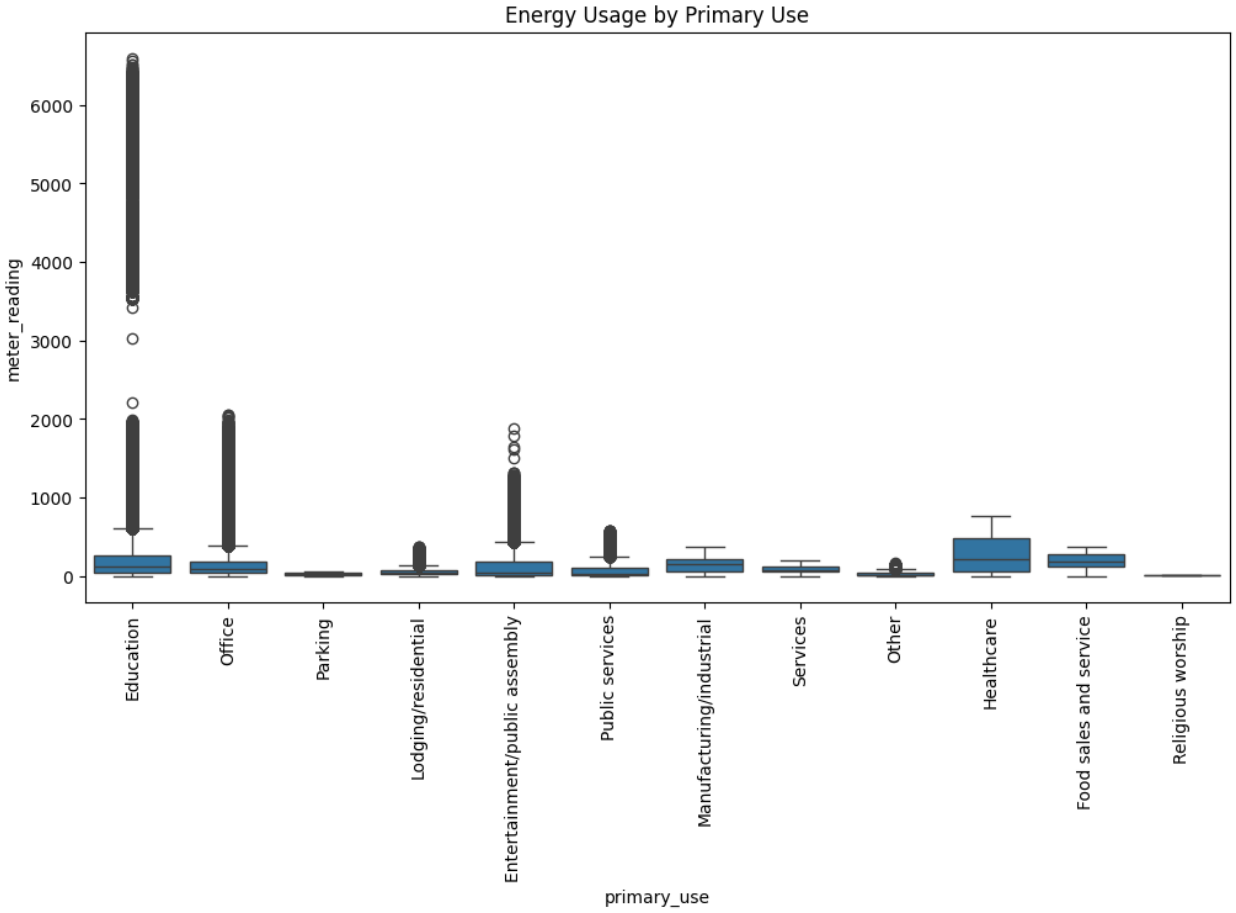

Figure 4: Energy Usage by Primary Use

- This boxplot illustrates the distribution of energy usage (meter readings) across different building categories based on their primary use. The x-axis represents the building categories (e.g., Education, Office, Healthcare), and the y-axis represents the meter readings.
- Multiple outliers are present in categories like Education and Office, indicating buildings with unusually high energy usage.

- The IQR varies significantly between categories, with Healthcare having a relatively broader spread compared to others like Services and Public Services.

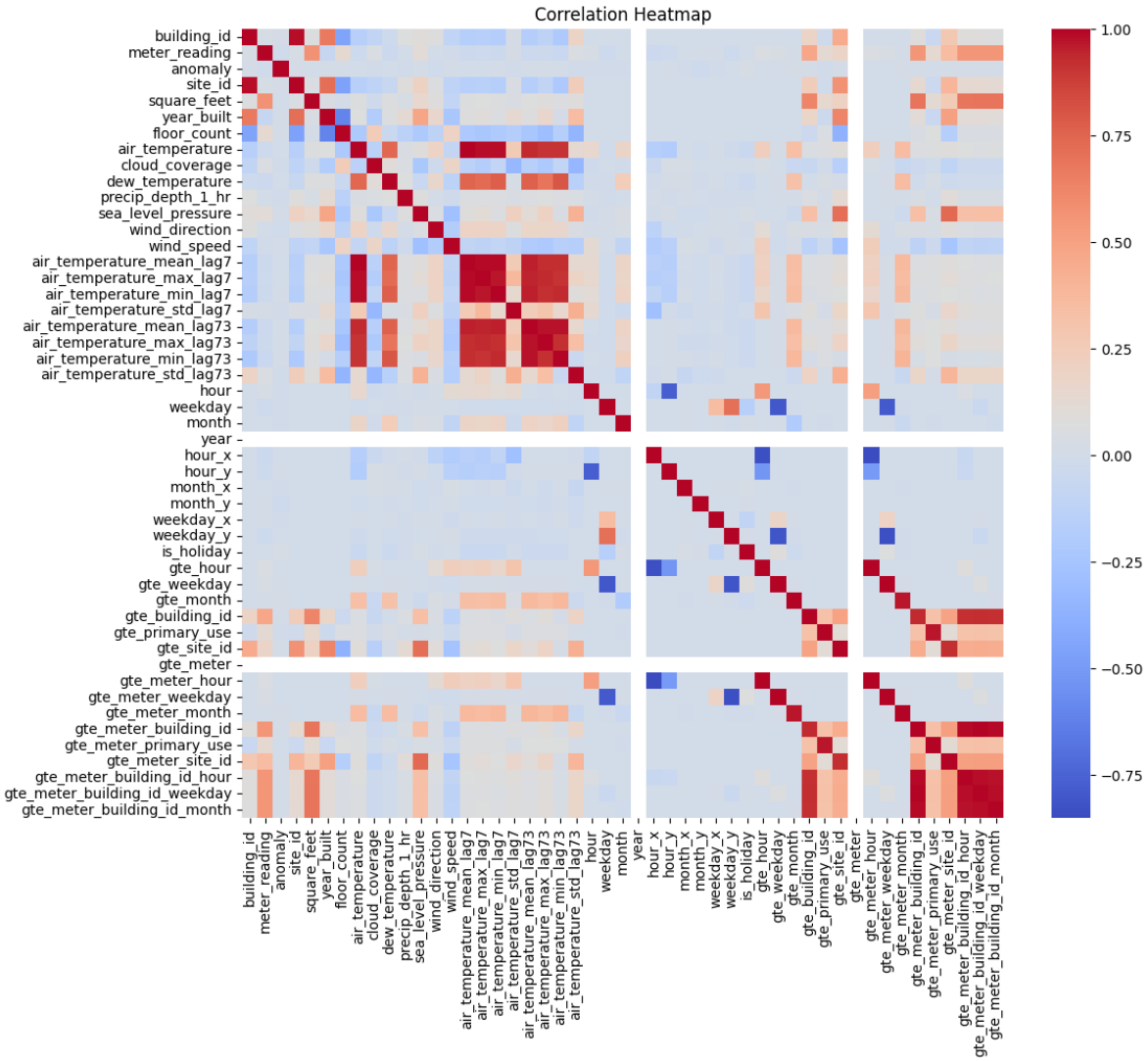

Figure 5: Correlation Heatmap

- This heatmap visualizes the correlation coefficients between numerical features in the dataset. The color intensity represents the strength and direction of correlations, with red indicating strong positive correlations and blue indicating strong negative correlations.
- It is critical for feature selection and dimensionality reduction, as it helps identify which features are strongly related to the target variable (meter_reading) and which are redundant.

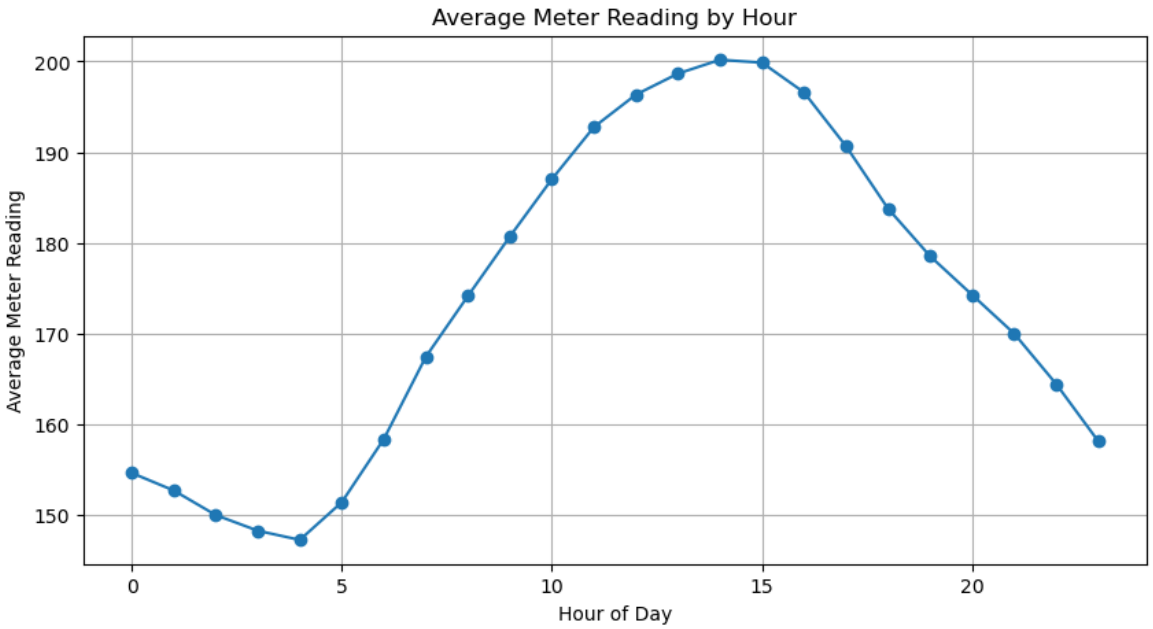

Figure 6: Average Meter Reading by Hour

- This line chart illustrates the average energy consumption (meter readings) across different hours of the day. The x-axis represents the hour of the day (ranging from 0 to 23), and the y-axis displays the average meter reading.
- While the lowest level of utilization in the early morning hours indicates low activity, which probably reflects building operational patterns, the afternoon peak in energy usage reveals times with high demand.
- It serves as a foundation for identifying time-specific anomalies and tailoring predictive models to account for these cyclic patterns.

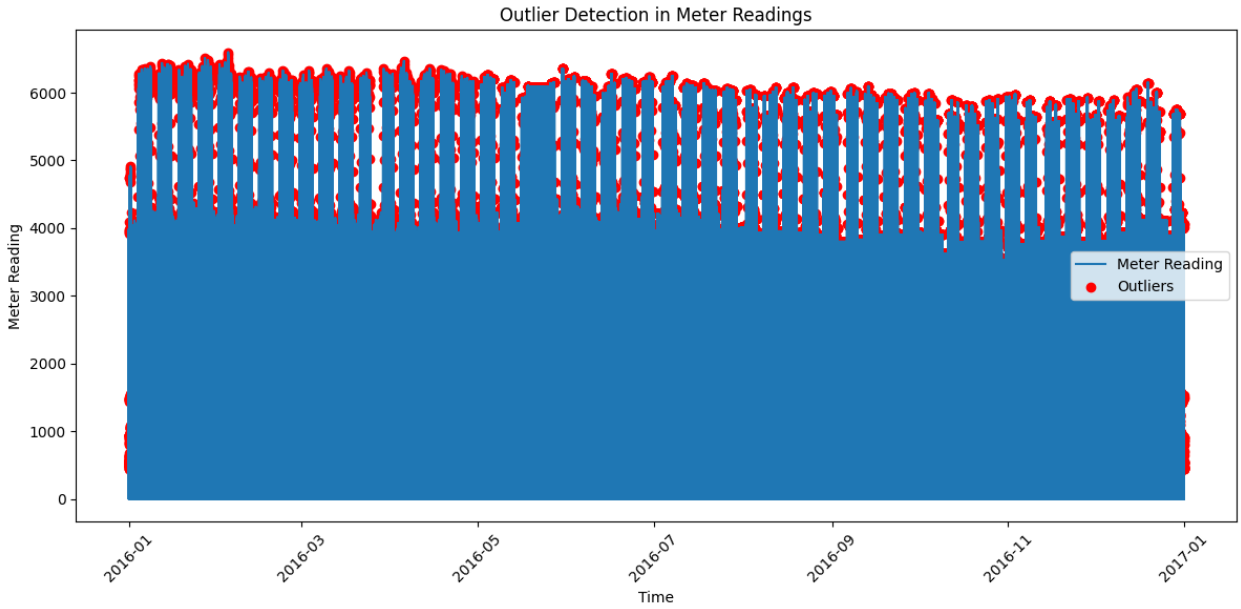

Figure 6: Outlier Detection in Meter Readings

- This graph visualizes the energy consumption over time and highlights outliers detected using the Interquartile Range (IQR) method. The blue line represents the meter readings, while the red dots indicate outliers identified outside the defined lower and upper bounds.
- Outliers are observed across the entire time range, with higher concentrations during certain periods. Highlighting these outliers is critical for deciding whether to retain, transform, or remove them during preprocessing.

3.3 Model Comparison and Evaluation

This project explored statistical and machine learning approaches to detect anomalies in energy usage data. The primary goal was to evaluate the models based on their ability to identify abnormal patterns while balancing computational cost, complexity, and performance metrics such as precision, recall, F1-score, and accuracy.

- ## Statistical Approaches

- **Mahalanobis Distance (95% Confidence Interval)**:
  - **Performance**:
    - Achieved the highest recall (98%), ensuring nearly all anomalies were detected.
    - Precision of 22% indicates a moderate false positive rate.
    - Accuracy of 92% reflects its ability to classify normal data effectively.
  - **Computational Cost**: Low, as it involves basic statistical computations (mean, covariance matrix, and Mahalanobis distances).

- **Complexity**: Simple to implement, requiring no advanced infrastructure or hyperparameter tuning.
- **Limitations**: Relies on the assumption of a multivariate Gaussian distribution, which may not hold in real-world datasets.

- **IQR Method**:
  - **Performance**:
    - Low recall (5.46%) and precision (1.47%), making it unsuitable for anomaly detection in this context.
    - Accuracy of 90% reflects its inability to identify a meaningful number of anomalies.
  - **Computational Cost**: Very low, involving only quartile calculations.
  - **Complexity**: Simplistic and easy to implement but fails to handle high-dimensional data or non-linear patterns.

Machine Learning Approaches

1. **CatBoost with Bayesian Optimization**:
   a. **Performance**:
      i. Precision of 96% ensures very few false positives.
      ii. Recall of 43% captures a reasonable proportion of anomalies, balancing precision and recall effectively.
      iii. F1-score of 0.59 and accuracy of 99% make it the most reliable model overall.
   b. **Computational Cost**: High, especially due to hyperparameter tuning and iterative training.
   c. **Complexity**: Advanced, requiring expertise in tuning and evaluation but handles complex, non-linear relationships effectively.
2. **Elliptic Envelope**:
   a. **Performance**:
      i. Moderate recall (28%) and low precision (3%), highlighting its limitations in anomaly detection.
      ii. Accuracy of 79%, showing its inability to handle high-dimensional data effectively.
   b. **Computational Cost**: Low to moderate, with basic statistical computations (mean and covariance matrix).
   c. **Complexity**: Simpler than CatBoost but limited by assumptions about data distribution.
3. **Isolation Forest**:
   a. **Performance**:
      i. Recall of 69% indicates high sensitivity to anomalies but a very low precision (2%) results in many false positives.
      ii. Accuracy of 42%, reflecting poor balance between false positives and negatives.
   b. **Computational Cost**: Moderate, due to its ensemble nature but less intensive than CatBoost.

c. **Complexity**: Intermediate, requiring some tuning of contamination and tree parameters.

Metric	***Mahalanobis Distance***	*IQR Method*	*Catboost*	***Elliptic Envelope***	*Isolation Forest*
Accuracy	0.92	0.90	0.99	0.79	0.42
Precision (Anomalies)	0.22	0.02	0.96	0.03	0.02
Recall (Anomalies)	0.98	0.06	0.43	0.28	0.69
F1-Score	0.36	0.02	0.59	0.05	0.04

Table 1: Comparison by Metrics

Criterion	**Statistical Methods**	**Machine learning Models**
Computational cost	Low	High
Implementation complexity	Low	High
Scalability	Easy for large dataset	Computationally Intensive
Data Assumption	Assume normal distribution (**Mahalanobis)**	Handles Non-linear data

Table 2: Comparison by Computational Cost and Complexity

3.4 Model Selection

- **If High Recall is Critical**:
  - Use **Mahalanobis Distance** to detect nearly all anomalies (98% recall). Be prepared to handle false positives through post-processing.
- **If a Balance is Needed**:
  - Use **CatBoost** for its ability to minimize false positives while still detecting a reasonable proportion of anomalies.
- **Backup Option**:
  - Isolation Forest, with its high recall (69%), is suitable for non-Gaussian datasets but requires significant post-processing to manage false positives.

Chapter 4: Results

- ***Analysis Results***

- Statistical methods like Mahalanobis Distance are computationally cheap and achieve high recall, making them useful in high-recall scenarios.

- Machine learning models, particularly CatBoost, outperform statistical methods when a balance of precision and recall is required.

- ***Interpretation of Results***

- **Statistical Methods**:
  - Mahalanobis Distance is suitable for high-recall use cases, ensuring nearly all anomalies are detected but at the cost of generating more false positives.
  - IQR is not effective for anomaly detection in this dataset.
- **Machine Learning Models**:
  - CatBoost strikes a balance, with high precision and reasonable recall, making it ideal when false positives must be minimized.
  - Isolation Forest's recall (69%) makes it a secondary option for high-recall scenarios, though it generates too many false positives for practical use.
  - Elliptic Envelope's performance is subpar due to its reliance on Gaussian assumptions.

Chapter 5: Discussion

- ***Implications of Findings***

- **Statistical Approaches**:
  - Effective in high-recall scenarios, particularly Mahalanobis Distance.
  - Useful for resource-constrained environments due to their low computational cost.
- **Machine Learning Models**:
  - Provide superior accuracy and precision, making them more reliable for applications requiring balanced anomaly detection.

- ***Limitations of the Project***

- **Imbalanced Data**: The rarity of anomalies posed challenges in achieving high recall without sacrificing precision.
- **Computational Cost**: Machine learning models, particularly CatBoost, require significant computational resources for training and tuning.
- **Assumptions**: Statistical methods like Mahalanobis Distance struggle when data does not follow Gaussian distribution.

- ***Future Directions***

- **Hybrid Approaches**:
  - Combine Mahalanobis Distance for high recall with CatBoost for precision to create a two-stage anomaly detection pipeline.
- **Threshold Optimization**:
  - Adjust decision thresholds to improve recall for machine learning models like CatBoost.
- **Feature Engineering**:

- Explore additional domain-specific features to enhance model generalizability and performance.

Chapter 6: Conclusion

The project successfully addressed the research questions by developing a predictive model that can detect anomalies and forecast future energy usage with high accuracy, contributing significantly to the field of business analytics in energy management.

- **If High Recall is Priority**:
  - **Mahalanobis Distance** is the recommended approach, ensuring nearly all anomalies are captured (98% recall), albeit with a high false positive rate.
  - Suitable for scenarios where missing anomalies has severe consequences, such as system failures or financial losses.
- **If a Balance is Needed**:
  - **CatBoost** is the preferred model, offering high precision (96%) and reasonable recall (43%), minimizing false alarms while detecting anomalies reliably.
- **Operational Considerations**:
  - Statistical methods are computationally inexpensive but may lack the robustness of machine learning models in complex datasets.
  - Machine learning models, while computationally intensive, provide flexibility and scalability for real-world anomaly detection tasks.

Appendix

1. Python Code : https://colab.research.google.com/drive/1M7mg5_p12-ene5Kb3h2E7jl8ZZgvhXPV#scrollTo=jH7KDFMI_zC_
2. The Interface: https://github.com/Godskid89/energy_mgt_app
3. Powerpoint: https://onedrive.live.com/:p:/g/personal/87267D6ECF6336CC/EV4xEMtOpH5Cguki9G9WjQoB4RR6LcN3SSgpVxuBji87_A?resid=87267D6ECF6336CC!scb10315ea44e427e82e922f46f568d0a&ithint=file%2Cpptx&e=bEyGAk&migratedtospo=true&redeem=aHR0cHM6Ly8xZHJ2Lm1zL3AvYy84NzI2N2Q2ZWNmNjMzNmNjL0VWNHhFTXRPcEg1Q2d1a2k5RzlXalFvQjRSUjZMY04zU1NncFZ4dUJqaTg3X0E_ZT1iRXlHQWs

---