Tennessee Energy Insights: A Predictive Analytics Project for Forecasting Energy Demand by Sector

Hamza Salah

DSC630 – Predictive Analytics

Professor Andrew Hua

Spring 2025

Table of Contents

Executive Summary

Tennessee Energy Insights is a proposed initiative that will leverage data analytics to examine and optimize residential electricity consumption across the state of Tennessee. Given the growing energy demand driven by new infrastructure developments, such as Elon Musk's factory in the region and the impact of shifting climate patterns, understanding the drivers of electricity usage has become a vital priority for utility companies and policy makers alike.

In this project, time series forecasting will be a key technique used to predict future energy demand based on historical data. Time series forecasting helps to model and predict the behavior of energy demand over time by capturing underlying trends, seasonality, and other patterns that can influence consumption. The primary focus will be on using historical temperature and electricity pricing data to develop a predictive model.

The anticipated outcomes of this project aim to provide actionable insights for utility providers, policymakers, and energy planners in Tennessee. By uncovering the factors that most significantly drive energy consumption, stakeholders will be better equipped to adjust pricing strategies, improve demand forecasting, and implement targeted energy efficiency programs. Ultimately, this initiative seeks to support more sustainable energy practices and inform future infrastructure and policy decisions in the state of Tennessee.

Introduction

Project Redirection

Originally focused on forecasting e-commerce sales, the project was refocused in Milestone 3 to explore residential energy consumption in Tennessee. This pivot was driven by a desire to work with a more realistic and socially relevant scenario, one that aligns more closely with current energy concerns in the region. It also takes advantage of publicly available environmental and economic data. While the topic has shifted, the core regression modeling framework remains consistent and has been adapted to this new context.

Research Problem

This research addresses the rise in electricity consumption in the state of Tennessee, considering recent developments such as Elon Musk's new project, which is expected to drive increased energy consumption through expanded industrial operations and technological infrastructure.

Purpose of the Study

This research aims to identify patterns in energy consumption across different sectors in the state of Tennessee, with a particular focus on the role of average temperature as a driving factor.

Significance of the Research

The findings from this project can help utility companies and energy planners better understand how consumption varies by sector and how it correlates with temperature trends. This insight can support more efficient energy distribution, policy planning, and sustainability efforts.

Origin

The project originated from a data science initiative focused on energy forecasting and climate-related consumption patterns. It centers on building accurate predictive models and uncovering sector-specific trends to inform future decision-making.

Stakeholders

Key stakeholders include city officials, utility companies, energy policy makers, and community planning agencies, all of whom have an interest in managing energy resources effectively.

Scope

The study covers energy consumption data from 2010 to 2024 across the Residential, Commercial, and Industrial sectors in Tennessee. Limitations include incomplete data for the Transportation sector, potential reporting inconsistencies, and the assumption that temperature is the primary external variable influencing consumption.

Data Collection and Structure

Data Sources

The project will use two key datasets to analyze residential energy consumption in Tennessee:

1. **Electricity Consumption Data (Sales):**
   This dataset retrieved from EIA.gov provides monthly electricity consumption data (in kWh) for residential customers. "Sales" refer to the total electricity sold to end-users, which serves as a proxy for energy consumption during the recorded period.
2. **Temperature Data :**
   Monthly temperature data (in degrees Fahrenheit) across Tennessee retrieved from NOAA will be used to examine how temperature fluctuations affect energy usage. This dataset will help assess the climatic impact on consumption.

The datasets will be merged to align monthly consumption data with the corresponding temperature and electricity price data, creating a comprehensive dataset for regression analysis.

Variables Description

The dataset comprises columns merged from multiple datasets, with additional variables retained specifically for usage determination during regression analysis.

- **Year**: Represents the year in which the data was recorded.
- **Month**: Represents the month in which the data was recorded.
- **Average_Temp**: Represents the average temperature during the recorded period.
- **Revenue**: Total revenue generated from electricity sales during the recorded period.
- **Sales (Consumption)**: Total amount of electricity sold during the recorded period, typically measured in kilowatt-hours (kWh) or similar units.

- **Customers**: Number of residential customers who purchased electricity during the recorded period.
- **Price**: Average price charged for electricity during the recorded period, typically measured in cents per kilowatt-hour (¢/kWh) or a similar unit.

Data Cleaning and Preparation

Describes the steps taken to handle missing or inconsistent data. The research questions will make more sense once the reader understands the data collection and cleaning process.

1. **Renaming Columns**:
   • Renamed columns to make them more meaningful and align them with the relevant sector names.
   o e.g., RESIDENTIAL → res_rev, COMMERCIAL → com_rev, etc.
2. **Dropping Rows and Columns**:
   • Dropped the first two rows (index=[0,1]) and the status column since they contained irrelevant information.
   • Removed the last row, which contained footer data not relevant to the analysis.
3. **Resetting Index**:
   • After dropping rows, reset the index of the DataFrame to ensure consistency.
4. **Converting Date Column**:
   • Combined the year and month columns to create a new date column, converting it into a datetime object with the day set to 1.
5. **Filtering Data for Temperature Data**:
   • Applied mini lambda functions to convert data into strings or integers and split the date column into year and month.
   • Renamed columns in the temperature DataFrame (df_temp) for clarity and restructured the date column into a proper datetime object.
6. **Filtering for Tennessee**:
   • Filtered the dataset to only include rows for Tennessee (df['state'] == 'TN'), as the project focuses on this state.

7. **Merging DataFrames**:
   • Merged the temperature data (df_temp) with the main data (df) on the date column to combine both datasets.

8. **Splitting by Sector**:
   • Created separate DataFrames for each sector: residential, commercial, industrial, transportation, and total, to analyze them individually.

9. **Renaming Columns in Sector DataFrames**:
   • Renamed the relevant columns in each sector DataFrame to general terms (e.g., revenue, consumption, customers, price), removing the sector-specific abbreviations.

10. **Standardizing and Transforming Sector Columns**:
    • Standardized the consumption and avg_temp columns in each sector DataFrame using a scaler (e.g., StandardScaler).
    • Created new columns consumption_scaled and avg_temp_scaled to store the transformed values.

Research Questions

Primary Research Question

How does average temperature and electricity pricing impact residential and sectoral electricity consumption patterns in Tennessee, and how can predictive analytics be used to forecast future consumption trends to help plan for the demand?

Secondary/Sub-Questions

- How does temperature impact electricity consumption in residential, commercial, industrial, and transportation sectors in Tennessee?

Methodology

Tools and Techniques

The following tools, programming languages, and frameworks were used throughout the analysis:

- **Python**: The primary programming language used for data analysis and modeling.
- **Pandas & NumPy**: For data wrangling, manipulation, and numerical computations.
- **Matplotlib & Seaborn**: For creating detailed visualizations and residual diagnostics.
- **Statsmodels**: To build ARIMA models, assess autocorrelation, and conduct time series diagnostics.
- **pmdarima**: Used for automatic ARIMA model selection (auto_arima).
- **Scikit-learn**: Employed for calculating evaluation metrics (e.g., RMSE, MAE, MAPE) and standardizing data where necessary.

Models Used

The models chosen for this analysis include **Linear Regression**, **ARIMA (Autoregressive Integrated Moving Average), and SARIMAX (Seasonal Autoregressive Integrated Moving Average with Exogenous Variables).**

- **Seasonality and Trends**:
  ARIMA and SARIMAX are particularly effective for time series data with strong trends and seasonal components—patterns clearly observed in the energy consumption data.
- **Univariate vs. Multivariate Forecasting**:
  While **Linear Regression** (specifically Multiple Linear Regression, MLR) uses multiple predictors for forecasting, **ARIMA** and **SARIMAX** are designed for univariate forecasting by modeling the internal time-dependent structure of the data.
  SARIMAX, on the other hand, extends ARIMA by incorporating external

variables (exogenous regressors), making it suitable for multivariate time series with seasonal influences.

- **Parameter Optimization**:
  Auto ARIMA (via the pmdarima package) was used to automatically select optimal parameters (p, d, q) for ARIMA-based models, including SARIMAX variants where applicable.

- **Diagnostic Checks**:
  Residual diagnostics such as histograms, Q-Q plots, ACF plots, and the **Ljung-Box test** were conducted to evaluate model assumptions and ensure residuals resembled white noise.

- **Forecast Evaluation**:
  Forecast accuracy was assessed using **RMSE** (Root Mean Squared Error), **MAE** (Mean Absolute Error), and **MAPE** (Mean Absolute Percentage Error), allowing for comparison across all models.

Evaluation Metrics

- **Mean Squared Error (MSE):** This will measure the average squared difference between the predicted and actual energy consumption values, providing insight into the accuracy of the regression model.
- **Mean Absolute Error (MAE):** This metric will provide a clearer interpretation of the average absolute difference between predicted and actual values in the same units as the target variable (kWh). It is useful for understanding the average magnitude of error in predictions.
- **Root Mean Squared Error (RMSE):** This metric, the square root of MSE, gives a more interpretable result in the same units as the target variable (kWh). RMSE is sensitive to large errors, making it useful for identifying significant prediction mistakes.
- **Mean Absolute Percentage Error (MAPE):** MAPE measures the average absolute percentage difference between predicted and actual values. It is helpful for understanding the relative error, especially when comparing models across different datasets or when absolute values are less important than proportional errors.

- **Coefficient of Determination (R Squared):** Represents the proportion of variance in the dependent variable that is predictable from the independent variables.

Justification of Methods

- **Auto ARIMA (via pmdarima)**

- **Reason for Use**: To ensure that the optimal parameters for the ARIMA model (p, d, q) were selected, the Auto ARIMA function was employed. This tool automates the model selection process based on statistical criteria such as AIC and BIC, which reduces the need for manual tuning and ensures the best fit for the data.

- **Diagnostic Checks (Residual Analysis, ACF, Q-Q Plot, Ljung-Box Test)**

- **Reason for Use**: These diagnostic checks were crucial for validating the assumptions of the ARIMA model and ensuring that it was appropriately capturing the underlying data structure. By examining the residuals, the model's predictive accuracy and adherence to assumptions (e.g., normality, no autocorrelation) could be assessed.

- **Evaluation Metrics (MSE, MAE, RMSE, MAPE, R2)**

- **Reason for Use**: These metrics were chosen because they provide both absolute and relative measures of model performance. MSE and RMSE give insights into the magnitude of prediction errors, while R^2 quantifies how well the model explains the variance in energy consumption. MAE and MAPE offer more intuitive, interpretable results for understanding prediction accuracy.

Data Analysis

Exploratory Data Analysis (EDA)

The dataset summarizes energy use across five sectors from 2010 to 2024. The Residential sector leads in revenue and consumption, followed by Commercial and Industrial, with Industrial offering the lowest prices. Transportation shows minimal activity. Combined sector data reflects overall averages: ~$803K revenue, ~8.3M units consumed, and 3.3M customers. Temperature remains steady (~59°F), while consumption and revenue vary widely across sectors.

Sector	Avg Temp (°F)	Revenue (Mean)	Consumption (Mean)	Customers (Mean)	Price (Mean)
Residential	59.10	$379,067	3,521,282	2,869,945	$10.80
Commercial	59.10	$300,732	2,811,132	492,098	$10.66
Industrial	59.10	$123,227	1,983,717	1,275	$6.16
Transportation	59.10	$5.08	45.66	0.33	$3.66
Combined	59.10	$803,031	8,316,176	3,363,320	$9.64

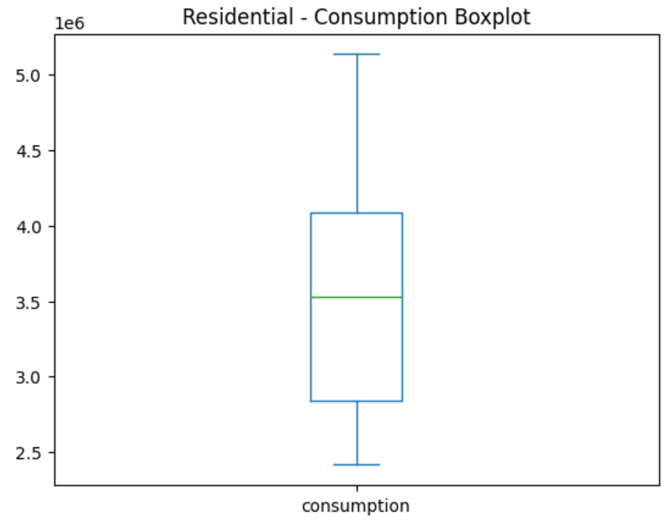
Techniques

In this project, I used a combination of **data cleaning**, **descriptive statistical analysis**, and **exploratory data analysis (EDA)** techniques. I summarized key metrics like mean, median, standard deviation, and ranges for variables such as revenue, consumption, customers, and price across different sectors. I also grouped data by sector and time to identify patterns, trends, and variability. These methods helped you compare energy usage behavior across sectors and uncover sector-specific insights.

The code I used for this project is listed in Appendix C (Code 1-17)

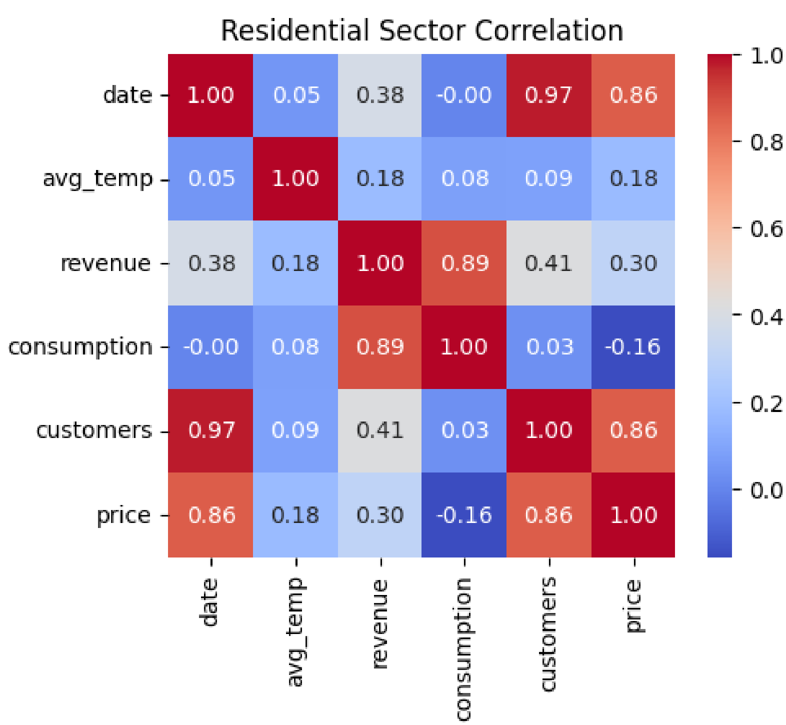
Visualizations

To check for outliers, I created a for loop that iterates over a list of DataFrames and outputs a boxplot. The plot below, which represents residential data, shows no outliers. The boxplots for the remaining sectors can be found in Appendix A (Figures 1-4).



The boxplots revealed four outliers in the commercial and industrial sectors. After reviewing the consumption data for these sectors, the values appeared consistent with typical usage patterns for the month of August. Since I planned to perform ARIMA modeling, I chose to retain these data points to capture the seasonal variation rather than remove them.

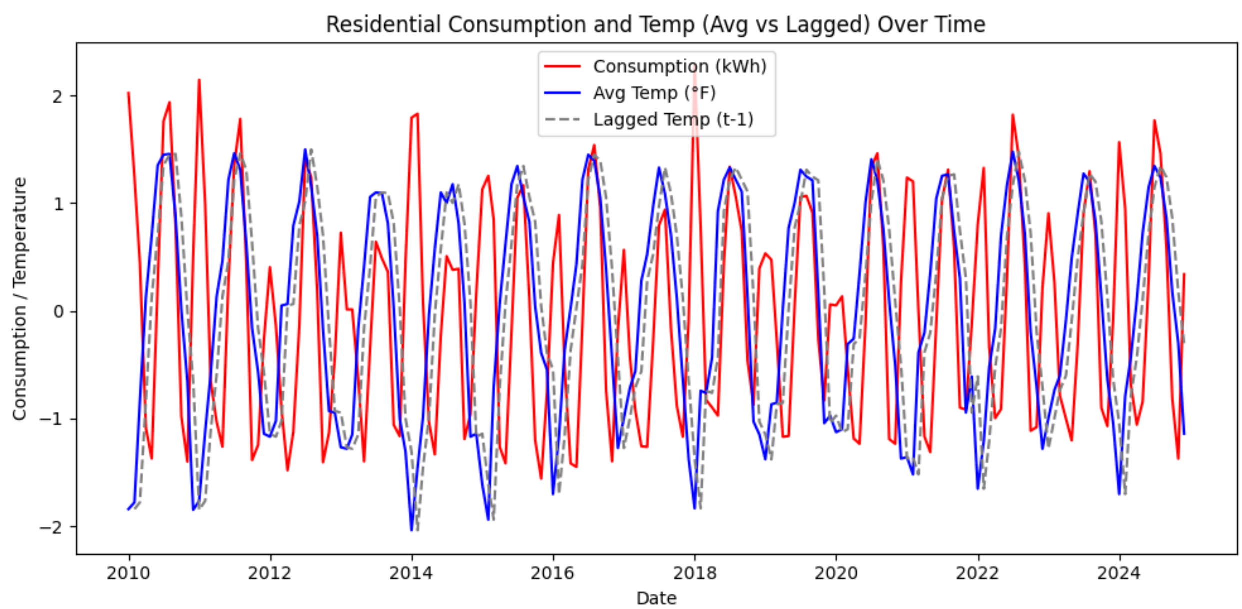
To analyze the relationship between electricity consumption and average temperature in the state of Tennessee, I created a correlation heatmap to visualize the strength of their correlation.



The correlation heatmap shows an 8% positive correlation which is very low, the other sectors in Appendix A (Figures 5-9) show a higher percentage, 60% for commercial and 26% for industrial and 9% for transportation. Overall, when combining all sectors, the correlation increases to 35%, highlighting that temperature influences energy consumption across sectors, but the strength of this relationship varies significantly by sector.

Further investigating the relationship between them, I graphed a time series dual axis chart of average temperature and residential energy consumption from 2010 to 2024. This visualization revealed seasonal patterns and potential correlations, such as increased consumption during periods of extreme temperatures. Additionally, I observed trends over the years that suggest how climate variability and long-term weather changes may be influencing residential energy consumption.

Extending the analysis to other sectors, I created similar visualizations for the commercial and industrial sectors. These graphs highlighted distinct consumption behaviors based on sector-specific patterns that required further research.



Residential consumption maintains a consistent relationship with temperature throughout the entire period, suggesting that weather-dependent usage—particularly HVAC—could be the primary driver of residential energy consumption.

The other sectors shown in Appendix A (Figures 9–12) follow a similar pattern, apart from the Industrial sector, which displays a steady decline in consumption regardless of temperature. From 2014 onward, the correlation weakens, indicating that temperature may not be the dominant factor influencing industrial energy use in later years.

It is also worth noting that the transportation sector is missing a significant portion of data and will therefore be excluded from both analysis and model development.

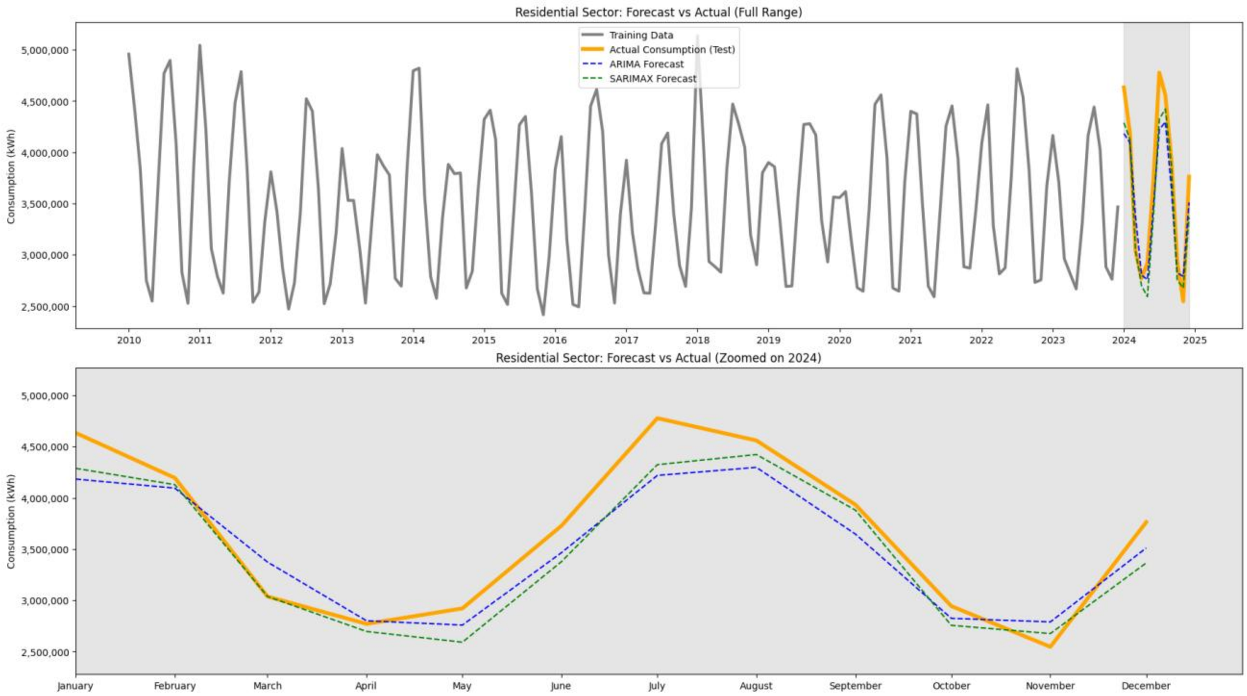
Next, I built multiple regression models to understand the relationship between energy consumption and key predictors such as lagged average temperature, price, and customer counts.

The regression analysis revealed that these variables, including temperature, alone had limited explanatory power, as evidenced by a low R^2 of 0.30, indicating that temporal dependencies and patterns were not adequately captured. Using this insight, I moved on to time series forecasting.

I began with a basic ARIMA model to capture trend and autocorrelation. Its performance was limited because it did not account for the strong seasonal patterns in electricity consumption, which highlighted the need for a seasonal approach. To address this, I applied auto_arima, which automatically selected the optimal seasonal ARIMA structure, significantly improving forecast accuracy by capturing both trend and seasonality without manual parameter tuning.

Finally, I implemented a SARIMAX model, manually specifying seasonal and non-seasonal orders and incorporating lagged temperature as an exogenous variable. This provided the most accurate forecasts and accounted for both seasonal patterns and external drivers.

Forecasting results for each sector are presented in Appendix A (Figures 13-18), and the corresponding performance metrics tables can be found in Appendix B (Table 1-9). The forecasted values (dashed blue line) closely follow the seasonal trend observed in previous years and align well with the actual 2024 consumption values, capturing both seasonality and trend.



The chart illustrates residential electricity demand forecasts compared to actual consumption values. The top chart the full historical range from 2010 to 2025, with the shaded area marking the 2024–2025 forecast period. Both ARIMA and SARIMAX follow the actual consumption trend, but SARIMAX aligns more closely, particularly around the seasonal peaks and troughs.

The bottom panel zoomed in in 2024, highlighting this difference more clearly. SARIMAX consistently mirrors the actual monthly fluctuations, while ARIMA slightly underestimates or lags. This is due to SARIMAX's incorporation of external variables like avg_temp and avg_temp_lagged with seasonal components, which allows it to better capture the underlying seasonal patterns and external influences affecting residential electricity consumption.

Forecast vs Actual Consumption (Residential)					
Month	Actual Consumption	ARIMA Forecast	SARIMAX Forecast	ARIMA Error %	SARIMAX Error %
1/1/2024	4,633,223.30	4,182,870.70	4,287,450.10	9.72	7.46
1/2/2024	4,194,306.60	4,094,578.90	4,128,448.60	2.38	1.57
1/3/2024	3,037,142.60	3,373,081.20	3,037,017.60	11.06	0
1/4/2024	2,770,036.10	2,800,275.50	2,696,185.40	1.09	2.67
1/5/2024	2,921,047.70	2,758,294.10	2,591,427.60	5.57	11.28

1/6/2024	3,727,583.80	3,465,227.10	3,377,617.00	7.04	9.39
1/7/2024	4,777,111.70	4,218,401.10	4,323,536.80	11.7	9.49
1/8/2024	4,559,641.70	4,297,404.60	4,422,505.40	5.75	3.01
1/9/2024	3,932,698.50	3,645,983.80	3,879,469.60	7.29	1.35
1/10/2024	2,943,483.70	2,824,181.20	2,755,658.40	4.05	6.38
1/11/2024	2,546,338.50	2,789,203.60	2,675,477.20	9.54	5.07
1/12/2024	4,633,223.30	4,182,870.70	4,287,450.10	9.72	7.46

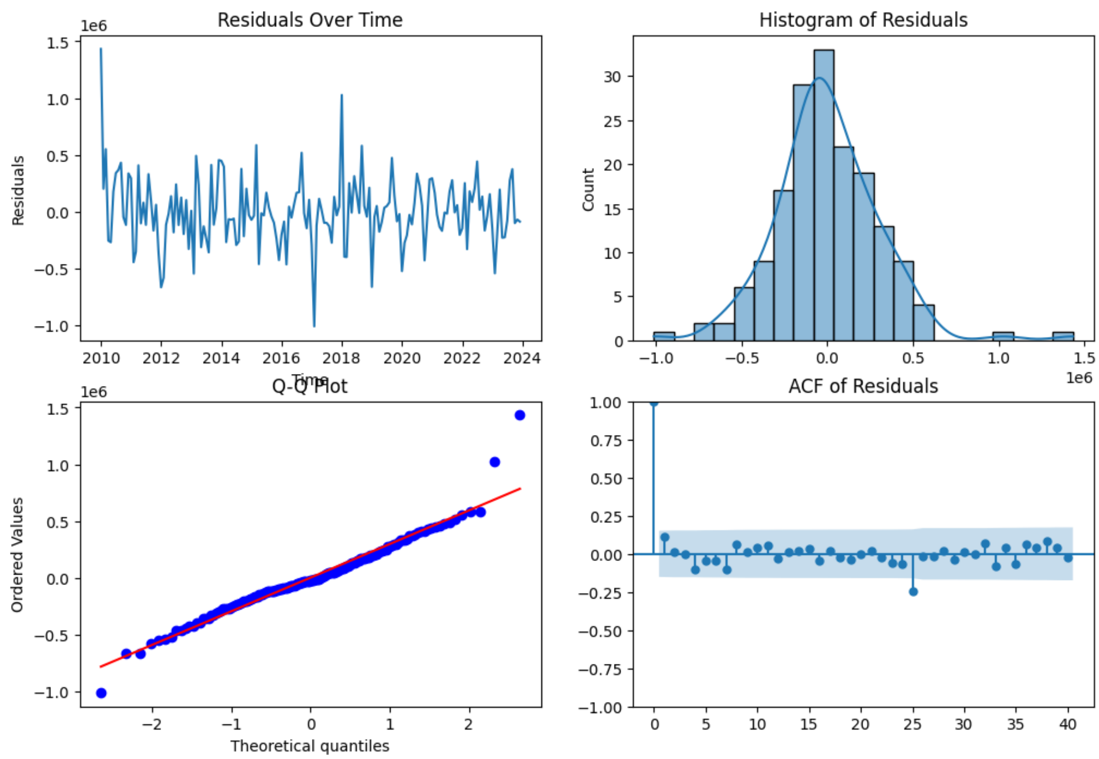
In 2024, SARIMAX and ARIMA models were used to forecast monthly energy demand across Tennessee's residential, commercial, industrial, and combined sectors. SARIMAX consistently outperformed ARIMA, delivering lower average errors in all sectors.

For residential, SARIMAX had a 6.10% average error versus ARIMA's 6.75%, providing more stable forecasts. Commercial forecasts were the most accurate, with SARIMAX averaging about 2% error. Industrial sector errors were around 4.1% with SARIMAX, lower than ARIMA. The combined sector showed SARIMAX errors below 3%, reflecting strong overall accuracy.

Both models saw higher errors in months with seasonal variability, but SARIMAX's use of external factors like temperature improved its performance throughout the year. For the remaining sectors, please refer to the Appendix B (Table 1-3).

Model Evaluation Metrics (Residential)		
Metric	ARIMA	SARIMAX
MAE	255,290.68	210,321.49
RMSE	291,811.84	257,878.30
MAPE %	6.82	5.69

SARIMAX outperformed ARIMA in every sector with lower MAE, RMSE, and MAPE. In the residential sector, MAE decreased by 18% and MAPE dropped from 6.82% to 5.69%. The commercial sector had the largest improvement, with MAE down 41% and MAPE from 3.5% to 2.02%. Industrial results also improved, with MAE reduced by 50% and MAPE from 8.14% to 4.10%. In the combined sector, SARIMAX lowered MAE from 440,419 to 255,197 and MAPE from 4.93% to 2.89%.



- The residuals are centered around zero with no discernible patterns.
- The histogram of residuals appears roughly bell-shaped, suggesting normality.
- The Q-Q plot shows residuals falling mostly along the 45° line.
- The autocorrelation function (ACF) of the residuals displays no significant spikes, supporting the Ljung-Box test result and indicating no strong autocorrelation.

Key Findings

Summary of Results

The analysis revealed several key insights regarding energy consumption across different sectors in Tennessee:

- **Residential & Commercial Sectors**: Energy consumption shows a clear seasonal pattern. The SARIMAX model successfully forecasted the 2024 demand values with minimal deviations from the actual observed data.
- **Industrial Sector**: ARIMA underestimates the peaks, showing more deviation from actual test data. SARIMAX, with its seasonal adjustments, captures the trend and fluctuates better, although not perfectly.
- **Transportation Sector**: Most data for this sector were missing, and therefore, no analysis or forecasting was conducted for transportation energy demand.
- **Model Performance**: SARIMAX outperforms ARIMA in forecasting energy demand for the Combined Sector, with a lower MAE of approximately 293,000 kWh compared to 445,000 kWh for ARIMA. It also shows better performance in RMSE (352,000 kWh vs. 508,000 kWh) and MAPE (3.3% vs. 5.0%). These results clearly indicate that SARIMAX is a more accurate and reliable model for sectors with strong seasonal energy usage patterns.

Interpretation of Findings

- **Residential & Commercial Sector Insights**: The strong correlation between residential and commercial energy consumption and temperature confirms the hypothesis that HVAC usage is a dominant driver of energy demand in this sector. The seasonal fluctuations are well captured by the SARIMAX model, providing a reliable forecasting tool for utility companies. This information is valuable for utility companies in predicting energy demand during peak seasons.
- **Industrial Sector Trends**: The steady decline in industrial energy consumption, especially post-2014, suggests a shift in industry practices or the adoption of energy-efficient technologies that are not influenced by weather. Understanding these trends is important for policy makers and utility

providers to account for changes in industrial consumption and to adjust capacity planning accordingly.

- **Transportation Data Gaps**: The lack of sufficient data for the transportation sector highlights an area for improvement in data collection and monitoring. If data were more complete, this sector could provide further insights into its relationship with energy consumption and temperature.
- **Model Effectiveness**: The use of SARIMAX for forecasting energy demand proved effective in capturing the time-dependent patterns in the data. The low RMSE and well-behaved residuals indicate that the model has predictive value and can be used for short-term forecasting. This can aid in strategic planning for energy distribution and usage management.

Sector	RMSE (SARIMAX)	Avg Actual Consumption	Relative RMSE (%)
Residential	257,878.30	3,723,053	7.32%
Commercial	91,846.85	3,080,864	3.27%
Industrial	95,338.05	1,848,551	4.81%
Combined	323,096.46	8,579,765	3.89%

- The **Commercial sector** has the lowest relative RMSE (~3.3% with SARIMAX), meaning its forecasts are the most accurate relative to typical consumption levels. This suggests the model captures commercial consumption patterns very well.
- The **Residential sector** has the highest relative RMSE (~7.3% with SARIMAX), indicating greater difficulty in forecasting residential energy use accurately. This likely reflects higher variability or complexity in household consumption behaviors.
- The **Industrial sector** has a moderate relative RMSE (~4.8% with SARIMAX), showing fairly accurate forecasts but with more errors than Commercial.
- The **Combined sector**, aggregating all sectors, shows a relative RMSE of about 3.9% with SARIMAX, reflecting solid overall forecast accuracy that benefits from averaging across sectors.

Recommendations

Based on the findings from this analysis, I suggest the following practical actions for stakeholders in Tennessee's energy sector, particularly utility companies and policy makers:

1. **Customer Alerts for High Demand and Peak Hours**
    - **Recommendation**: Implement text message alerts to customers during high demand periods or peak hours. By notifying customers about the upcoming peak times, they can adjust their energy usage, potentially leading to a reduction in overall consumption.
2. **Incentive Programs for Energy-Efficient Practices**
    - **Recommendation**: Develop and promote programs that incentivize households to adopt energy-efficient appliances and practices. These programs could be tied to off-peak hours, encouraging consumers to use electricity more efficiently and reduce their overall consumption.
3. **Focus on Industrial Sector Efficiency**
    - **Recommendation**: Encourage industries to adopt more energy-efficient technologies through tax incentives or grants. Also, support the continued shift towards sustainability initiatives that reduce industrial energy consumption irrespective of temperature.
4. **Improve Data Collection for Transportation Sector**
    - **Recommendation**: Prioritize efforts to improve data collection for the transportation sector to better understand its relationship with energy consumption. This may include gathering more granular data on energy use across different transportation modes (e.g., electric vehicles, commercial fleets) and its dependence on temperature.

Practical Implications

These recommendations have several real-world applications:

- **Demand Management**: By alerting customers to peak demand times, utilities can better manage demand, reduce the strain on energy infrastructure.

- **Environmental Impact**: Encouraging energy efficiency at both the residential and industrial levels could contribute to significant reductions in energy consumption and carbon emissions. This aligns with broader sustainability goals and helps meet regulatory requirements aimed at reducing the state's carbon footprint.
- **Cost Savings**: Both consumers and utilities stand to benefit from reducing energy consumption during peak periods. For consumers, it may lead to lower electricity bills, while utilities can avoid the high costs associated with energy generation during peak demand.

Risks and Mitigation Strategies

Identified Risks

- **Data Quality:** One potential risk is the quality and completeness of the data. Missing or inconsistent data in either temperature or electricity consumption records could affect the model's accuracy. To mitigate this, we will implement comprehensive data cleaning techniques, including handling missing values through imputation and removing any outliers that could distort the model.
- **Model Overfitting/Underfitting:** With linear regression models, there's the risk of overfitting (if the model fits the training data too well) or underfitting (if the model does not capture the true patterns in the data). To address this, we will evaluate multiple models and use cross-validation techniques to ensure the selected model generalizes well to unseen data.
- **External Factors:** The model might not capture external factors such as holidays, special events, or regional economic shifts, which can influence energy consumption. These factors can lead to model bias or reduced accuracy.

Contingency Plan

If the ARIMA model does not yield satisfactory results, we will explore SARIMAX which can better capture complex relationships between temperature, kWh cost, and other influencing factors. This model offers greater flexibility and may uncover interactions that ARIMA might not capture.

Ethical Considerations

Data Privacy and Confidentiality

Since the dataset is aggregated at a state level and does not include personally identifiable information (PII), there are no data privacy concerns. However, care will be taken to ensure that the data used for the project is publicly available and adheres to ethical data usage standards.

Bias and Fairness

All assumptions made during the analysis, such as the choice of features or transformation methods, will be clearly communicated. Any conclusions derived from the model will be supported by data, and the limitations of the model (e.g., the inability to capture some external variables) will be openly discussed. This ensures that the findings are not misleading or overgeneralized.

Conclusion

Summary of Research

This study analyzed energy consumption patterns across Tennessee's sectors, emphasizing the impact of temperature on seasonal consumption, particularly in the residential sector. Forecasts for 2024 demand were developed using both ARIMA and SARIMAX models, with SARIMAX consistently outperforming ARIMA in accuracy.

SARIMAX achieved lower average errors (MAPE) across sectors: 5.69% for residential, 2.02% for commercial, 4.10% for industrial, and 2.89% combined. The relative RMSE values further highlight forecast performance, with the commercial sector showing the lowest error (~3.3%) and the residential sector the highest (~7.3%), indicating differing levels of forecasting difficulty.

A MAPE of 3% means that forecasted consumption typically deviates from actual values by only 3%, while a relative RMSE of around 4% indicates forecast errors are about 3% of average consumption. Both metrics demonstrate strong predictive accuracy.

These findings offer valuable insights for utilities and policymakers to improve consumption forecasting, optimize resource allocation by sector.

Future Research Directions

Future work in this area could address several key areas for further investigation:

1. **Data Expansion for the Transportation Sector**: Collecting and analyzing more comprehensive data on transportation energy consumption, particularly as electric vehicles become more widespread, would provide valuable insights into how temperature impacts energy usage in this sector.
2. **Incorporating Economic and Technological Factors**: Future models could integrate additional variables, such as economic indicators (e.g., GDP growth, unemployment rates) and advancements in energy-efficient technologies, to better understand their impact on energy consumption across sectors.
3. **Long-Term Forecasting**: Extending the forecasting horizon beyond 2024 to assess long-term trends and the potential effects of climate change.

References

U.S. Energy Information Administration (EIA). (2024). Monthly Energy Consumption Data. Link

National Centers for Environmental Information (NOAA). (2024). Monthly Average Temperature Data. Link

Appendices

Appendix A: Graphs

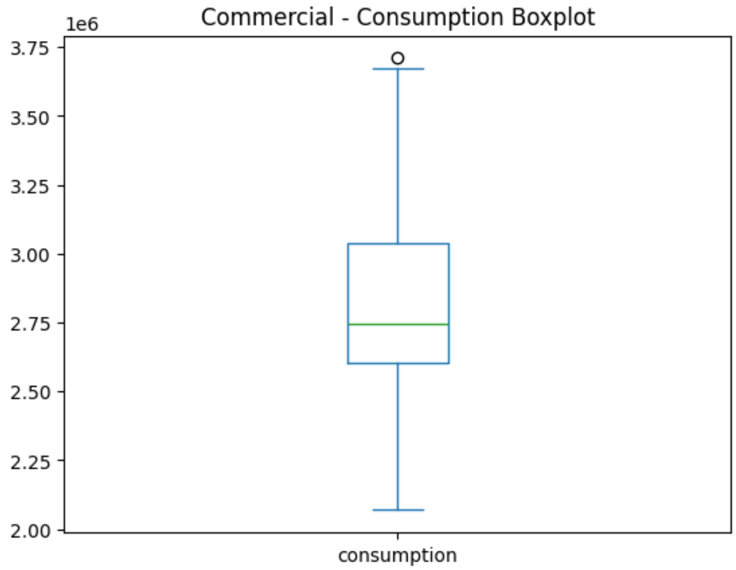


Figure 1: Box Plot (Commercial Sector)

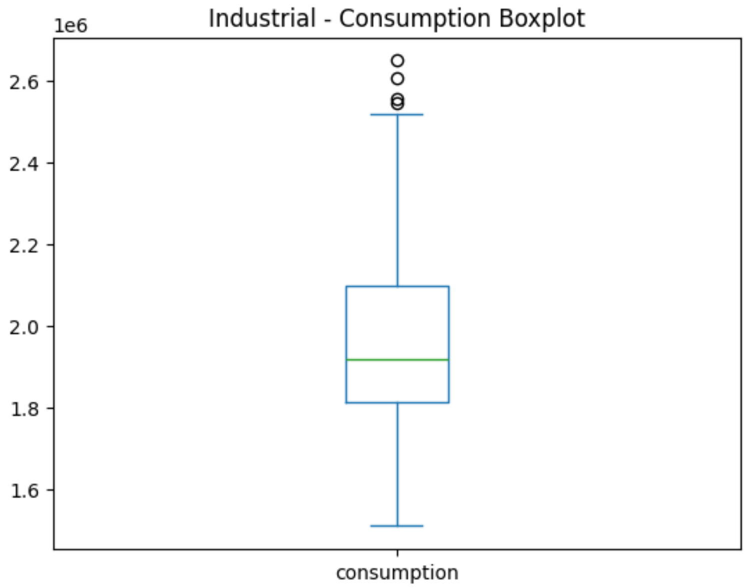


Figure 2: Box Plot (Industrial Sector)

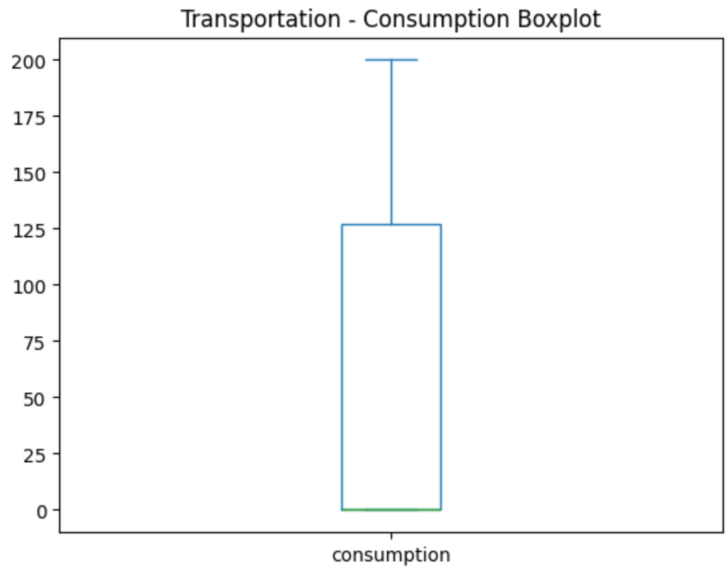


Figure 3: Box Plot (Transportation Sector)

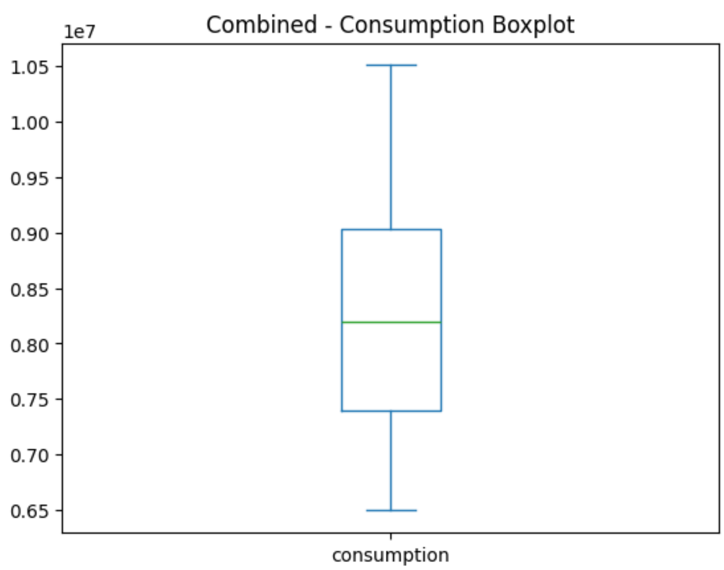


Figure 4: Correlation Matrix (All Sectors)

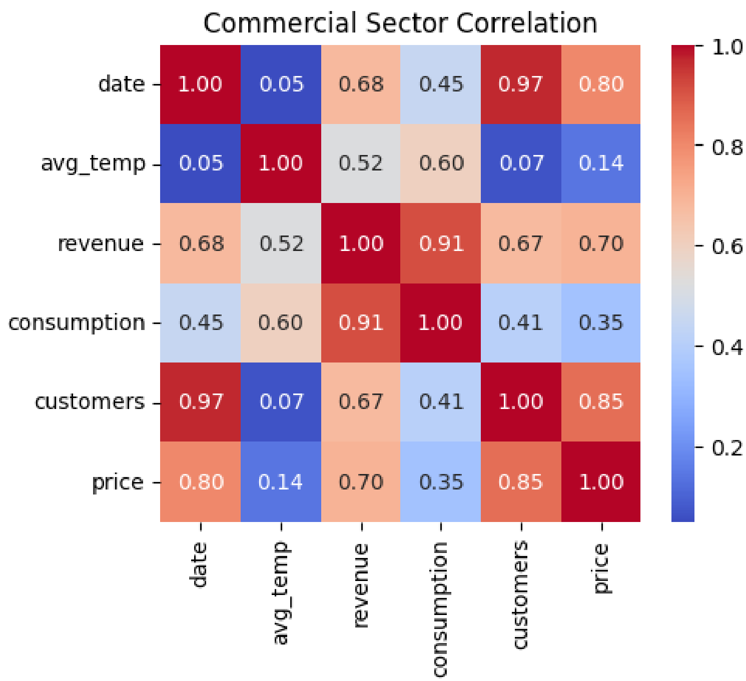


Figure 5: Correlation Matrix (Commercial Sector)

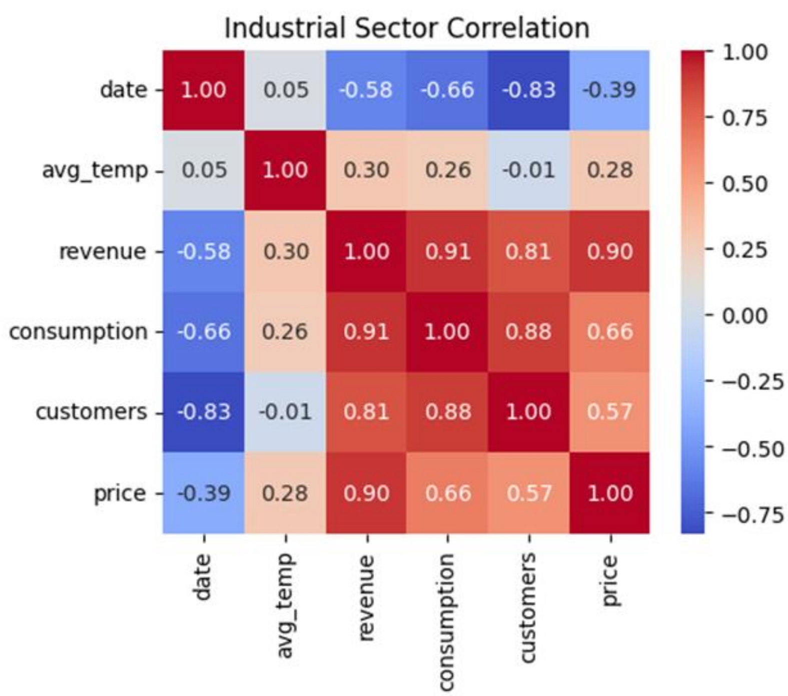


Figure 6: Correlation Matrix (Industrial Sector)

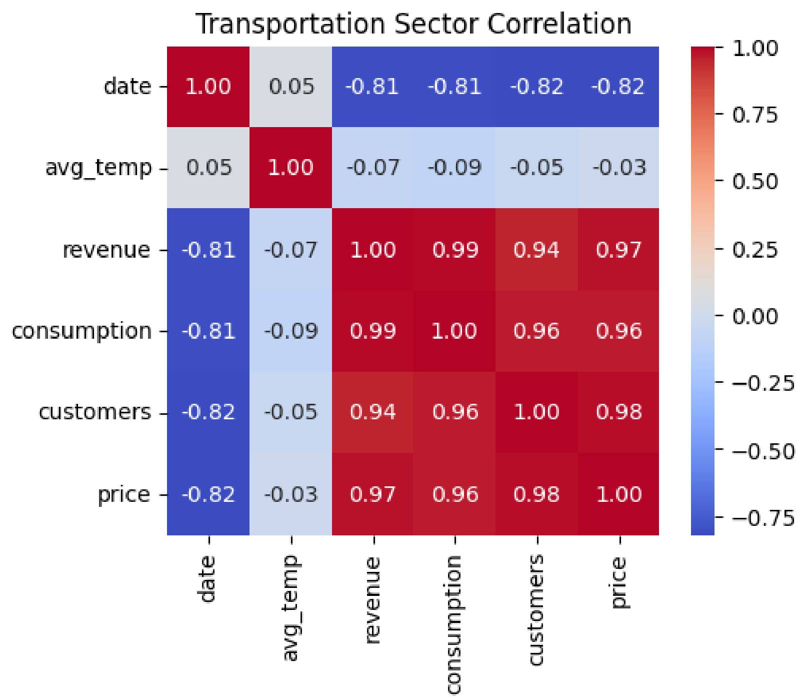


Figure 7: Correlation Matrix (Transportation Sector)

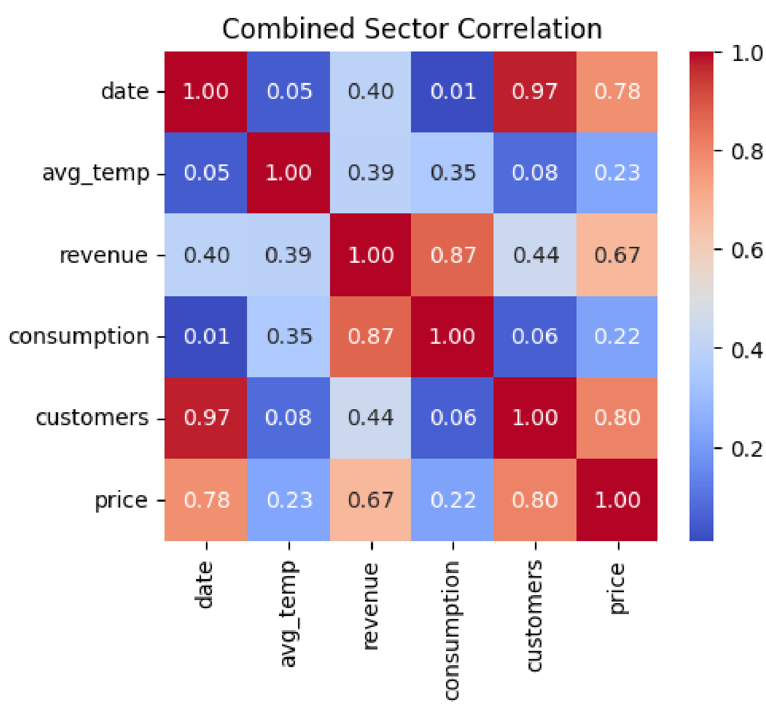


Figure 8: Correlation Matrix (All Sectors)

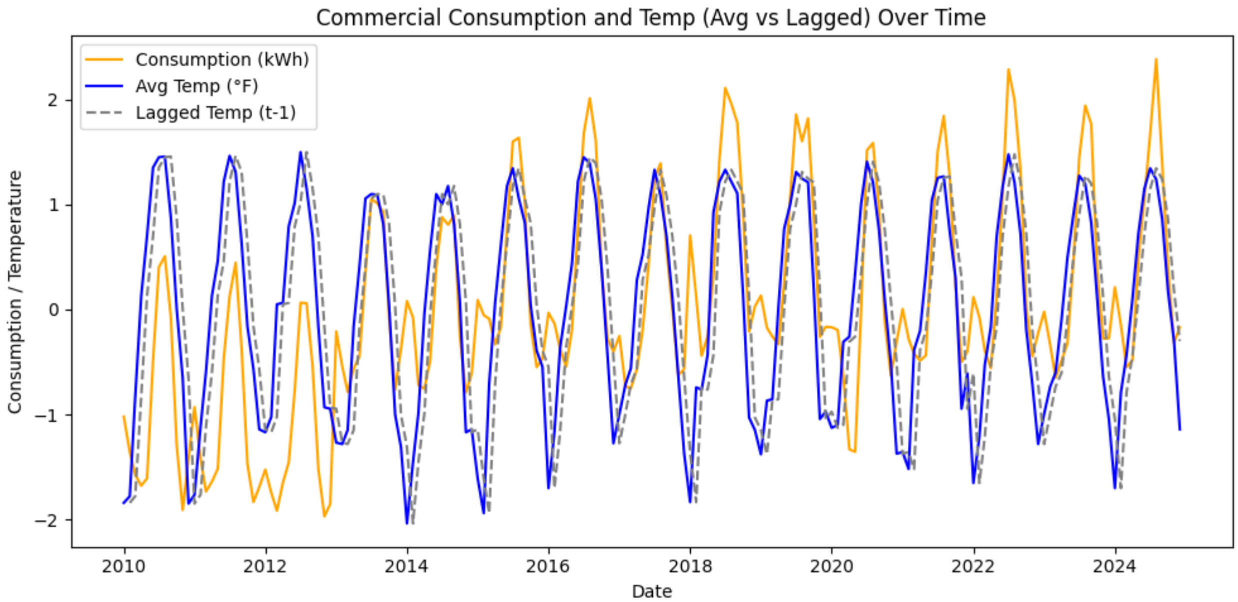


Figure 9: Time Series (Commercial Sector)

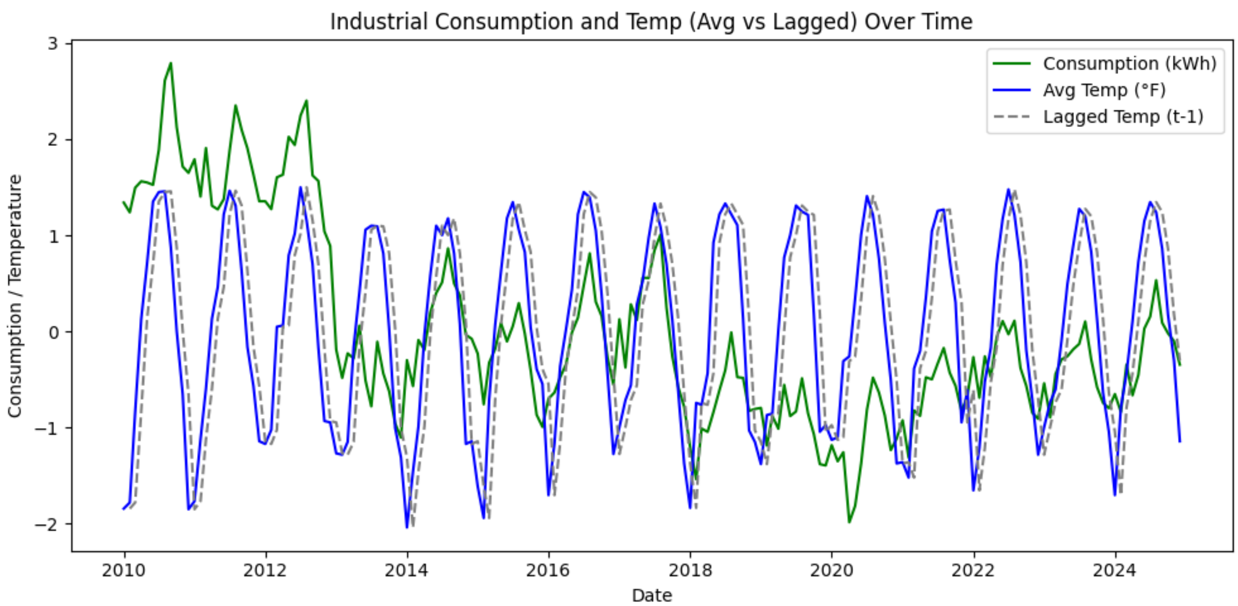


Figure 10: Time Series (Industrial Sector)

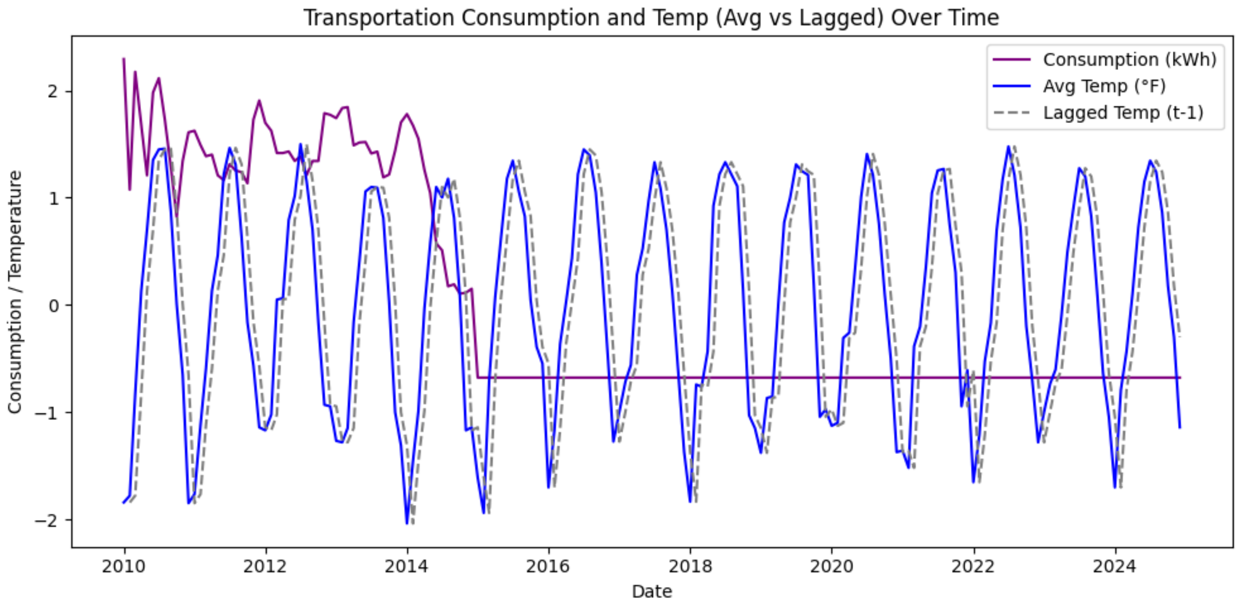


Figure 11: Time Series (Transportation Sector)

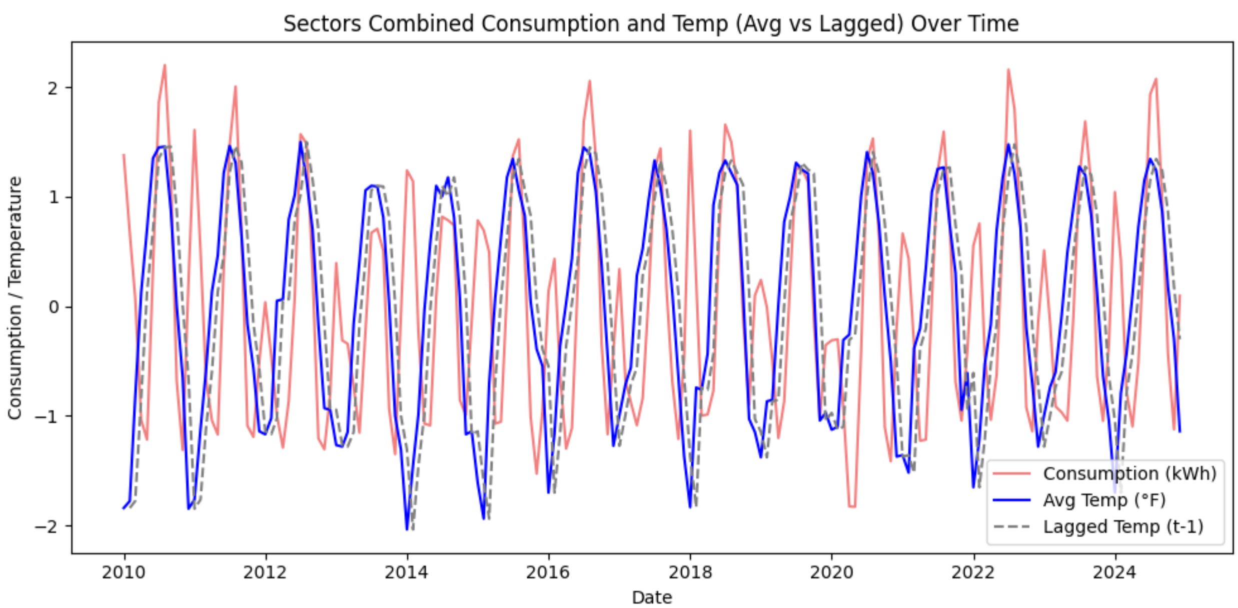


Figure 12: Time Series (All Sectors)

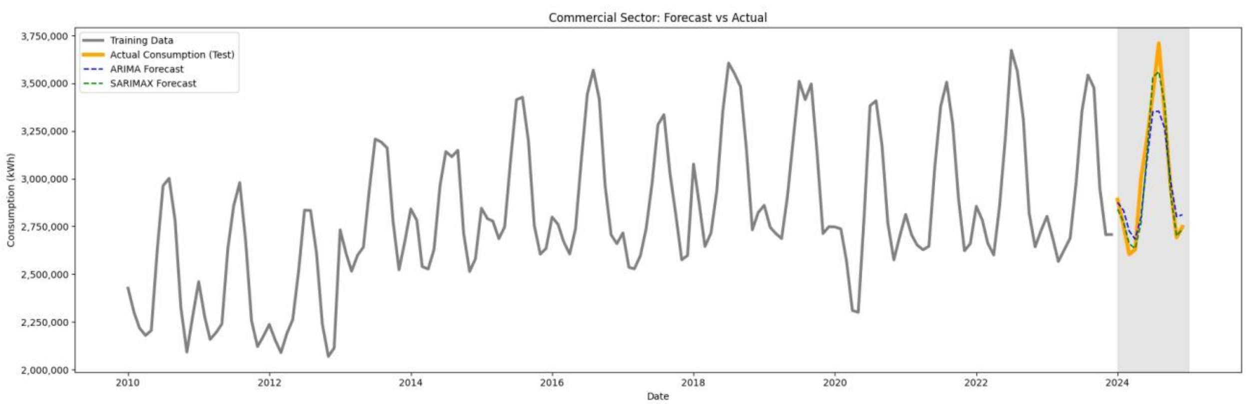


Figure 13: Forecast vs. Actual (Commercial Sector)

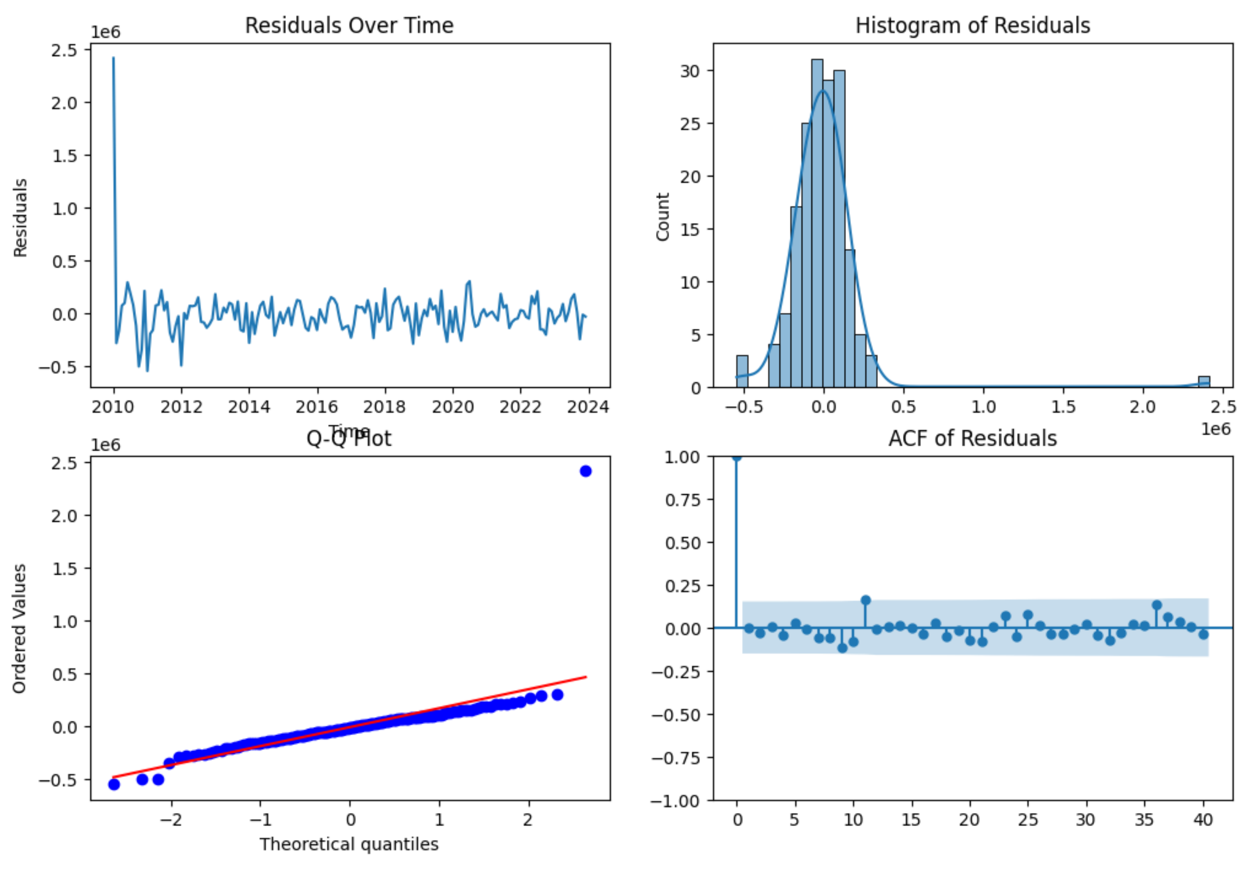


Figure 14: Residual Diagnostic Plots (Commercial Sector)

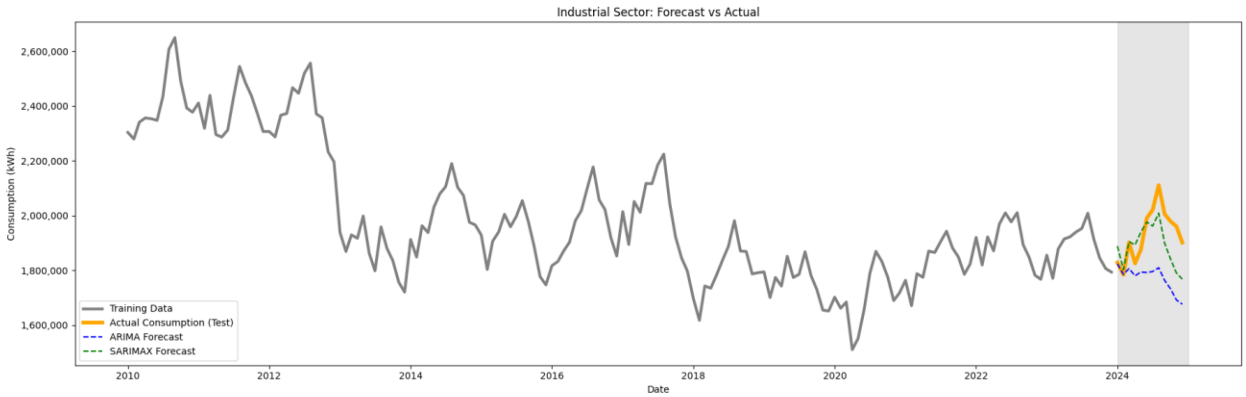


Figure 15: Forecast vs Actual (Industrial Sector)

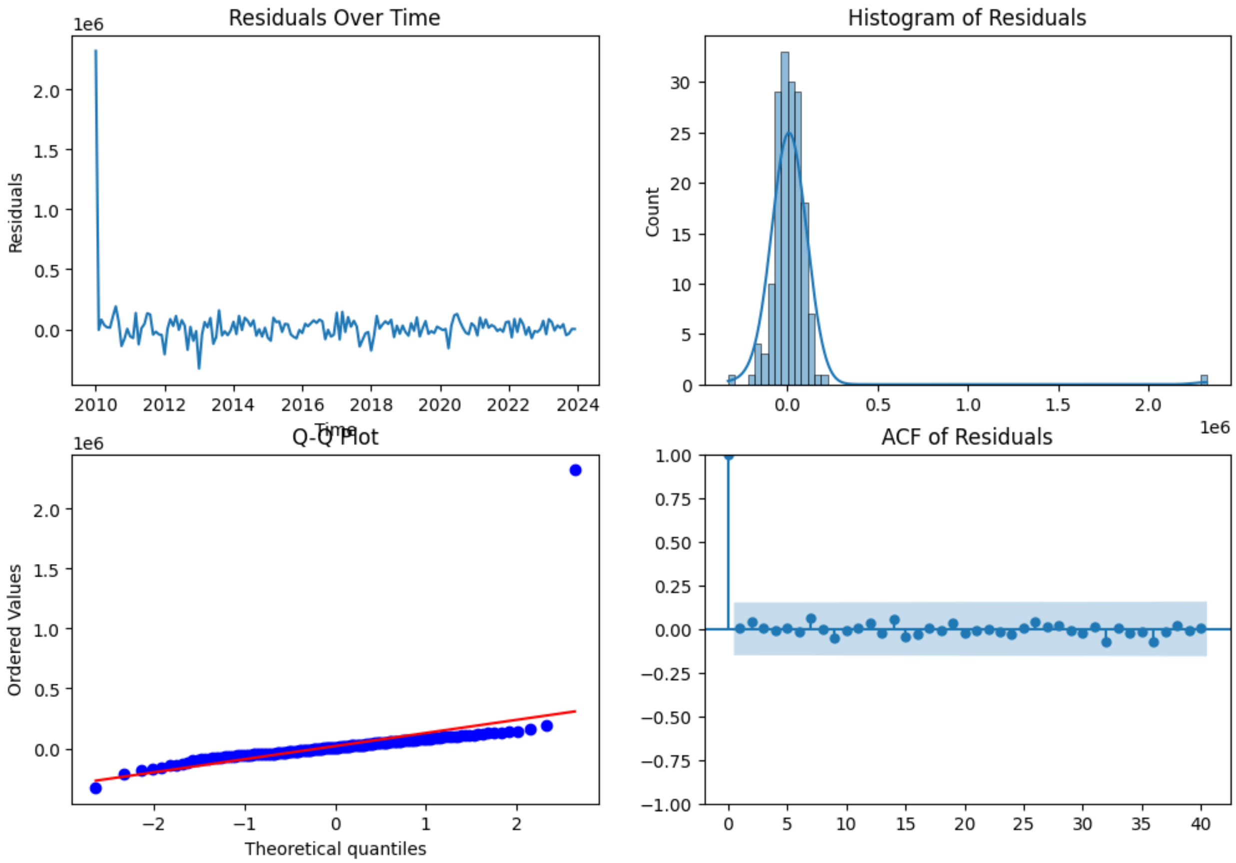


Figure 16: Residual Diagnostic Plots (Industrial Sector)

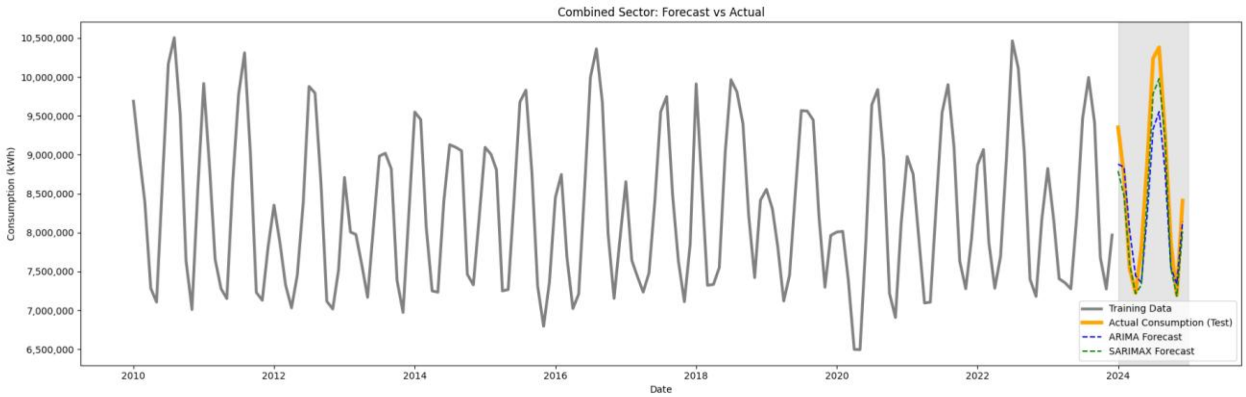


Figure 17: Forecast vs Actual (All Sectors)

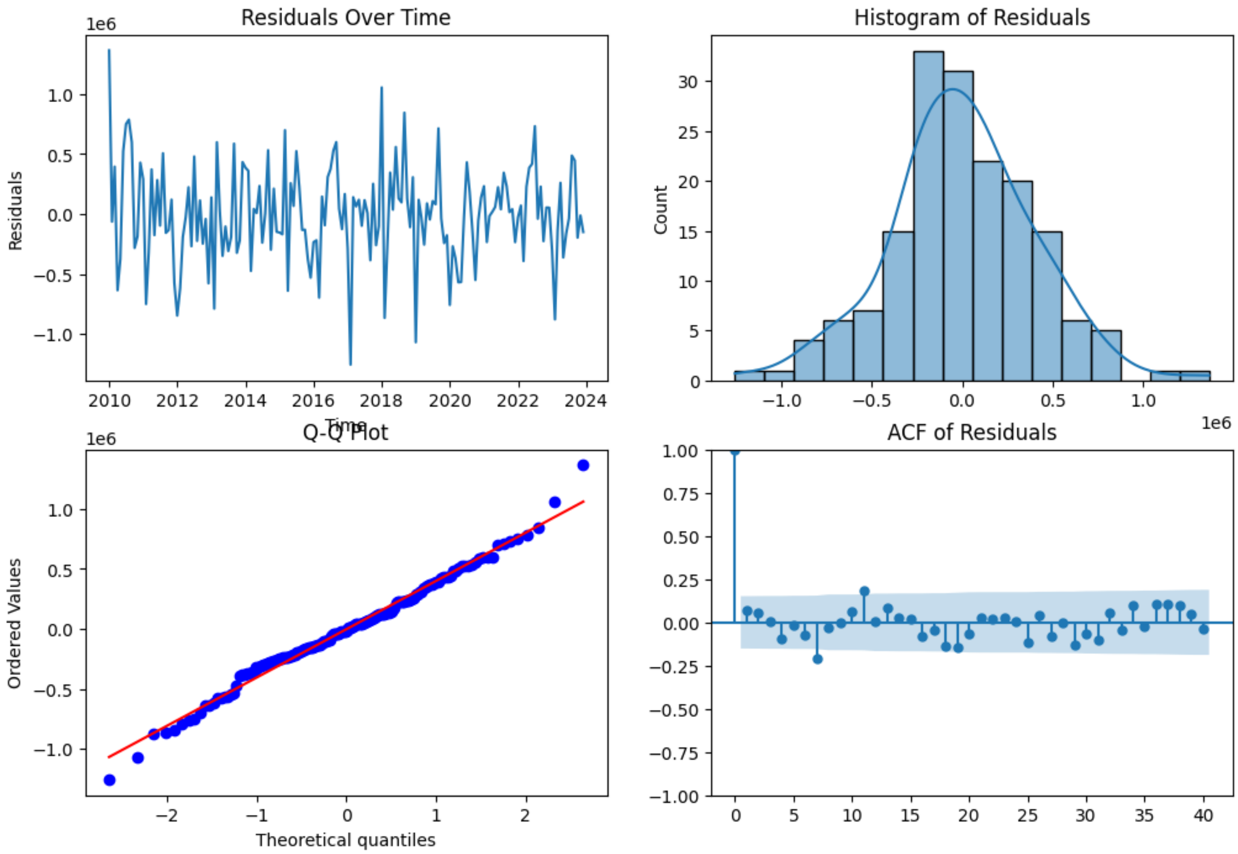


Figure 18: Residual Diagnostic Plots (All Sectors)

Appendix B: Tables

Forecast vs Actual Consumption (Commercial)					
Month	Actual Consumption	ARIMA Forecast	SARIMAX Forecast	ARIMA Error %	SARIMAX Error %
1/1/2024	2,890,816.10	2,875,675.30	2,852,474.10	0.52	1.33
1/2/2024	2,755,125.20	2,833,852.60	2,764,548.70	2.86	0.34
1/3/2024	2,604,083.00	2,726,136.10	2,655,001.10	4.69	1.96
1/4/2024	2,629,362.50	2,682,896.50	2,627,733.60	2.04	0.06
1/5/2024	2,999,197.30	2,797,577.50	2,762,486.80	6.72	7.89
1/6/2024	3,208,536.90	3,098,091.40	3,126,065.70	3.44	2.57
1/7/2024	3,441,162.70	3,350,968.20	3,530,284.20	2.62	2.59
1/8/2024	3,709,786.00	3,353,286.50	3,562,501.60	9.61	3.97
1/9/2024	3,334,835.00	3,262,416.20	3,385,395.80	2.17	1.52
1/10/2024	2,957,963.90	2,986,210.60	2,913,788.90	0.95	1.49
1/11/2024	2,692,088.70	2,801,359.10	2,687,995.70	4.06	0.15
1/12/2024	2,747,414.40	2,811,151.20	2,738,324.30	2.32	0.33

Table 1: Forecast vs Actual Consumption (Commercial)

Forecast vs Actual Consumption (Industrial)					
Month	Actual Consumption	ARIMA Forecast	SARIMAX Forecast	ARIMA Error %	SARIMAX Error %
1/1/2024	1,828,454.60	1,823,752.10	1,901,214.80	0.26	3.98
1/2/2024	1,783,948.40	1,785,181.00	1,805,042.20	0.07	1.18
1/3/2024	1,901,566.50	1,806,425.00	1,901,773.70	5	0.01
1/4/2024	1,825,348.10	1,779,751.00	1,888,960.70	2.5	3.48
1/5/2024	1,875,659.70	1,793,714.90	1,937,374.60	4.37	3.29
1/6/2024	1,991,040.60	1,792,202.60	1,974,742.70	9.99	0.82
1/7/2024	2,020,591.00	1,795,613.80	1,961,381.10	11.13	2.93
1/8/2024	2,110,735.90	1,809,472.30	2,008,638.00	14.27	4.84
1/9/2024	2,005,230.00	1,762,950.60	1,897,356.00	12.08	5.38
1/10/2024	1,979,181.20	1,732,929.10	1,838,474.60	12.44	7.11
1/11/2024	1,959,803.10	1,691,621.10	1,783,794.50	13.68	8.98
1/12/2024	1,901,258.80	1,676,166.30	1,763,897.20	11.84	7.22

Table 2: Forecast vs Actual Consumption (Industrial)

Forecast vs Actual Consumption (Combined)					
Month	Actual Consumption	ARIMA Forecast	SARIMAX Forecast	ARIMA Error %	SARIMAX Error %
1/1/2024	9,352,494.00	8,930,525.80	9,034,740.50	4.51	3.4
1/2/2024	8,733,380.20	8,891,068.30	8,681,236.60	1.81	0.6
1/3/2024	7,542,792.10	7,952,452.30	7,564,846.90	5.43	0.29
1/4/2024	7,224,746.70	7,288,638.80	7,239,877.40	0.88	0.21
1/5/2024	7,795,904.70	7,288,240.00	7,311,697.20	6.51	6.21
1/6/2024	8,927,161.30	8,304,605.90	8,450,109.80	6.97	5.34
1/7/2024	10,238,865.40	9,411,246.80	9,800,883.40	8.08	4.28
1/8/2024	10,380,163.60	9,517,293.40	10,022,274.20	8.31	3.45
1/9/2024	9,272,763.50	8,628,724.70	9,224,991.50	6.95	0.52
1/10/2024	7,880,628.80	7,386,111.10	7,543,082.90	6.28	4.28
1/11/2024	7,198,230.30	7,273,906.60	7,185,356.70	1.05	0.18
1/12/2024	8,411,253.60	8,214,372.20	7,911,292.50	2.34	5.94

Table 3: Forecast vs Actual Consumption (Combined)

Model Evaluation Metrics (Commercial)		
Metric	ARIMA	SARIMAX
MAE	108,490.62	63,651.59
RMSE	139,735.03	91,846.85
MAPE %	3.5	2.02

Table 4: Model Evaluation Metrics (Commercial Sector)

Model Evaluation Metrics (Industrial)		
Metric	ARIMA	SARIMAX
MAE	161,291.95	79,912.10
RMSE	191,639.47	95,338.05
MAPE %	8.14	4.10

Table 5: Model Evaluation Metrics (Industrial Sector)

Model Evaluation Metrics (Combined)		
Metric	ARIMA	SARIMAX
MAE	440,419.31	255,197.14
RMSE	512,627.68	323,096.46
MAPE %	4.93	2.89

Table 5: Model Evaluation Metrics (Combined Sector)

Appendix C: Code

```
import numpy as np
import pandas as pd
import seaborn as sns
import scipy.stats as stats
import matplotlib.pyplot as plt
import matplotlib.ticker as ticker
from statsmodels.tsa.arima.model import ARIMA
from statsmodels.graphics.tsaplots import plot_acf
from statsmodels.stats.diagnostic import acorr_ljungbox
from pmdarima import auto_arima
from sklearn.metrics import mean_squared_error, mean_absolute_error, mean_absolute_percentage_error,
root_mean_squared_error
from sklearn.preprocessing import StandardScaler

import warnings
from pandas.errors import SettingWithCopyWarning
warnings.simplefilter(action='ignore', category=SettingWithCopyWarning)
```

Code 1: Import Libraries

```
url_1 = (r"C:\Users\Hamza\OneDrive\Documents\Data Science\Masters\6. Spring 2025\DSC630-T302 "
        r"Predictive Analytics (2255-1)\Weekly Assignments & Material\Week 9\HS861M 2010-.xlsx")

url_2 = (r"C:\Users\Hamza\OneDrive\Documents\Data Science\Masters\6. Spring 2025\DSC630-T302 "
        r"Predictive Analytics (2255-1)\Weekly Assignments & Material\Week 9\TN_AVG_TEMP_2010_2024.csv")

df_temp = pd.read_csv(url_2, skiprows=3)
df = pd.read_excel(url_1)
```

Code 2: Load Data

```
df.rename(columns={
    "Unnamed: 0":"year", "Unnamed: 1":"month", "Unnamed: 2":"state", "Unnamed: 3":"status",
    "RESIDENTIAL": 'res_rev', 'Unnamed: 5': 'res_sales', 'Unnamed: 6': 'res_cust', 'Unnamed: 7' : 'res_price',
    "COMMERCIAL": 'com_rev', 'Unnamed: 9': 'com_sales', 'Unnamed: 10': 'com_cust', 'Unnamed: 11' : 'com_price',
    "INDUSTRIAL": 'ind_rev', 'Unnamed: 13': 'ind_sales', 'Unnamed: 14': 'ind_cust', 'Unnamed: 15' : 'ind_price',
    "TRANSPORTATION": 'trans_rev', 'Unnamed: 17': 'trans_sales', 'Unnamed: 18': 'trans_cust', 'Unnamed: 19' : 'trans_price',
    "TOTAL": 'total_rev', 'Unnamed: 21': 'total_sales', 'Unnamed: 22': 'total_cust', 'Unnamed: 23' : 'total_price'
}, inplace=True)

df = df.drop(index=[0, 1], columns=['status'])
df.reset_index(drop=True, inplace=True)
df = df.convert_dtypes()
df = df.drop(df.index[-1])
df['date'] = pd.to_datetime(df[['year', 'month']].assign(day=1))
```

Code 3: Initial Data Cleaning

```
df_temp.rename(columns={"Date" : "date", 'Value': 'avg_temp'}, inplace=True)
df_temp['date']= df_temp['date'].astype(str)
df_temp['year'] = df_temp['date'].str[0:4]
df_temp['month'] = df_temp['date'].str[4:6]
df_temp['date'] = pd.to_datetime(df_temp[['year', 'month']].assign(day=1))
df_temp.drop(columns=['year', 'month'], inplace=True)
```

Code 4: Temperature Data Cleaning

```
df = df[df['state'] == 'TN']
df_merged = pd.merge(df_temp, df, on='date', how='inner')
```

Code 5: Filter and Merge

```
df_res = df_merged[['date', 'avg_temp', 'res_rev', 'res_sales', 'res_cust', 'res_price']]
df_com = df_merged[['date', 'avg_temp','com_rev', 'com_sales', 'com_cust', 'com_price']]
df_ind = df_merged[['date', 'avg_temp','ind_rev', 'ind_sales', 'ind_cust', 'ind_price']]
df_trans = df_merged[['date','avg_temp', 'trans_rev', 'trans_sales', 'trans_cust', 'trans_price']]
df_total = df_merged[['date', 'avg_temp','total_rev', 'total_sales', 'total_cust', 'total_price']]

sectors = [df_res, df_com, df_ind, df_trans, df_total]
for df in sectors:
    col_names = df.columns
    df.rename(columns={
        col_names[2]: 'revenue',
        col_names[3]: 'consumption',
        col_names[4]: 'customers',
        col_names[5]: 'price'
    }, inplace=True)
```

Code 6: Create Sector Subsets

```
sector_names = ["Residential", "Commercial", "Industrial", "Transportation", "Combined"]
for i, dfcor in enumerate(sectors):
    sector_naming = sector_names[i]
    print(f"{sector_naming} Sector")
    print(dfcor.describe())
    print("***"*30)
```

Code 7: Summary Statistics

```
for i, dfcor in enumerate(sectors):
    sector_naming = sector_names[i]
    print(f"{sector_naming} Sector")

    # Plot only the 'Consumption' column
    dfcor['consumption'].plot(kind='box')

    # Add a title for clarity
    plt.title(f"{sector_naming} - Consumption Boxplot")

    # Show each plot separately
    plt.show()

    q1 = dfcor['consumption'].quantile(0.25)

    q3 = dfcor['consumption'].quantile(0.75)

    iqr = q3 - q1
    outliers = dfcor[(dfcor['consumption'] < q1 - 1.5 * iqr) | (dfcor['consumption'] > q3 + 1.5 * iqr)]

    print(f"{sector_naming} Sector has {len(outliers)} outliers")

    print("***"*30)
```

Code 8: Box Plots

```
sector_names = ["Residential", "Commercial", "Industrial", "Transportation", "Combined"]

for i, dfcor in enumerate(sectors):
    sector_naming = sector_names[i]
    corr = dfcor.corr()

    plt.figure(figsize=(5, 4))
    sns.heatmap(corr, annot=True, cmap='coolwarm', fmt=".2f")
    plt.title(f"{sector_naming} Sector Correlation")
    plt.show()
```

Code 9: Correlation Heatmap

```
scaler = StandardScaler()

for df in sectors:
    df[['consumption_scaled', 'avg_temp_scaled']] = scaler.fit_transform(df[['consumption', 'avg_temp']])

colors = ["red", "orange", "green", "purple", "lightcoral"]

sector_names = ["Residential", "Commercial", "Industrial", "Transportation", "Sectors Combined"]

for df, sector_name, color in zip(sectors, sector_names, colors):
    plt.figure(figsize=(10, 4))
    plt.plot(df['date'], df['consumption_scaled'], label='Consumption (scaled)', color=color)
    plt.plot(df['date'], df['avg_temp_scaled'], label='Avg Temp (scaled)', linestyle='--', color='blue')
    plt.title(f"{sector_name} - Scaled Consumption vs. Avg Temp Over Time")
    plt.xlabel("Date")
    plt.ylabel("Scaled Values")
    plt.legend()
    plt.tight_layout()
    plt.show()
```

Code 10: Standardize and Plot Time Series

```
sectors = [df_res, df_com, df_ind, df_total]
for df in sectors:
    df.set_index("date", inplace=True)
```

Code 11: Prepare Sector DataFrames

```
x = df_total.drop(['revenue','customers', 'consumption', 'date', 'consumption_scaled', 'avg_temp_scaled'], axis = 1)
y = df_total['consumption']
X_train, X_test, y_train, y_test = train_test_split(x, y, test_size=0.1, random_state=101)
```

Code 12: Create Features & Target Variables & Training Split

```
mls = LinearRegression()
mls.fit(X_train,y_train) # fitting the model
predictions = mls.predict(X_test) # making predictions
```

Code 13: Regression Model Training & Prediction

```
# Calculate correlation of all features with 'consumption' target
# Exclude 'consumption', 'consumption_scaled', 'date', 'revenue', and 'avg_temp_scaled' from consideration
correlations = df_total.corr()['consumption'].drop([
    'consumption', 'consumption_scaled', 'date', 'revenue', 'avg_temp_scaled'])
# Sort features by absolute correlation in descending order to prioritize strongest relationships
sorted_features = correlations.abs().sort_values(ascending=False).index.tolist()
```

Code 14: Create Correlation Dataframe

```
results = []  # List to store performance metrics for different numbers of features
# Loop through feature subsets from 1 up to all sorted features
for i in range(1, len(sorted_features) + 1):
    selected_features = sorted_features[:i]  # Select top 'i' features based on correlation
    x = df_total[selected_features]          # Feature subset dataframe
    y = df_total['consumption']              # Target variable

    # Split data into train and test sets (90% train, 10% test), fixed random seed for reproducibility
    X_train, X_test, y_train, y_test = train_test_split(x, y, test_size=0.1, random_state=101)
    model = LinearRegression()     # Initialize linear regression model
    model.fit(X_train, y_train)    # Train model on training data
    preds = model.predict(X_test)  # Predict on test data

    # Calculate R-squared and RMSE on test data to evaluate model performance
    r2 = r2_score(y_test, preds)
    rmse = np.sqrt(mean_squared_error(y_test, preds))
    # Store the number of features, selected features, and performance metrics
    results.append((i, selected_features, r2, rmse))

# Print the results for each subset of features tried
for count, features, r2, rmse in results:
    print(f"{count} features | R2: {r2:.4f} | RMSE: {rmse:,.0f} | Features: {features}")
```

Code 15: MLR on Selected Features

```
sectors_training_df_sets = []

sectors_testing_df_sets = []

for i, df in enumerate(sectors):
    train_df = df[df.index < '2024-01-01']

    test_df = df[df.index >= '2024-01-01']

    sectors_training_df_sets.append(train_df)

    sectors_testing_df_sets.append(test_df)
```

Code 16: Train-Test Split for ARIMA

```
# Custom RMSE function
def root_mean_squared_error(y_true, y_pred):
    return np.sqrt(mean_squared_error(y_true, y_pred))

sector_names = ["Residential", "Commercial", "Industrial", "Combined"]

for i, dataf in enumerate(sectors_training_df_sets):
    sector_name = sector_names[i]
    print(f"Training models for {sector_name} sector:")

    test_df = sectors_testing_df_sets[i]

    # Ensure datetime index and set frequency
    dataf.index = pd.to_datetime(dataf.index).to_period('M').to_timestamp()
    dataf = dataf.asfreq('MS')

    test_df.index = pd.to_datetime(test_df.index).to_period('M').to_timestamp()
    test_df = test_df.asfreq('MS')

    # --- ARIMA model ---
    arima_model = auto_arima(
        dataf['consumption'],
        seasonal=True,
        m=12,
        stepwise=True,
        suppress_warnings=True,
        error_action='ignore'
    )
    arima_forecast = arima_model.predict(n_periods=12)
    arima_forecast_series = pd.Series(arima_forecast, index=test_df.index)

    # --- SARIMAX model ---
    sarimax_model = SARIMAX(
        dataf['consumption'],
        exog=dataf[['avg_temp_lag_1']],
        order=(1, 1, 1),
        seasonal_order=(1, 1, 1, 12),
        enforce_stationarity=False,
        enforce_invertibility=False
    )
    sarimax_results = sarimax_model.fit(disp=False)
    sarimax_forecast = sarimax_results.predict(
        start=len(dataf),
        end=len(dataf) + len(test_df) - 1,
        exog=test_df[['avg_temp_lag_1']]
    )
    sarimax_forecast.index = test_df.index

    # Create comparison DataFrame
    comparison_df = pd.DataFrame({
        'Month': test_df.index,
        'Actual Consumption': test_df['consumption'].values,
        'ARIMA Forecast': arima_forecast_series.values,
        'SARIMAX Forecast': sarimax_forecast.values
    })

    # Calculate Errors
    comparison_df['ARIMA Error %'] = (
        abs(comparison_df['ARIMA Forecast'] - comparison_df['Actual Consumption']) / comparison_df['Actual Consumption'] *
100
    )
    comparison_df['SARIMAX Error %'] = (
        abs(comparison_df['SARIMAX Forecast'] - comparison_df['Actual Consumption']) / comparison_df['Actual
Consumption'] * 100 )
    # Format output
    for col in ['Actual Consumption', 'ARIMA Forecast', 'SARIMAX Forecast']:
```

```
    comparison_df[col] = comparison_df[col].map('{:,.1f}'.format)
comparison_df['ARIMA Error %'] = comparison_df['ARIMA Error %'].map('{:.2f}'.format)
comparison_df['SARIMAX Error %'] = comparison_df['SARIMAX Error %'].map('{:.2f}'.format)

print(f"---- {sector_name} Sector: Forecast Comparison ----")
print(comparison_df.to_string(index=False))
print("\n" + "="*60 + "\n")

# Evaluation Metrics
print("Model Evaluation Metrics:")
metrics_data = {
    'Metric': ['MAE', 'RMSE', 'MAPE %'],
    'ARIMA': [
        mean_absolute_error(test_df['consumption'], arima_forecast),
        root_mean_squared_error(test_df['consumption'], arima_forecast),
        mean_absolute_percentage_error(test_df['consumption'], arima_forecast) * 100
    ],
    'SARIMAX': [
        mean_absolute_error(test_df['consumption'], sarimax_forecast),
        root_mean_squared_error(test_df['consumption'], sarimax_forecast),
        mean_absolute_percentage_error(test_df['consumption'], sarimax_forecast) * 100
    ]
}
metrics_df = pd.DataFrame(metrics_data)
metrics_df['ARIMA'] = metrics_df['ARIMA'].map(lambda x: f"{x:,.2f}")
metrics_df['SARIMAX'] = metrics_df['SARIMAX'].map(lambda x: f"{x:,.2f}")
print(metrics_df.to_string(index=False))
print("\n" + "="*60 + "\n")

# Plotting: Full chart and zoom-in on 2024
fig, axs = plt.subplots(2, 1, figsize=(18, 10), sharey=True, constrained_layout=True)

# --- Top chart: yearly x-axis ---
axs[0].plot(dataf.index, dataf['consumption'], label='Training Data', color='grey', linewidth=3)
axs[0].plot(test_df.index, test_df['consumption'], label='Actual Consumption (Test)', color='orange', linewidth=4)
axs[0].plot(test_df.index, arima_forecast_series, label='ARIMA Forecast', color='blue', linestyle='--')
axs[0].plot(test_df.index, sarimax_forecast, label='SARIMAX Forecast', color='green', linestyle='--')
axs[0].axvspan(test_df.index[0], test_df.index[-1], color='grey', alpha=0.2)
axs[0].set_title(f"{sector_name} Sector: Forecast vs Actual (Full Range)")
axs[0].set_ylabel("Consumption (kWh)")
axs[0].legend()
axs[0].xaxis.set_major_locator(mdates.YearLocator())
axs[0].xaxis.set_major_formatter(mdates.DateFormatter('%Y'))

# --- Zoom-in chart for 2024 with Month names only ---
axs[1].plot(dataf.index, dataf['consumption'], label='Training Data', color='grey', linewidth=3)
axs[1].plot(test_df.index, test_df['consumption'], label='Actual Consumption (Test)', color='orange', linewidth=4)
axs[1].plot(test_df.index, arima_forecast_series, label='ARIMA Forecast', color='blue', linestyle='--')
axs[1].plot(test_df.index, sarimax_forecast, label='SARIMAX Forecast', color='green', linestyle='--')
axs[1].axvspan(pd.Timestamp("2024-01-01"), pd.Timestamp("2024-12-31"), color='grey', alpha=0.2)
axs[1].set_xlim(pd.Timestamp("2024-01-01"), pd.Timestamp("2024-12-31"))
axs[1].set_title(f"{sector_name} Sector: Forecast vs Actual (Zoomed on 2024)")
axs[1].set_ylabel("Consumption (kWh)")
axs[1].xaxis.set_major_locator(mdates.MonthLocator(interval=1))
axs[1].xaxis.set_major_formatter(mdates.DateFormatter('%B'))

# Common formatting
for ax in axs:
    ax.yaxis.set_major_formatter(ticker.FuncFormatter(lambda x, _: f'{x:,.0f}'))

plt.show()
print("\n" + "#"*80 + "\n")
```

Code 17: Modeling, Forecasting, and Evaluation