Power City, USA Energy Forecasting Technical Report

Dhaval Delvadia (lead), Lili Booher, Sidney Fox, and Sierra Sellman

Introduction/Problem Statement

The fictional Power City, USA wants to implement renewable wind turbine and solar array energy but needs better insight on renewable production capacity and the City's energy consumption. The goal of this project is to apply advanced learning techniques to accurately model energy consumption and production to buy down the risk of load management for the city. To accurately forecast energy needs for a given day and hour, the models need to understand weather conditions which impact wind and solar production and the ebbs and flows of the population's energy consumption based on age demographics and eight sectors (food service, healthcare, K-12 schools, lodging, office, residential, grocery, stand-alone retail), to include increased demand from charging electric vehicles. The analysis was focused on developing, training, and testing dozens of machine learning methods to identify the most performant models. The most accurate models were then executed against the hold-out scenario year to model six days: March 15th, June 26th, July 3rd, October 13th, November 19th, and December 25th. These dates were most likely selected to represent a range of challenges within the datasets; seasonal changes, impact of holiday's, and nuances of various consumption sectors.

Data Preprocessing

Data for this project was contained within 14 comma separated files, Table 1 in the Appendix provides a summary of each file such as description, timespan of dataset, number of variables, and if the files contained duplicate or missing values. The 14 raw files needed to be joined together to create four distinct datasets to be used for analysis and modeling: 1) Solar 2) wind, 3) consumption, and 4) scenario dataset. The scenario dataset was created by joining powercity_weather_scenario and calendar_days_scenario files together. The wind, solar, and consumption datasets will serve as the training and testing data used to create the models, while the scenario dataset will be used to validate the final models and provide forecasts for the challenge scenario year. Within each dataset, the csv files need to be joined and data anomalies (missing values, skewness, gaps in time series, duplicates, outliers, etc.) need to be resolved, the following sections provide details on the steps taken.

Solar Dataset Preprocessing

The three files (solarray_weather.csv, solararray_solarangle.csv, and solararray_production.csv) were merged to create the solar production dataset. The challenge

with this dataset was the number of missing values across the various variables. Most notably, the solarray_production.csv was missing 22,618 entries when compared to the timespan of the dataset (January 2010 - August 2014). This was due to the fact that during night time hours, when the electricity production is zero and those entries were not recorded. The attributes that were missing values included dew point, humidity fraction, precipitation, pressure, temperature, visibility, wind speed, and solar elevation features. Table 8 in the Appendix provides a summary of the number of missing entries for each variable. To resolve missing values for dew point, humidity fraction, precipitation, pressure, temperature, visibility, wind speed, and solar elevation, values were grouped by month, day, hour and replaced by its mean. The cloud cover fraction variable was replaced with the median since it was categorical variable that had one of four values. The Electricity (KW/Hr) production target variable specifically any missing nighttime values were replaced with zero, while daytime values were replaced with the mean value. Once missing values were resolved, 20,546 entries had zero production and 20,776 entries had electricity values greater than zero. Outliers were identified for most dependent variables as well as target variable, but the decision was made to keep all outliers since they may contain significance on the model.

Wind Dataset Preprocessing

The two files (windfarm_production.csv and windfarm_windspeed.csv) were merged to create the wind production dataset. Date and text formatting were needed so unique date/time values could be generated for each dataset. A third (empty) dataframe was created that was a complete sequence of all days between 25 March 2011 to 31 December 2012 in 24-hour intervals. The two. csv's were then joined to the sequence dataframe. It was confirmed that no duplicates existed, there were 169 missing values, and 15,385 observations. Although the SAS competition document mentioned that accompanying weather data was supplied, the repository did not contain weather information for the wind farm. The 169 missing values were replaced with the mean of the missing attribute. The final dataset contained 15,554 records and six variables: Year, Month, Day, Hour, Wind Speed (m/s), and Electricity production (Kw/Hr), with electricity production as the target variable.

Consumption Dataset Preprocessing

Six files (powercity_solar_angle_consumption.csv, powercity_weather_consumption.csv, powercity_consumption.csv, calendar_days_consumption.csv, power_city_population.csv, and car_charging.csv) were merged to create the consumption dataset. The target variable for consumption is a decimal value of units of kilowatt per hour per square foot (Total_Electricity_KW_SQFT). To convert the unit to kilowatts per an hour, the power_city_population.csv and sector_user_matrix.xlsx were used to determine total population and then estimate the percentage of each age demographic spread per sector to calculate total square footage for each sector (Table 3). This information was joined to the previously mentioned merged consumption dataset. The resultant dataset now has the same target as the solar and wind datasets: *Total_Electricity_KW*. In the early stages of model development, both targets (Total_Electricity_KW_SQFT and Total_Electricity_KW) were used to ensure the model could accurately predict both targets.

Scenario Dataset Preprocessing

Two files (powercity_weather_scenario.csv and calendar_days_scenario.csv) were merged to create the scenario dataset. A key variable to predict solar energy, solar elevation was not included within the scenario dataset. Thus, solar elevation was imputed using the solar dataset and calculating the median solar elevation variable by grouping by month, day, and hour. After merging the file, missing values were imputed for the Cloud_Cover_Fraction, Dew_Point, Humidity_Fraction, Precipitation, Pressure, Temperature, Visibility, and Wind_Speed. It is worth noting the final dataset does not contain any of the three target variables: wind electricity production, solar electricity production, or electricity consumption. Considering this was the dataset that was going to be used to generate forecasting for the city, no analysis or modelling was performed against this dataset. Feature engineering to impute the target variables are discussed in the validation section.

Exploratory Data Analysis

Electrical energy production and consumption are dependent on a number of variables, including weather, population, date/time, and type of day. Because of the many influencing factors, the production and consumption is volatile and changes on an hourly basis. Exploratory data analysis of each of the three datasets was performed to get a better understanding of the trends within the time series and the dependent variables.

Solar

The solar dataset contained 14 variables, out of which ten were continuous variables. These ten variables had a large variance between values as well as skewness issue. This can be seen in the summary statistic table provided under Table 4. Therefore, numerous normalization techniques, such as log, square root, Min Max Scaler, Yeo-Johnson, and Quantile, were applied to the solar data to address the skew. Before and after transformation histograms of these techniques are shown in Figure 1. A key challenge with the transformations was the large frequency of zeros present in the target variable. The reason there were almost 20,546 data points with zero energy was due to the fact that no solar energy is produced during nighttime hours. Even after applying each of these normalization techniques the target variable Electricity production (KW/h) was not normally distributed. Of the normalization techniques applied, Quantile and Yeo-Johnson Transformations performed the best and were used during the mode execution. The Yeo-Johnson transformation can be thought of as an extension of the Box-Cox transformation. It handles both positive and negative values, whereas the Box-Cox transformation only handles positive values. Both can be used to transform the data so as to improve normality. The quantile transformation makes each array in a set of arrays have the same distribution to make it normalize.

Furthermore, reviewing the correlation plot shown in Figure 2 that the strongest correlation was between Electricity_KW_HR and the Solar_Elevation. There was also multicollinearity between dewpoint and temperature. However, it was determined that it would be better to retain it for the

investigation. Additionally, reviewing the heatmap and boxplot (Figure 3) of the Electrical energy production in KW/h in month and year, it was determined that more Electrical energy was produced consistently in the month of July every year from 2010 to 2014. It was also determined that more electrical energy was produced in the months of April to October in the year of 2012. Finally, reviewing the time series plot shows in Figure 4, it can be seen that most of the features in the solar dataset shows seasonal hourly trends plotted over 2010 to 2014.

Wind

The relationship between wind speed and electricity production is non-linear, and has a sigmoid shape as shown in Figure 8. Representing the fact that until the wind reaches a speed of approximately 5 m/s electricity production will be zero, and that electricity production levels out at after wind speed reaches approximately 13 m/s. Analysis showed there is a monthly seasonal trend within this data, wind speed and therefore electricity production is highest during the fall and winter months and lower during the spring and summer months (Figure 9 and Figure 10). Seasonality was not as strong with the day value, as seen in the heat map within Figure 11. Additionally, there was an hourly trend, where for a given hour there was seasonality with the prior and following hour giving it a cyclic quality (Figure 12).

Month, Day, and Hour variables are cyclic in nature and a key concern when dealing with cyclical features is how the values were encoded such that it is clear to deep learning algorithms that the feature occurs in cycles. Figure 13 illustrates this problem for the Hour variable, there are jump discontinuities at the end of each day when the hour overflows from 23 to zero. After applying the sin transformation, the jagged lines are replaced with smooth, continuous curves. Therefore a sin transformation was applied with the formulas: Month = sin(2*pi*<month value>/12), Day = sin(2*pi*<day value>/31), and Hour = sin(2*pi*<hour value>/24). For example, this transformation changed the range of hour values from 0-23, to 1 to -1.

The wind dataset had outliers, and interquartile range analysis of the dataset identified 76 outliers (Figure 14); 73 of which were anomalies within the dataset where the windspeed was greater than 5 m/s but the electricity generated was zero. Three additional observations were removed due to significant influence. The three additional observations that were removed were: 1) December 13, 2011 with a windspeed of 15.3 and electricity production of 64,032, 2) December 27, 2012 with a windspeed of 12.7 and electricity production of 59,560 and 3) January 5, 2012 with a windspeed of 17.6 and electricity production of 48,728. within both the wind speed and wind production variables were identified and will be resolved based on modeling approach. The final dataset contained 15,478 records and six variables: Year, Month, Day, Hour, Wind Speed (m/s), and Electricity production (Kw/Hr), with electricity production as the target variable.

Consumption

For the electricity consumption dataset, the baseline statistics of the continuous variables - *Electricity_KW_SQFT, Solar_Elevation, Cloud_Cover_Fraction, Dew_Point, Humidity_Fraction, Precipitable_Water, Temperature,* and *Visibility* - were reviewed to determine the mean,

standard deviation, minimum value, maximum value, and median (Table 8). After creating and reviewing the summary statistics table, it was discovered that the majority of continuous variables contained a large variance between values and would require normalization, transformation, or scaling to address the skewness present within the data. A number of normalization techniques were iteratively applied to the data and the adjusted distributions were compared across all of the normalization techniques to determine which normalization technique exhibited the best distribution of the data. Z-score, arcsinh, exponential, lambert, log, orderNorm, square root, and yeo-johnson normalization techniques were applied to the continuous variables. Comparison charts can be found in Figure 27. A review of each comparison chart showed that most of the continuous variables were very receptive to normalization, and the orderNorm and yeo-johnson normalization techniques consistently exhibited near perfect bell curve distribution.

Due to the time element within the data, with month, day, and hour all present within the dataset, various plots were created to display the target variable, *Electricity_KW_SQFT*, against different time elements to determine if outliers existed within the data and if seasonality or other time specific elements directly impacted electricity consumption. Due to the presence of different sectors as another variable that could potentially impact electricity consumption, each graph was created for each sector within the data to allow for sector to sector comparisons. The sector comparison chart proved to be a particularly effective approach because it exhibited stark differences in electricity consumption by each sector.

A similar approach was taken to analyze the impact of the hour of the day (Figure 29) and the weekday (Figure 30) had on electricity consumption by sector. To summarize the differences in electricity consumption between sectors at a high level, a ridgeline plot (section XX.XX in the Appendix) was created to show changes in the distribution of the target from sector to sector.

Methods

After identifying seasonal trends, removing outliers, and normalizing and transforming values a combination of linear, non-linear, time series, and ensemble models were developed, trained, and tested on each dataset to identify the most performant model for hourly prediction. A summary of the various modeling techniques along with pros and cons are identified below, an explanation of performance metrics used, followed by specific methods applied to each dataset.

Linear Models

Numerous linear regression models were attempted, to include various regularization techniques to avoid overfitting: multivariate, ridge, lasso, adaptive lasso (AdaLasso), and elastic net regression. Ridge regression applies regularization given by the L2-norm, which minimizes the impact of irrelevant features. Lasso regression solves a linear regression model where the model is trained with the L1 prior as the regularizer, which avoids overfitting by penalizing large coefficients and will actually set them to zero if they are not relevant. AdaLasso, like Lasso regression, reduces the coefficients of features that do not contribute to the accuracy of the

model. Lasso shrinkage produces biased estimates for the large coefficients, and thus it could be suboptimal in terms of estimation risk. AdaLasso compensates for this bias by utilizing adaptive weights to penalizing select coefficients.[1] While elastic-net regression model combines the penalties of ridge and lasso and is trained with L1 and L2 priors as a regularizer.

Non-Linear Models

Several non-linear regression models were attempted, to include: Decision trees, Neural networks (NN), Support Vector Regressors (SVR), and sigmoid regression. Decision Trees are flowchart-like structures in which each internal node represents a test on an attribute, each branch represents the outcome of the test. Decision trees are greedy algorithm and heavily influenced by the data selected to train the model. NN's attempt to simulate the network of neurons in the human brain so that it can learn and build on the initial model provided to make future decisions. Artificial Neural Networks (ANN), a kind of NN used within this project, uses different layers of mathematical processing to determine how to react to information received by the network. Although NN's have a reputation to outperform approaches that have stronger statistical foundations, they tend to be more difficult to work with and harder to explain. An SVR model is a representation of the examples as points in space, mapped so that the examples are divided by a hyperplane. New examples are then mapped into that same space and predicted to belong to a value range based on which side of the gap they fall. The goal with SVR is to minimize error, individualizing the hyperplane which maximizes the margin, keeping in mind that part of the error is tolerated.

Growth patterns resembling sigmoidal curves are common in biological systems, to include wind turbine systems. There are numerous software packages that provide curve fitting, but the R package sicegar enables automated fitting of thousands of sigmoidal curves with minimal human supervision [2]. The sigmoid curve is represented using the logistic function (Verhulst, 1845), the formula and visualization are shown in Figure 22.

Time-Series Models

AutoRegressive Integrated Moving Average (ARIMA) models combine the features from an autoregressive model and a moving average model and is defined for a non-stationary process where the mean and variance are not constant over time and can be used with seasonal data with additional parameters that identify the number of periods in the season. ARIMA was considered as a possible method to predict energy consumption because of the datasets in this project contained time features and a seasonality that would impact the prediction. This model was not pursued after initial testing because an ARIMA model is only able forecast on the historical data upon which it was built (energy consumption) and could not be used on a different dataset (scenario) even though the datasets shared time and weather features. A Backtest function utilizing 85% of the data to build the ARIMA model on the Consumption data produced an RMSE of 10.5%.

Ensemble Techniques

Ensemble methods use multiple learning algorithms to obtain better predictive performance than could be obtained from any of the constituent learning algorithms alone. Our team applied four different ensemble techniques: Random Forest (RF), gradient boosting, ada boosting, and XGBoost. Random Forest is an ensemble prediction model consisting of a collection of different regression trees (CART) which are trained through bagging and random variable selection. The tree development rationale of trees in RF is the same as that of CART, which is through recursive partitioning. In recursive partitioning, the exact position of the cut-point and the selection of the splitting variable strongly depend on the distribution of observations in the learning sample.[3] Gradient boosting is a type of indirectly generated tree ensemble model. At each step of the model, a new tree is trained against the negative gradient of the loss function. The loss function value is oftentimes analogous to the residual error. Ada boosting combines weak classifiers to form a strong regression model. The model retrains the algorithm iteratively by choosing the training set based on the accuracy of the previous training set. While XGBoost in an implementation of the gradient boosted decision tree algorithm designed for speed, as well as performance.

Metrics

In this paper, the prediction performance against the three datasets is measured by considering four frequently used prediction accuracy evaluation indices: the coefficient of determination (R^2), the Root Mean Square Error (RMSE), mean squared error (MSE), and the Mean Absolute Error (MAE). R^2 measures the goodness of fit of a model. A high R^2 value indicates the predicted values perfectly fit the observed values, the formula and explanation of variables are provided in Figure 23. RMSE stands for the sample standard deviation of the residuals between predicted and observed values. This measure is used to identify large errors and evaluate the fluctuation of model response regarding variance. RMSE punishes large errors severely because it geometrically amplifies the error, the formula is provided in Figure 24. MAE is a statistical indicator that describes the accuracy of the prediction by comparing the absolute difference of the residuals from the observed values, the formula is provided in Figure 25.

To jointly evaluate the prediction performance across numerous models, a composite evaluation index called the Performance Index (PI), which combines R^2, RMSE, and MAE into one single measure to more easily compare the prediction performance. The Performance Index (PI) was presented and implemented by Wang, Z. et al [] using three evaluation indices, R2 RMSE, and Mean Average Percentage Error (MAPE) however, because two of our datasets contain zero kw/hr in the observed values we could not use the MAPE index and replaced it with MAE, the formula and explanation of variables are provided in Figure 26.

Solar Methods

After exploratory data analysis, it was evident the target variable (Electricity KW/hr) was not normally distributed even after applying various transformations. This was in large part due to half of the target values being zero values recorded for night time solar energy production.

Thus, it was determined to train the multivariate linear model, Lasso Regression, Ridge Regression, ElasticNet Model, Decision Tree, Random Forest, Neural Network, SVR, Xgboost, and Autoregressive Integrated Moving Average (ARIMA) time series analysis models on min max scaler, Yeo-Johnson, and Quantile Transformations since they appeared to be somewhat normally distributed. Validation dataset was employed to test each model and record Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE) as noted in the paper review. From the validation result, the lowest RMSE model with the specific transformation was then utilized to test those specific scenario dataset's 6 dates.

The dataset was first split into 80% training and 20% validation. Each model was trained on the training dataset and tested on the validation dataset with different transformations noted earlier. Furthermore, cross validation (CV) with k-fold of 10 was also conducted on each model. The ARIMA model did not work after training. Therefore, the ARIMA model was dropped from the investigation. After training each model, the CV metric noted earlier was recorded in Table 5. Furthermore, once it was determined from CV that Random Forest and XGBoost had the lowest RMSE, the feature selection was used to find the best predictors. For both models Solar Elevation was noted as 94 percent followed by Hour, month, and day. Therefore, during the scenario testing Random Forest and XGboost only had Solar Elevation, Hour, and Month as the predictors as shown in Table 5.

Since the lowest CV RMSE values were for the Random Forest and XGBoost models with Min Max Scaler, Yeo-Johnson, and Quantile Transformations, six days scenario tests were conducted on these models. The metric of these test was recorded in Table 5. Based on this metric, Random Forest and XGboost with min max scaler had the lowest MSE, RMSE, and MAE. Out of these two lowest MSE, RMSE, and MAE was XGboost. Additionally, the plot of March 15th showing predicted vs actual Electricity (KW/hr) for RF and Xgboost models with min max transformation was shown in Figure 7. Therefore, the predictions were made based on these models for those 6 days of the scenario file and we recorded the metric noted earlier.

Wind Methods

Exploratory data analysis of the wind dataset revealed the underlying data is not linear. In order to provide a robust analysis of numerous model's transformations were applied to normalize the dataset as much as possible. Cube root on the target variable (Figure 15) and square root on wind speed (Figure 16) reduced skewness and kurtosis as much as possible. However, even with those transformations the underlying sigmoid shape was still evident (Figure 17) and the linear regression performed poorly. In preparation for the non-linear and ensemble modeling techniques, min/max scaling was applied to the electricity and wind speed variables. Data was split into train/test with an 80/20 split and the entire dataset was used when using cross-validation.

Eight models were trained and tested on the scaled wind dataset: decision tree, decision tree with bagging, random forest, gradient boosting, ada boosting, neural networks, SVR, ARIMA, and XGBoost. Comparison of train/test split and cross-validation showed that models trained with cross-validation increased in performance by approximately 3-5%. Feature importance

found that wind speed dominated importance with 94.5%, month was 4.2%, day was 0.78%, while hour was 0.51%. ARIMA was applied but could not resolve the seasonality and was abandoned. After initial comparison of the eight models the random forest and XGBoost performed the best and hyper parameter tuning was executed using CV grid search, to include number of features used: one variable, windspeed; two variables, wind speed and month; three variables, wind speed, month, and hour; four variables, wind speed, month, day, and hour. However, it was found the most successful model used wind speed, month, and hour to predict electricity production. Both Random forest and XGBoost seem to account for the monthly and hourly seasonality within the wind data. Table 6 and Table 7.

Consumption Methods

To predict energy consumption, linear regression, lasso regression, ridge regression, elastic net regression, decision tree regressor, random forest regressor, neural networks, ARIMA, adaptive lasso, gradient boosting regressor, ada boost regressor, and XG boost regressor techniques were used. Each model was implemented using the Sci-Kit Learn library within Python. ARIMA and adaptive Lasso were implemented in R Studio. The ARIMA model was not used because it could not predict on a data set that was not used to create the model. The adaptive Lasso model did not return predictions that were sufficiently accurate as the other models and was also discarded. An iterative approach was taken to produce the energy consumption models. With each technique, different hyper parameters were changed and the model metrics with each hyper parameter change were recorded. The most performant model for a given technique was determined by reviewing the R^2, mean squared error (MSE), mean absolute error (MAE), root mean squared error (RMSE), and explained variance values for the test and training datasets. The ideal model for a given technique would have a high R^2, an explained variance very close to the R^2, and an RMSE close to zero. Comparing the metrics for both test and train allowed the group to determine if a given model was overfitting if the train metrics exhibited far greater performance when compared to the test metrics.

Using the previously defined criteria for optimal performance, the model with the highest (or lowest, in the case of RMSE) metrics from each technique was added to a summary table. Said summary table can be found in Table 9. Each model predicted both the KW usage for car charging, as well as the overall energy consumption for each sector at the hourly level. Ensemble and neural network methods are able to predict multiple targets (multi-output) intuitively within Sci-Kit Learn, while linear regression models must be wrapped in the MultiOutputRegressor function within Sci-Kit Learn to predict more than one target.

Results

Performance metrics for each dataset were compared and the best models were hyper-parameter tune to find the best performing models. For both the solar and wind datasets, XGBoost performed the best. While random forest performed the best on the consumption datasets.

The most performant model for the solar data was modeled using was XGBoost with min/max scaling on the Solar Elevation, Hour, and Month variables to predict solar electricity production (KW/hr) with an R^2 of 0.0.699, MAE of 7048, and RMSE of 12717. (Figure 7) Even after finding the best model to be XGBoost, it was noted that the model still has high error in predicting the exact Energy Production. Furthermore, XGBoost model also generated negative electricity values, but during post-processing these values were changed to zero.

The most performant model for the wind data was modeled using XGBoost with min/max scaling on wind speed, and sin transformation on month, and hour variables to predict wind electricity production (kw/hr); with an R^2 of 0.981, MAE of 0.015, and RMSE of 0.024 (Figure 18). Although the performance metrics were very similar between the XGBoost and Random Forest models, XGBoost was selected because it accounted for the monthly seasonality better than Random Forest. There is a strong monthly seasonal trend in the wind dataset, wind speeds are higher in the fall/winter months and lower in the spring/summer months. Figure 19 contains two scatterplots, the plot on the left is the Actual vs. Predicted values for each month for the XGBoost model, and the scatterplots on the right are for the Random Forest model. Random forest was not capturing this trend as well and therefore underperforming during the fall/winter months and over performing during the spring/summer months. XGBoost does generate negative electricity values, but post-processing changes those to zero.

The most performant model for the consumption data was modeled using random forest. The previously mentioned metric values for random forest exhibited a model that explained much of the dataset's variance, evidenced by the model's R^2 value approximately 100% for the training dataset and 99.9% for the testing dataset Table 9. The model also exhibited an extremely low RMSE at 0.0010. The actual versus predicted values were plotted within a scatter plot to examine visually examine the relationship. Said graphs can be viewed in Figure 32 and Figure 33. Ensemble and neural network techniques consistently exhibited very strong performance in predicting both the car charging electricity consumption, as well as a sector's energy consumption. Initial techniques that predicted a single target, electricity consumption, exhibited strong performance among all of the selected models. It was when the second target was introduced that a clear delineation presented itself in terms of performant and non-performant models. Linear regression models exhibited an average R^2 value of 58.39%, while non-linear regression techniques exhibited an average R^2 value of 97.48%.

Validation and Prediction

Scenario dataset

The scenario dataset did not contain the wind, solar, or consumption target variables (wind electricity production (kw/hr), solar electricity production (kw/hr), or electricity consumption (kw/sqft/hr)) and therefore feature engineering needed to be executed to allow for assessment

of model performance against the scenario. This was accomplished by querying solarray_weather.csv, solararray_solarangle.csv, and solararray_production.csv merged files.

Exploratory data analysis and feature importance during modeling indicated that wind speed, month, and hour were the most influential factors for determining the amount of electricity that can be produced at a wind turbine. However, because there is less than two years of wind training data it was not feasible to include hour within the estimate. Therefore, in order to provide a best-guess estimate for wind electricity production for the scenario dataset, the training dataset was queried for the same month and wind speed and then averaged to generate wind electricity values. For example, Figure 20 shows that to estimate electricity for March 15th at hour zero, the wind training dataset was queried, and the mean was calculated for all electricity values where month was equal to March and wind speed was equal to 3.1, returning an electricity value of 1,082.3 kw/hr. Predictions were then run against the scenario dataset and compared to the 'actual' electricity kw/hr (Figure 21).

After modeling various models and find the best to be XGboost with min max transformation, the six specific dates from scenario file were tested for accuracy. However, as it was mentioned earlier the scenario file did have the actual Solar Electricity (KW/h) production values. In order to impute these values original merged training and testing dataset was queued on Month, Day, and Hour. However, since there were four years (2010 to 2012) data, the actual production value was calculated by taking the mean of those four query for the same Hour, Day, and Month. This was repeated for each hour and each month. Based on these actual values, the prediction values from XGBoost was compared to the actual Electricity Produce and MSE, RMSE, and MAE were calculated.

Conclusion

Power City, USA wants to implement renewable wind turbine and solar array energy but needs better insight on renewable production capacity and the City's energy consumption. We have provided a robust analysis of linear, non-linear, time series, and ensemble modeling techniques to identify the most performant models to predict hourly solar energy production, hourly wind energy production, and hourly energy consumption by the population of over 105,000 people while accounting for an increase in the number of electric vehicles. We identified strong seasonal trends within each dataset and the most performant models account for those trends. Specifically, the monthly and hourly trends visible within the solar and wind data. Feature selection was applied to reduce the large datasets to a smaller number of variables, which increased performance and simplified the models. Across all three datasets, ensemble learning techniques performed the best with Random Forest and XGBoost being the most accurate based on numerous performance metrics. The most accurate models were then executed against the hold-out scenario year to model six days: March 15th, June 26th, July 3rd, October 13th, November 19th, and December 25th.

Comparison of model predictions for those six days reveals solar production out performs wind production, but considering solar is not a viable resource for night time hours wind energy can

support nighttime energy needs (Figure 34 - Figure 39) . Renewable energy cannot accommodate all energy consumptions needs but it can cover a good portion during the middle of the day during peak solar production (Figure 40- Figure 45). The three models generated can provide invaluable tools for the city to forecast energy needs and potential energy production based on weather conditions and their current population demographics.

References

[1] Hui Zou, "The Adaptive Lasso and Its Oracle Properties", p. 1, http://users.stat.umn.edu/~zouxx019/Papers/adalasso.pdf

[2] Pina, André, Carlos Silva, and Paulo Ferrão. "Modeling hourly electricity dynamics for policy making in long-term scenarios." *Energy Policy* 39, no. 9 (2011): 4692-4702.

[3] Wang, Zeyu, Yueren Wang, Ruochen Zeng, Ravi S. Srinivasan, and Sherry Ahrentzen. "Random Forest based hourly building energy prediction." *Energy and Buildings* 171 (2018): 11-25.

[4] Caglar, M. Umut, Ashley I. Teufel, and Claus O. Wilke. "Sicegar: R package for sigmoidal and double-sigmoidal curve fitting." *PeerJ* 6 (2018): e4251.

Appendix

Preprocessing

Table 1: Summary of Original Datasets

		Summary of Original Datasets											
		Date Start	Hour Start	Date End	Hour End	Time Increment	Timeframe	Columns	Rows	Duplicates	Missing Values	Leap Day	Description
1	powercity_weather_scenario	1-Jan-XXXX	0	31-Dec-XXXX	23	0-23	12 months	13	8,784	no	no	yes	Weather feature data for Scenario year
2	calendar_days_scenario	1-Jan-XXXX	NA	31-Dec-XXXX	NA	NA	12 months	7	366	no	no	yes	Provides details on Holiday and school day for each day in the scenario file
3	powercity_population	NA	NA	NA	NA	NA	NA	8	44	no	no	NA	Population by census tract, by age group
4	sector_use_matrix	NA	NA	NA	NA	NA	NA	6	8	no	no	NA	Aligns with the powercity_consumption file on Sector.
5	calendar_days_consumption	1-Jan-XXXX	NA	31-Dec-XXXX	NA	NA	12 months	7	365	no	yes	no	Provides details on Holiday and school day for each day in the consumption file
6	car_charging	1-Jan-XXXX	1	31-Dec-XXXX	24	1-24	12 months	6	8,760	no	no	no	Electricity consumption by electric car charging (evenings)
7	powercity_consumption	1-Jan-XXXX	1	31-Dec-XXXX	24	1-24	12 months	6	70,080	yes	no	no	Will need to pivot the sector column to reduce observations / rows.
8	powercity_weather_consumption	1-Jan-XXXX	1	31-Dec-XXXX	24	1-24	12 months	12	8,760	no	no	no	Weather features data for Consumption year
9	powercity_solarangle_consumption	1-Jan-XXXX	1	31-Dec-XXXX	24	1-24	12 months	6	8,783	no	no	yes	Solar angle data for Consumption year
10	solararray_production	4-Jan-10	16	31-Aug-14	20	1-24	56 months	3	18,704	no	yes	yes	Electricity production by photovoltaic solar panels
11	solararray_solarangle	4-Jan-10	1	31-Aug-14	5	1-24	56 months	6	40,820	yes	yes	yes	Solar angle data for specified date range
12	solararray_weather	1-Jan-10	0	18-Sep-14	13	0-23	57 months	13	41,322	no	yes	yes	Weather feature data for specified date range
13	windfarm_production	25-Mar-11	1	31-Dec-12	24	1-24	22 months	3	15,384	no	yes (174)	yes	Wind electricity production by day/hour
14	windfarm_windspeed	24-Mar-11	19	31-Dec-12	23	0-23	22 months	6	15,389	no	yes (169)	yes	Wind Speed by day/hour

Table 2: Sector Use Matrix

Sector	<5 SQFT	5>18 SQFT	18>25 SQFT	25>65 SQFT	65+ SQFT	Total SQFT
Food Service	-	430.66	4,162.04	4,951.88	1,194.86	10,739.44
Health Care	41,309.40	245.66	615.97	4,647.15	970.21	47,788.39
K-12 Schools	-	1,130,335.82	-	-	-	1,130,335.82
Lodging	-	146.80	706.92	6,751.95	1,090.63	8,696.30
Office	-	-	200,249.73	937,189.62	1,373.08	1,138,812.43
Residential	343,213.65	1,318,869.20	5,871,824.24	56,083,162.32	7,891,377.96	71,508,447.38
Grocey	-	85.59	315.57	2,214.40	282.62	2,898.18
Stand Alone Retail	-	166.79	2,823.68	5,873.38	1,239.14	10,102.99
Grand Total	***384,523.05***	***2,450,280.53***	***6,080,698.14***	***57,044,790.70***	***7,897,528.49***	***73,857,820.93***

Solar:

Table 3: Solar Data Missing Values

Final Table Missing Values per Attribute					
Attributes	**Missing**	**Attributes**	**Missing**	**Attributes**	**Missing**
Location	0	Dew_Point	270	Wind_Speed	184
Year	0	Humidity_Fraction	270	Solar_Elevation	2,222
Month	0	Precipitation	12,590	Electriciy_KW_HR	22,618
Day	0	Pressure	15,342		
Hour	0	Temperature	150		
Cloud_Cover_Fraction	191	Visibility	142		

Table 4: Solar Data Summary Statistics

Solar Production Dataset Summary Statistic							
Variable	**Mean**	**Std.Dev**	**Median**	**Min**	**Max**	**Skew**	**Kurtosis**
Cloud Cover Fraction	0.46	0.47	0.25	0.0	1	0.14	-1.88
Dew Point	3.92	10.53	4.40	-27.2	26.10	-0.18	-0.87
Humidity Faction	0.70	0.16	0.72	0.14	1	-0.50	-0.39
Precipitation	0.12	0.86	0.0	0	41.1	18.55	513.13
Pressure	991.10	6.11	991.03	958.4	1011.2	-0.28	1.17
Temperature	9.54	11.42	10.0	-23.9	37.2	-0.12	-0.91
Visibility	14.57	3.71	16.09	0	16.09	-2.48	4.96
Wind Speed	4.1	2.45	3.6	3.94	18.5	0.75	0.87
Solar Elevation	1.07	34.28	0.57	-69.23	69.36	0.0	-0.9
Electricity_KW_HR	32317.89	50731.17	2207.68	0.0	205619.36	1.57	1.23

Table 5: Solar, Cross Validation Metric for Solar with Three Transformations

	DATA: SOLAR (with Quantile transformation)				
	CV				
	R^2	MSE	MAE	RMSE	Explained Variance
Linear Regression	0.823	0.025	0.128	0.159	0.824 (+/- 0.085)
Lasso Regression	-0.026	0.145	0.372	0.385	0 (+/- 0.0)
Ridge Regression	0.824	0.025	0.128	0.159	0.824 (+/- 0.085)
ElasticNet Model	0	0.149	0.372	0.385	0.0 (+/- 0.0)
Decision Tree	0.96	0.006	0.036	0.076	0.96 (+/-0.02)
Random Forest	0.979	0.003	0.028	0.055	0.979 (+/- 0.011)
Neural Network					
SVR	0.867	0.019	0.0115	0.138	0.882 (+/-0.055)
Xgboost	0.962	0.005	0.04	0.074	0.962 (+/- 0.02)

	DATA: SOLAR (with MinMax transformation)				
	CV				
	R^2	MSE	MAE	RMSE	Explained Variance
Linear Regression	0.591	0.024	0.121	0.154	0.593 (+/- 0.21)
Lasso Regression	0	0.061	0.194	0.247	0 (+/- 0.0)
Ridge Regression	0.591	0.024	0.121	0.154	0.593 (+/-0.21)
ElasticNet Model	0	0.061	0.194	0.247	0 (+/- 0.0)
Decision Tree	0.715	0.017	0.06	0.129	0.715 (+/- 0.131)
Random Forest	0.849	0.009	0.046	0.094	0.85 (+/- 0.079)
Neural Network					
SVR	0.725	0.016	0.1	0.126	0.73 (+/- 0.137)
Xgboost	0.845	0.009	0.052	0.095	0.846 (+/- 0.074)

	DATA: SOLAR (with Yeo Johnson transformation)				
	CV				
	R^2	MSE	MAE	RMSE	Explained Variance
Linear Regression	0.774	0.22	0.395	0.47	0.776 (+/- 0.062)
Lasso Regression	0.775	0.22	0.395	0.469	0.776 (+/-0.06)
Ridge Regression	0.774	0.22	0.395	0.47	0.776 (+/- 0.062)
ElasticNet Model	0.356	0.631	0.782	0.795	0.364 (+/- 0.01)
Decision Tree	0.978	0.022	0.056	0.147	0.978 (+/-0.017)
Random Forest	0.988	0.012	0.046	0.108	0.988 (+/- 0.008)
Neural Network					
SVR	0.893	0.104	0.231	0.323	0.893 (+/-0.035)
Xgboost	0.968	0.031	0.074	0.176	0.968 (+/- 0.02)

Figure 1: Solar, Histograms of raw Electricity (KW/Hr) and Solar Elevation with different transformations

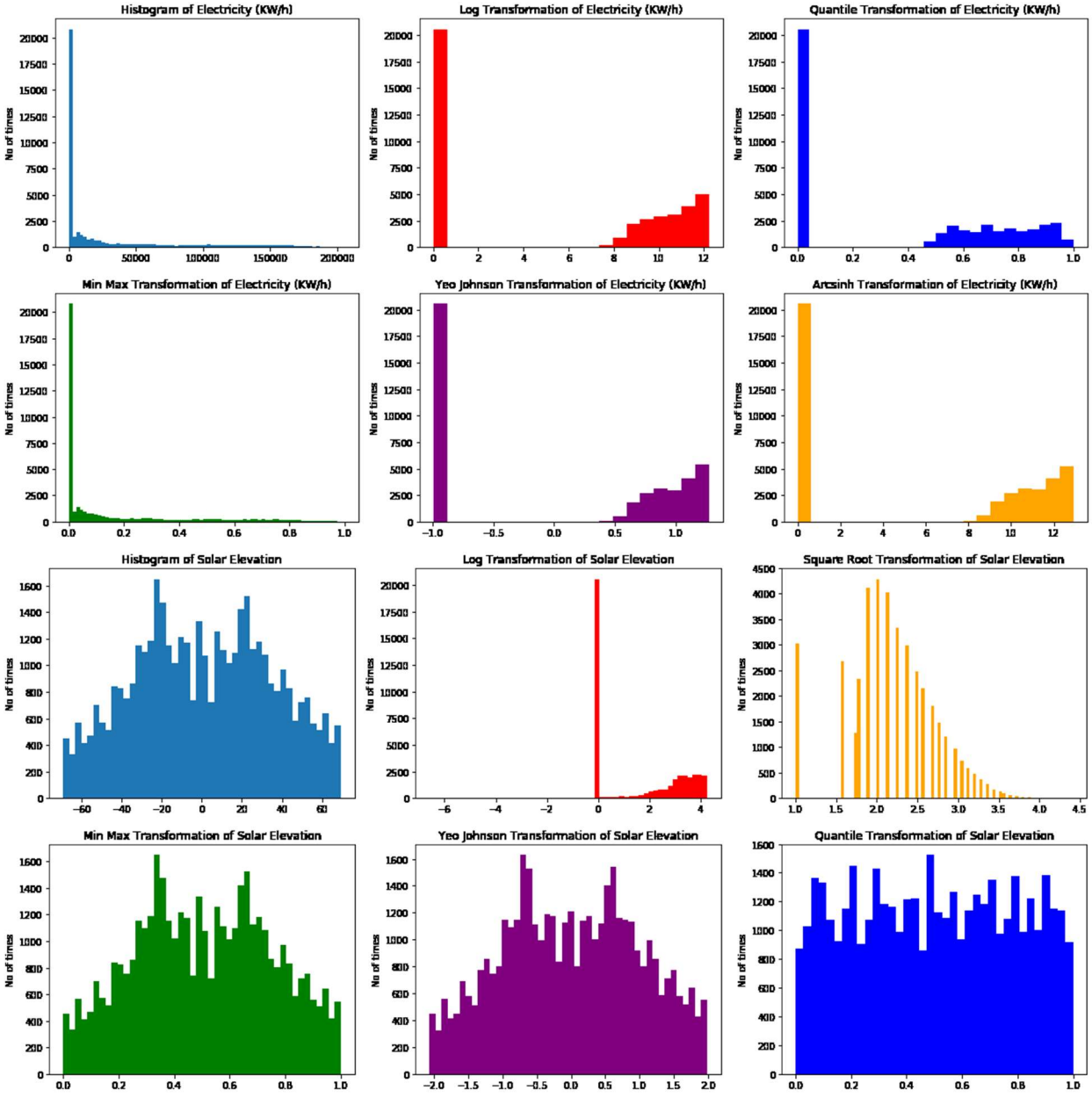


Figure 2: Solar Dataset Correlation Plot

Figure 3: Solar, Heatmap of Solar Dataset target value Electricity (KW/hr) over year and month

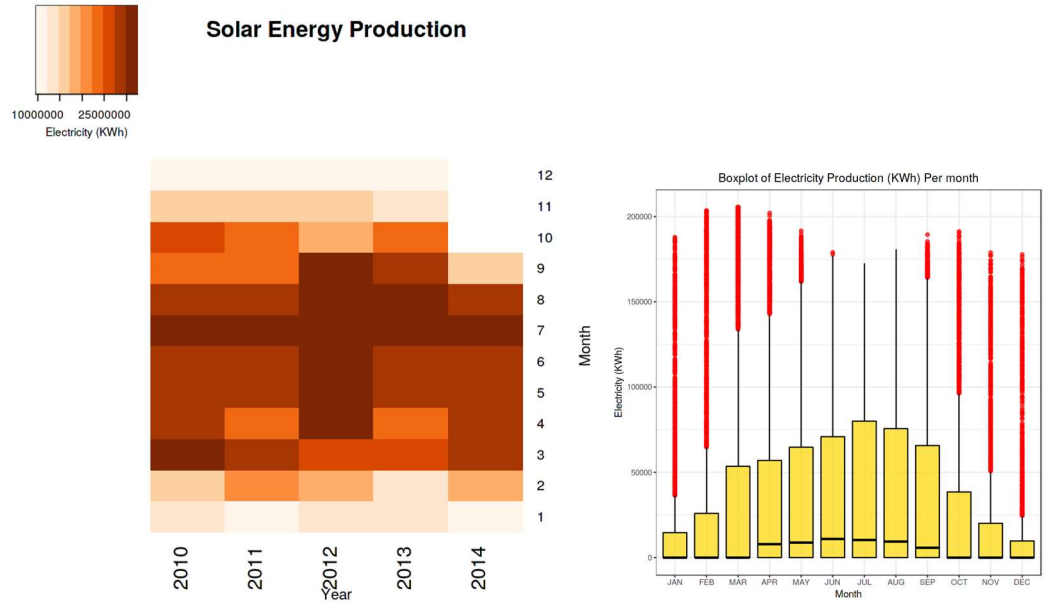


Figure 4: Solar, Graph of few Solar features with smoothing line to see the change over time

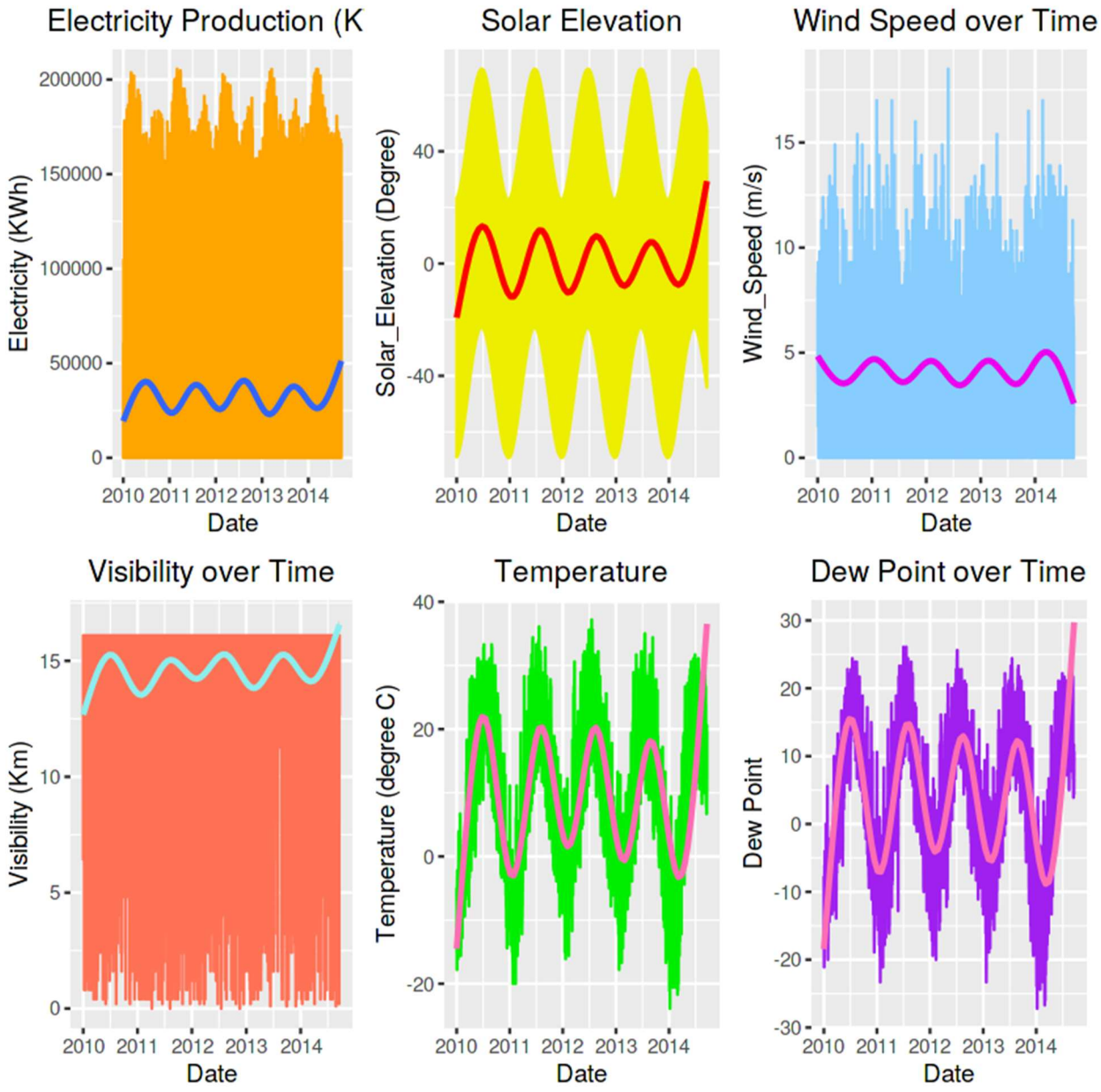


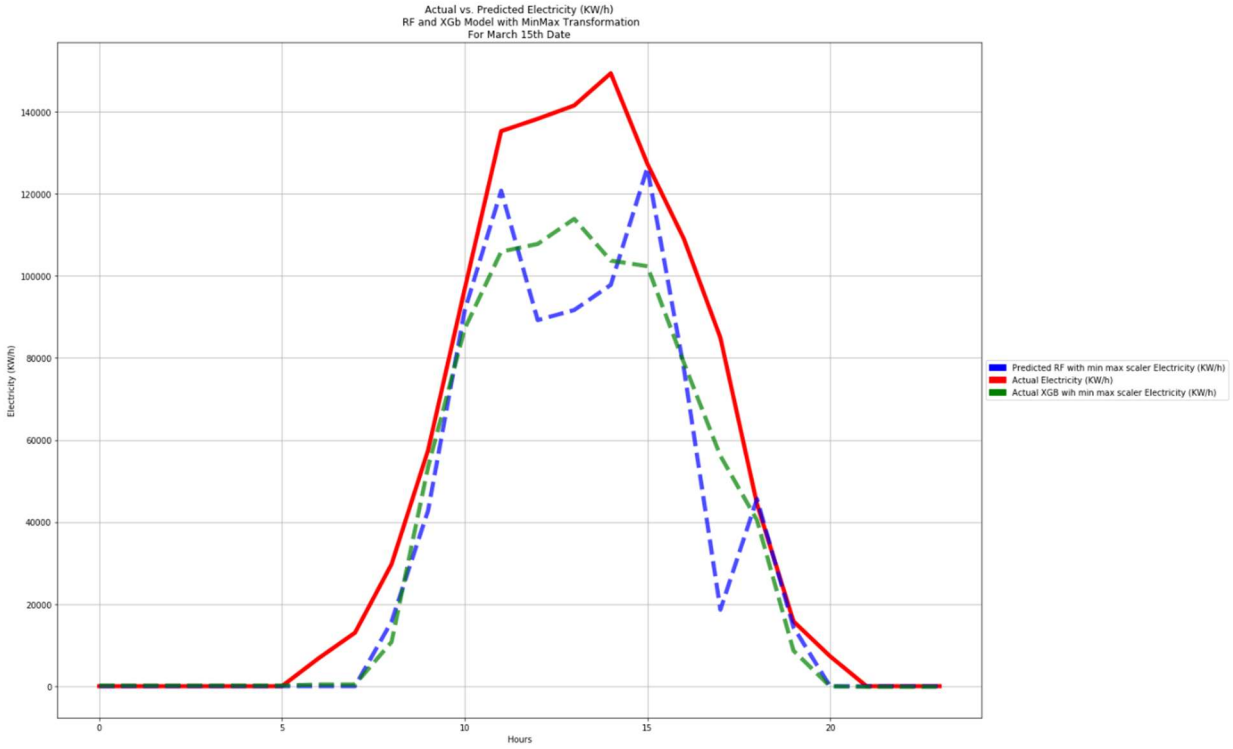
Figure 5: Solar, Feature Importance

```
Features sorted by their score:
[(0.9405, 'Solar_Elevation'), (0.0255, 'Hour'), (0.0124, 'Month'), (0.0063, 'Day'), (0.0035, 'Cloud_Cover_Fraction'), (0.003, 'Temperature'), (0.0026, 'Humidity_Fraction'), (0.
0018, 'Pressure'), (0.0015, 'Dew_Point'), (0.0014, 'Wind_Speed'), (0.0007, 'Year'), (0.0005, 'Precipitation'), (0.0004, 'Visibility')]
```

Figure 6: Solar, Six-days Scenario File Test Data:

	6 days Scenario File Test								
	MinMax Transformation			Yeo-Johnson Transformation			Quantile Transformation		
	MSE	RMSE	MAE	MSE	RMSE	MAE	MSE	RMSE	MAE
Random Forest	312284184	17671	9140	582821451	24142	13632	410461463	20260	11300
Xgboost	161729682	12717	7048	261230471	16163	8944	179084767	13382	7977

Figure 7: Solar, Actual Predicted with Random Forest, Predicted with XGBoost for March 15



Wind:

Figure 8: Wind speed vs Electricity Generation Sigmoid Curve

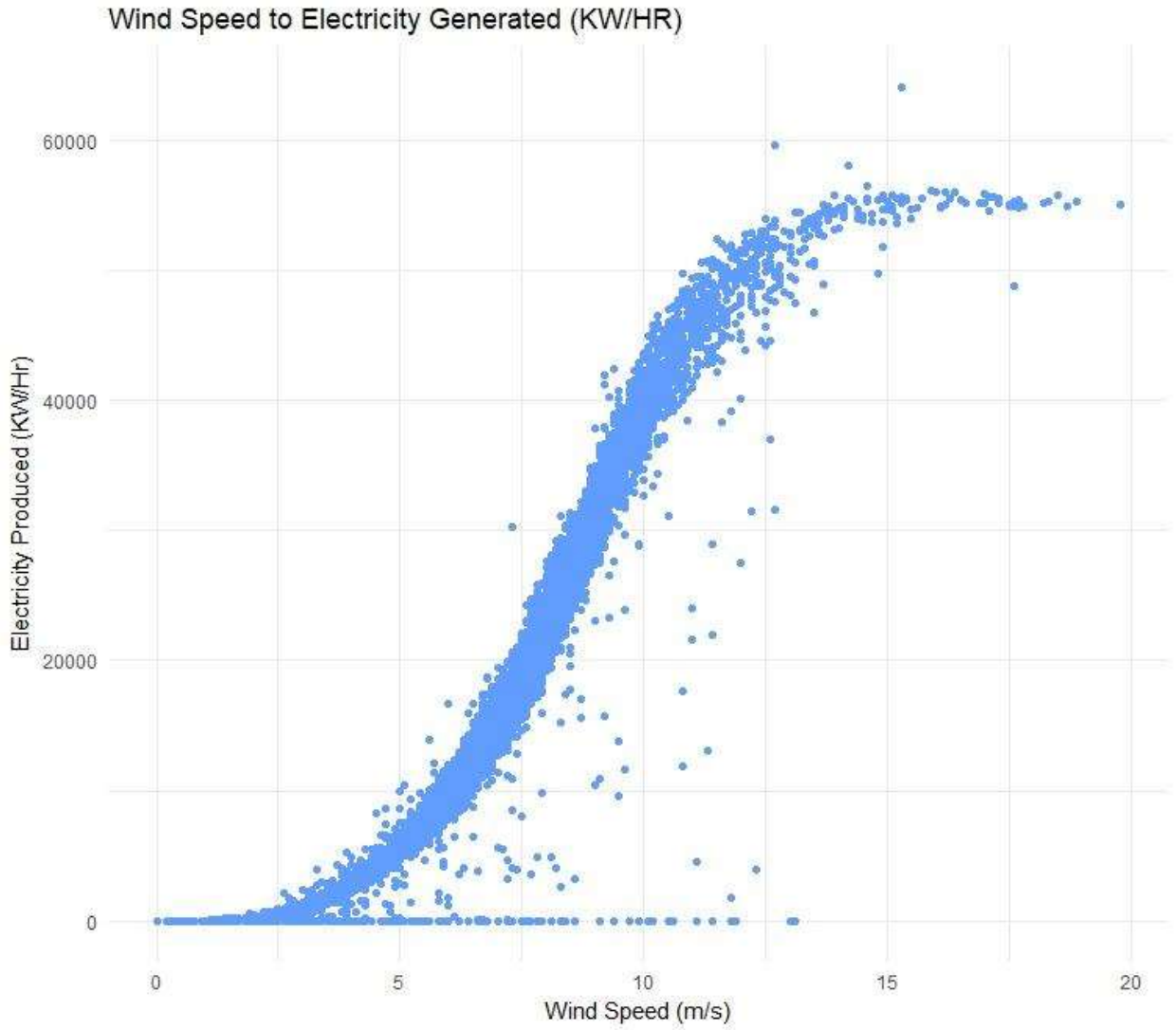
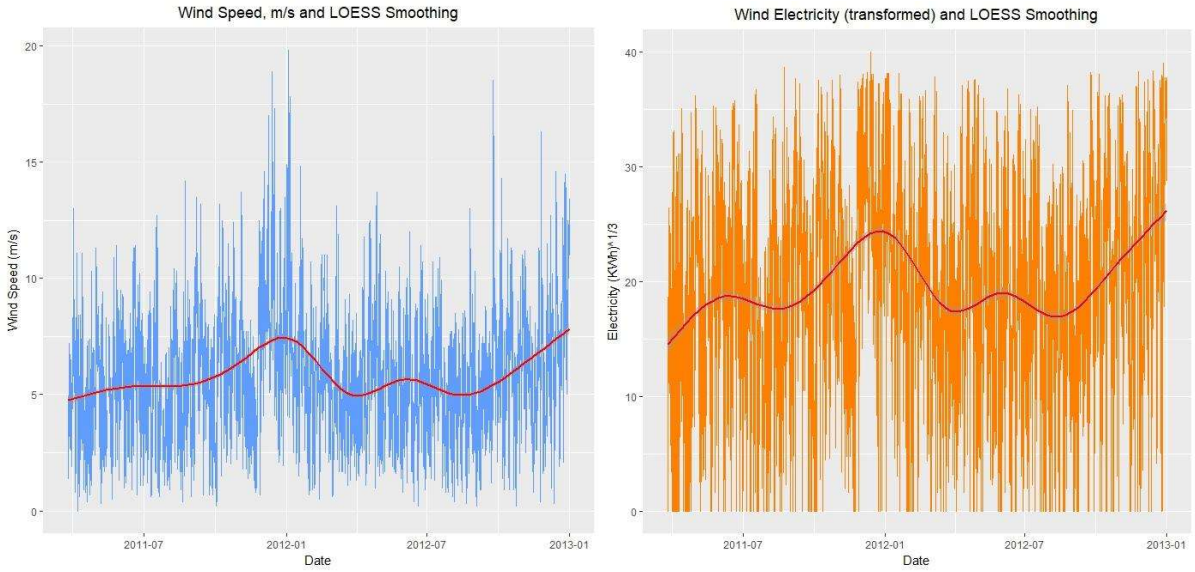


Figure 9: Time series of wind speed and electricity production to show seasonality:



Wind speed and Electricity production follow the same pattern, and electricity production is heavily influenced by wind speed.

Figure 10: Average wind electricity by month and hour showing trends

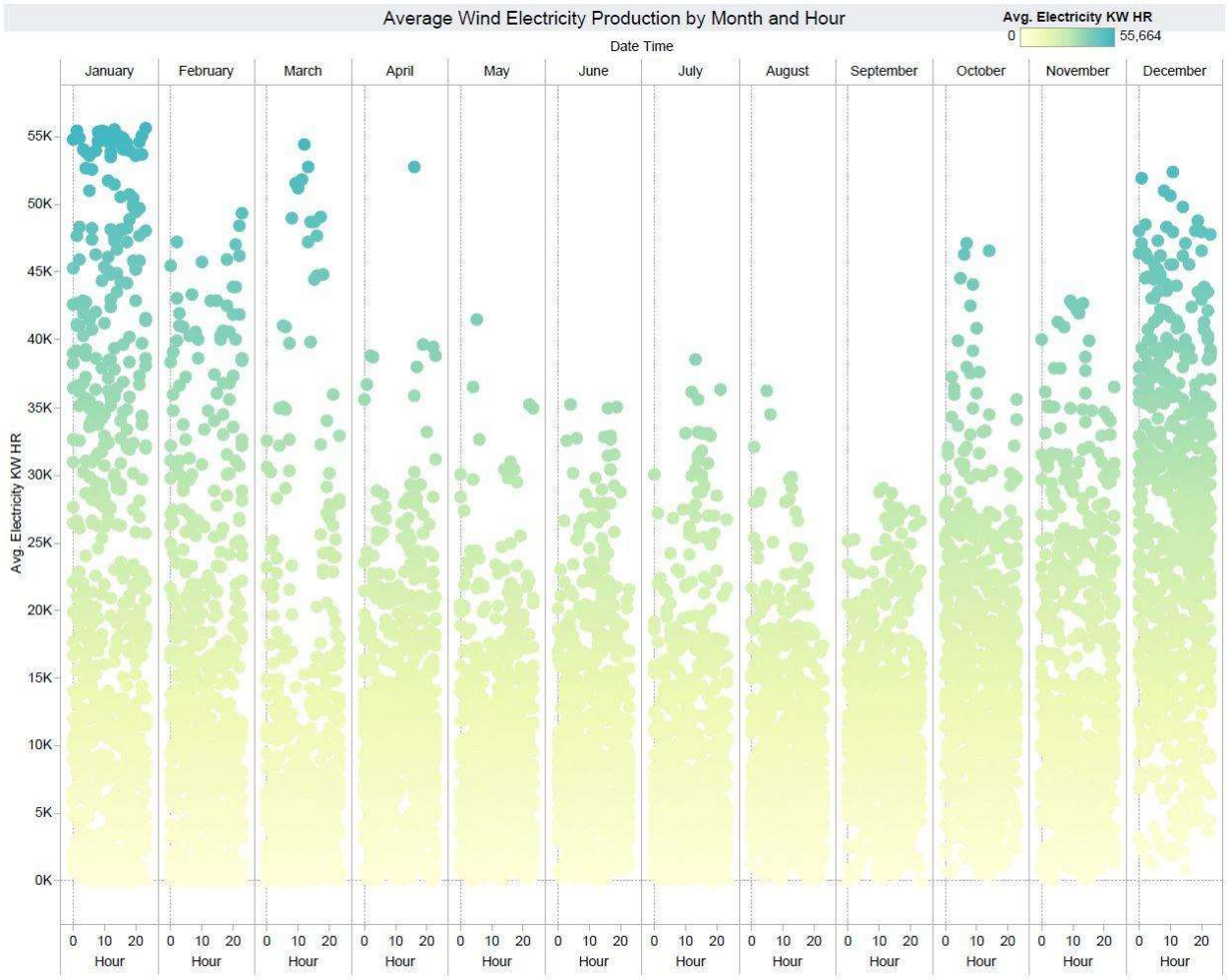


Figure 11: Sum of Wind Electricity per day to show weak daily seasonal trend

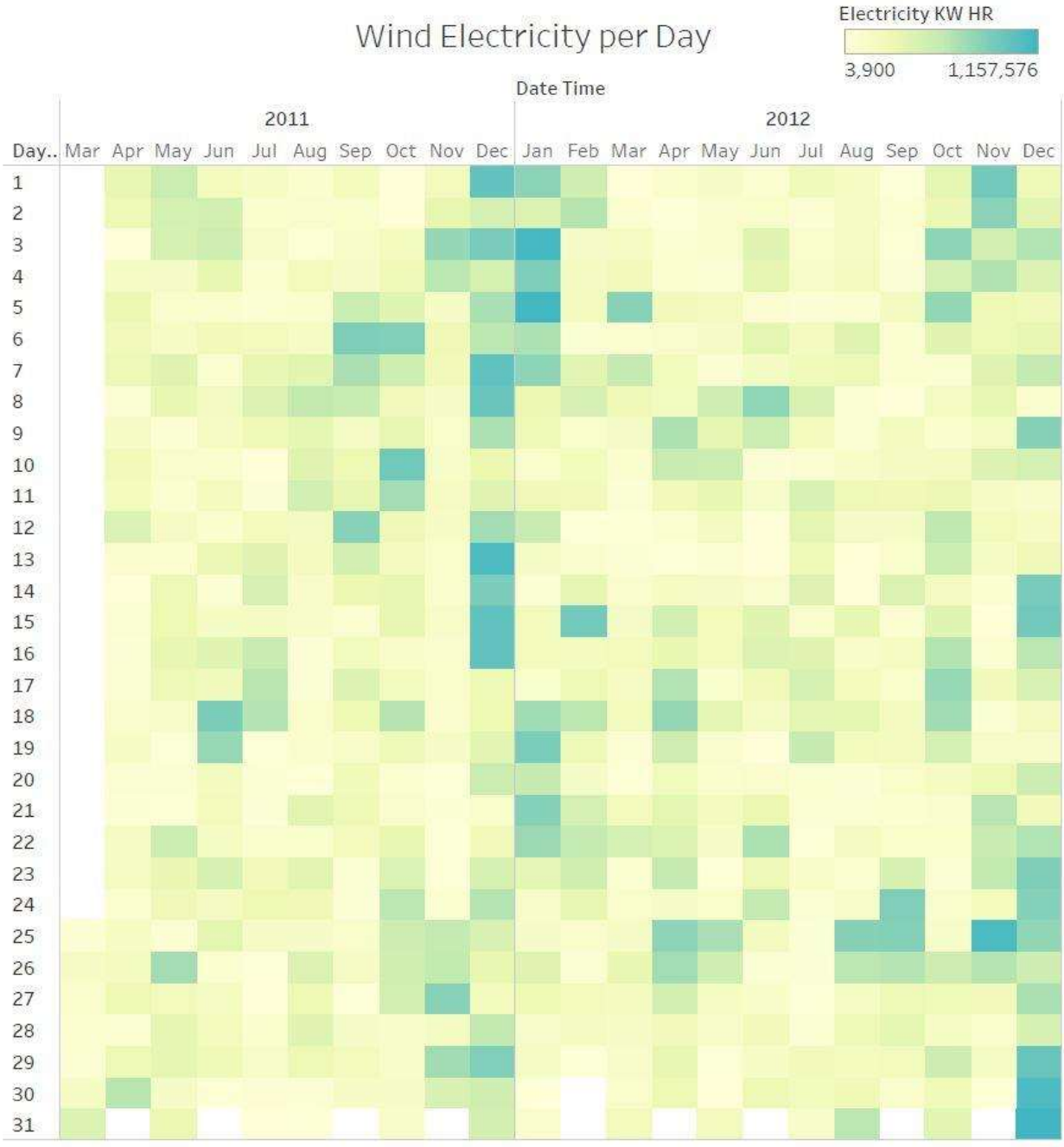


Figure 12: Wind: Violin plot of wind speed

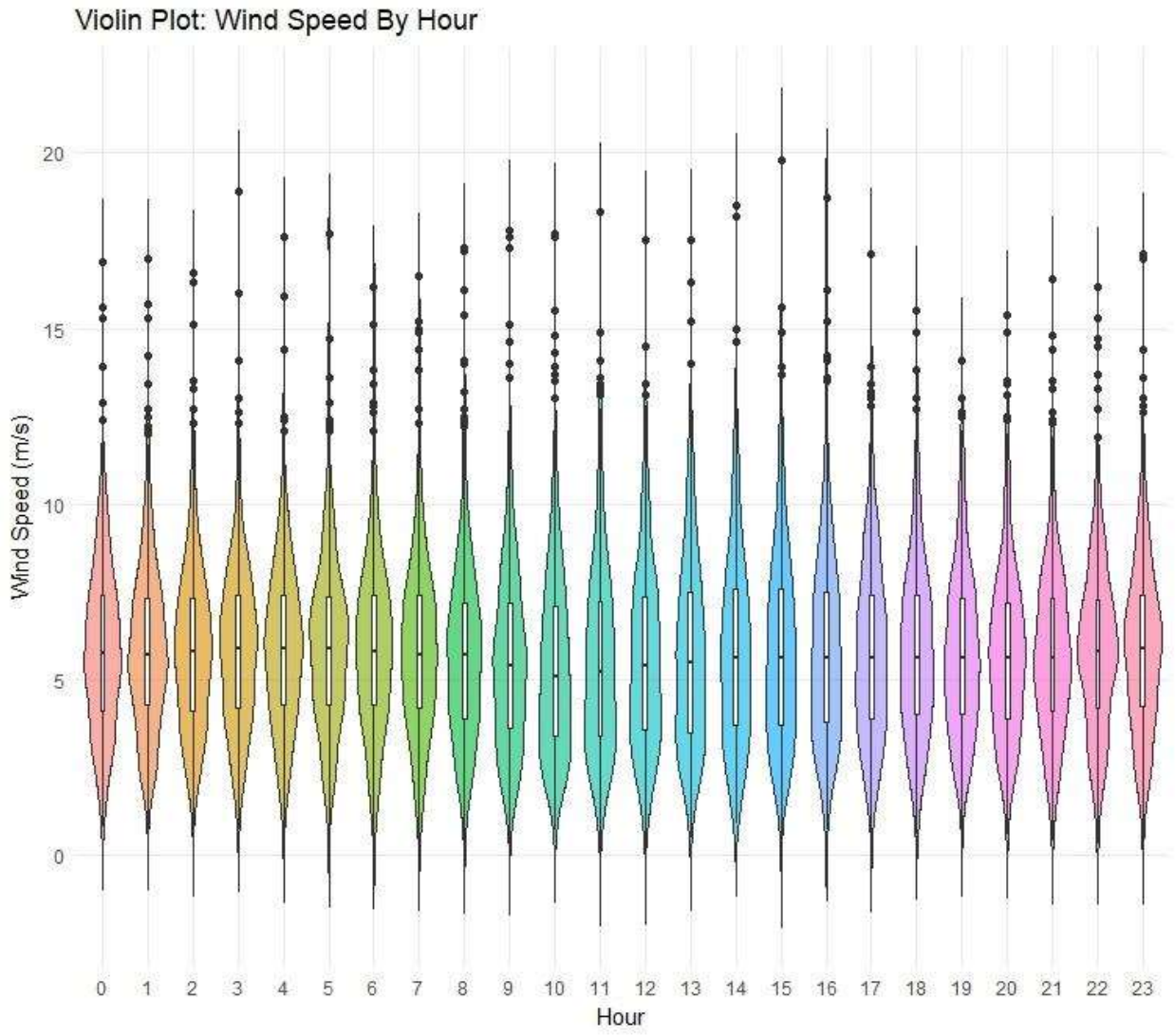


Figure 13: Wind: SIN Transformation of Hour Values

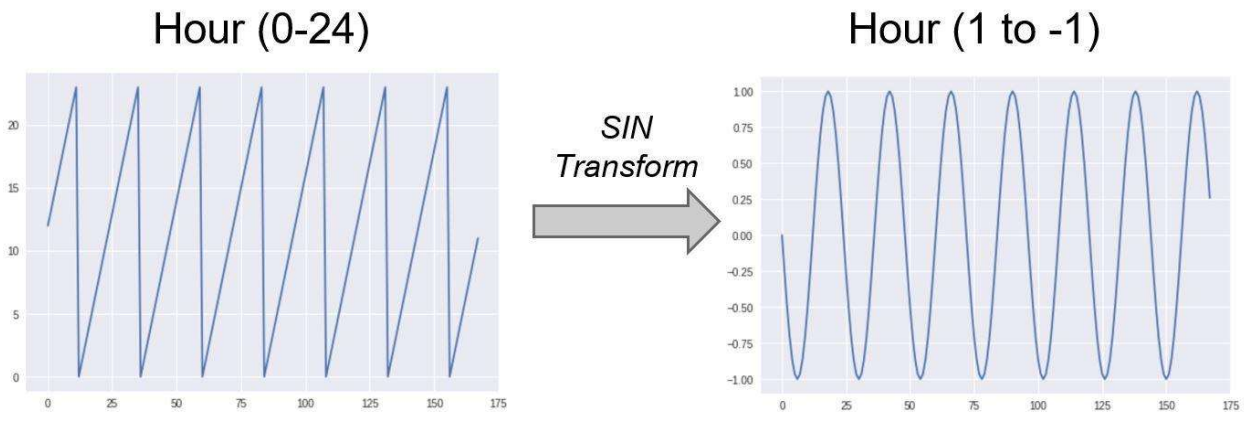


Figure 14: Wind: Wind speed with outliers removed

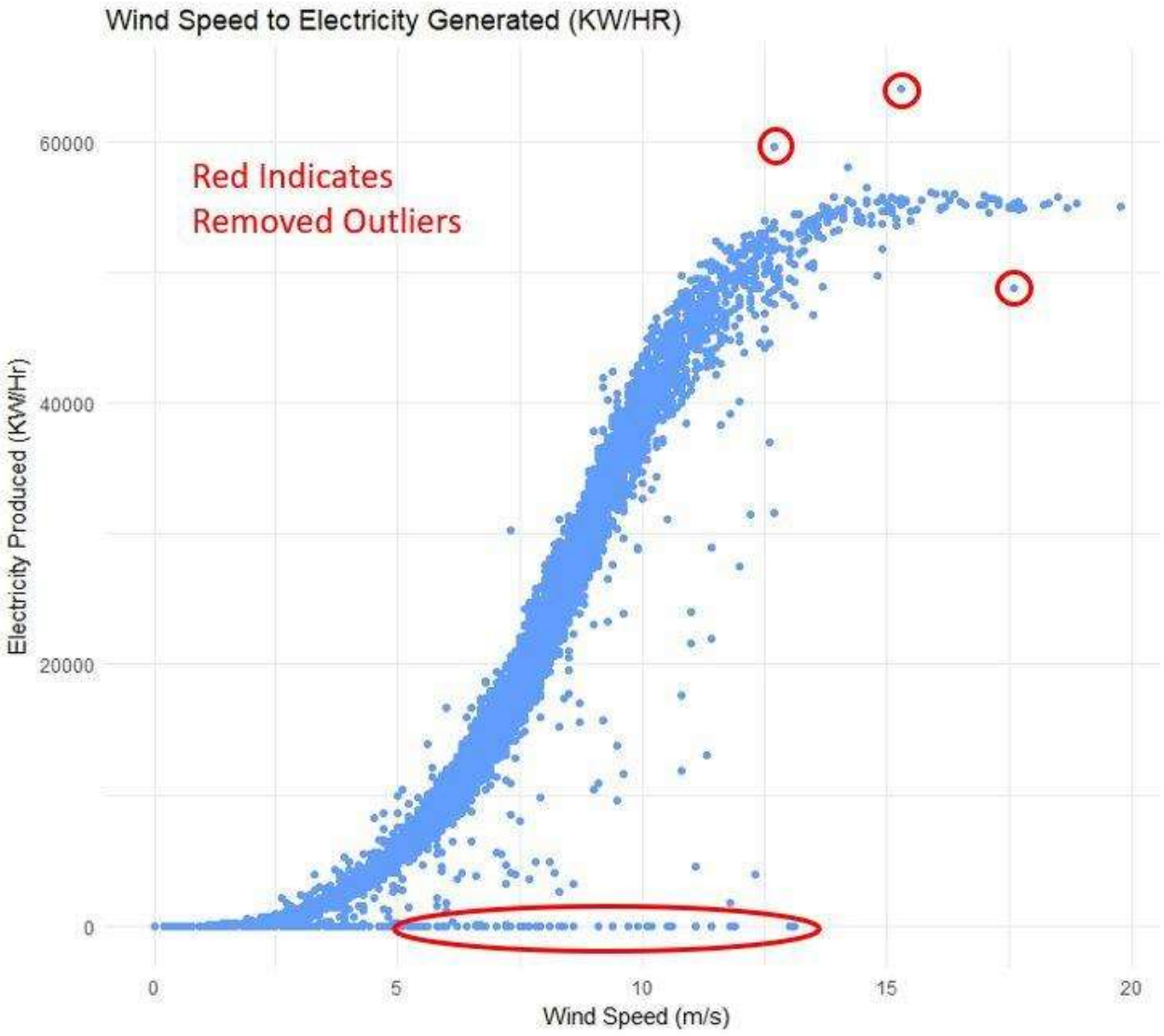


Figure 15: Transformations applied to Wind electricity per hour to make it more normally distributed

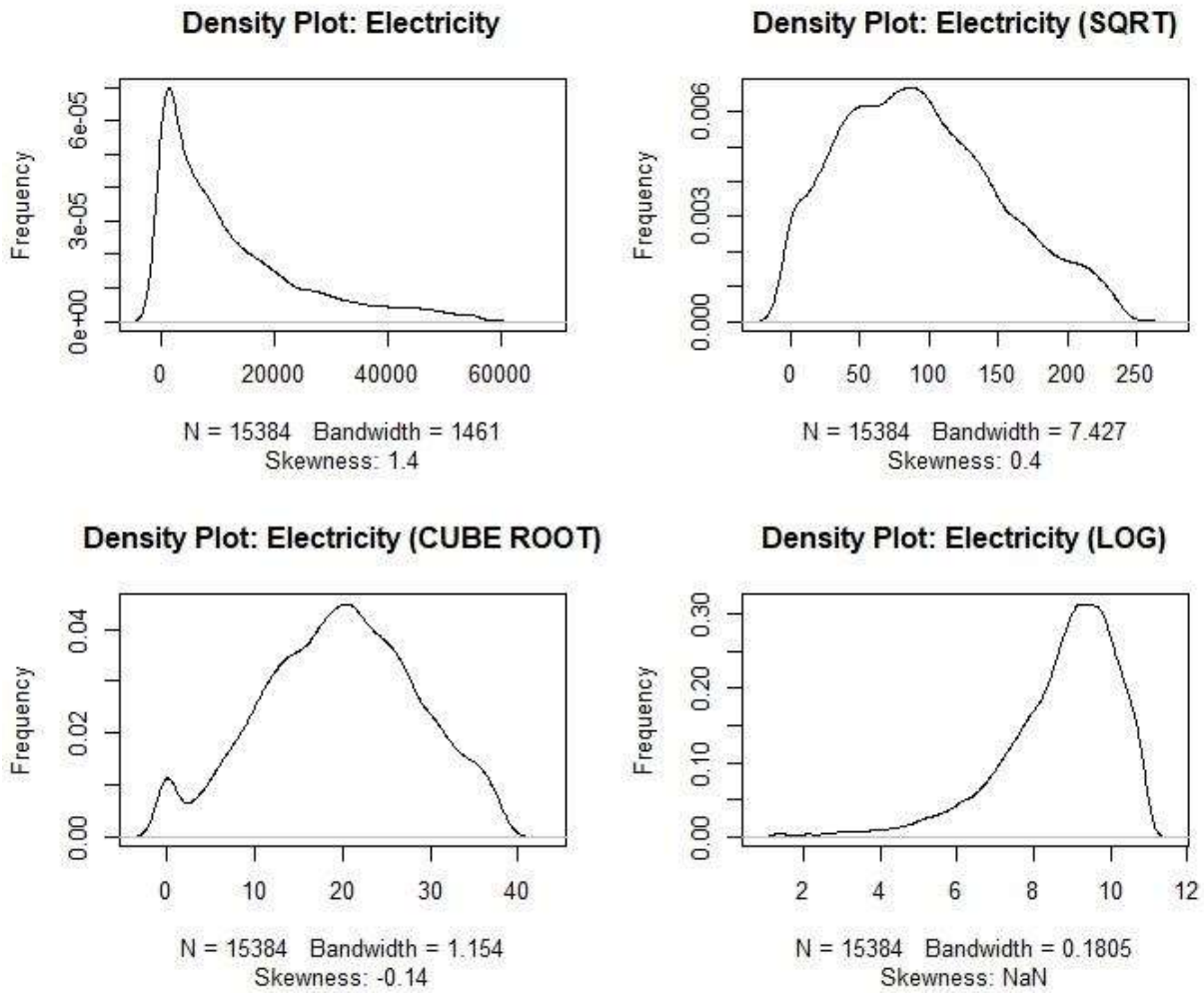


Figure 16: Transformations applied to Wind speed to make it more normally distributed

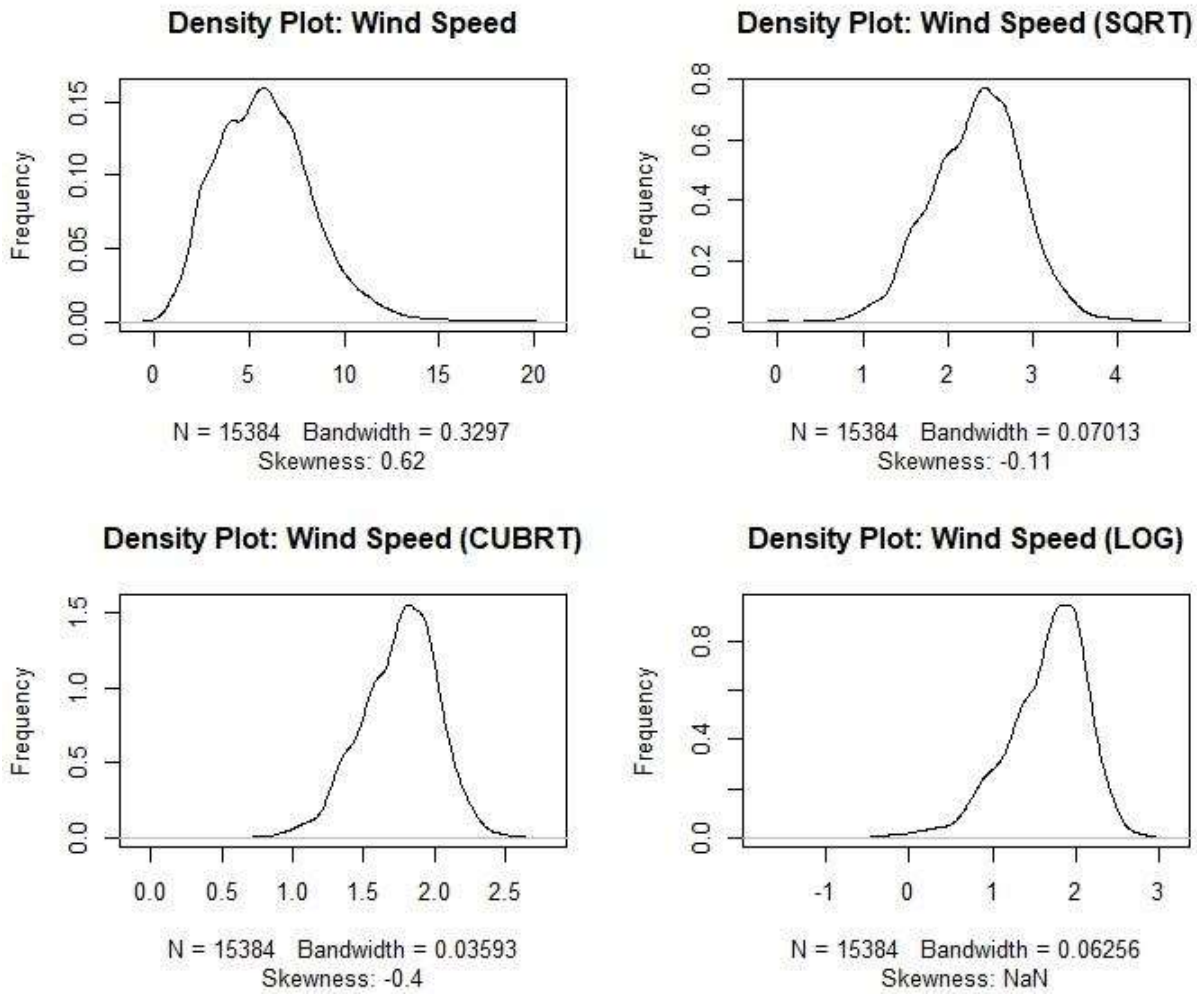


Figure 17: Transformed Wind Speed to Electricity Generated

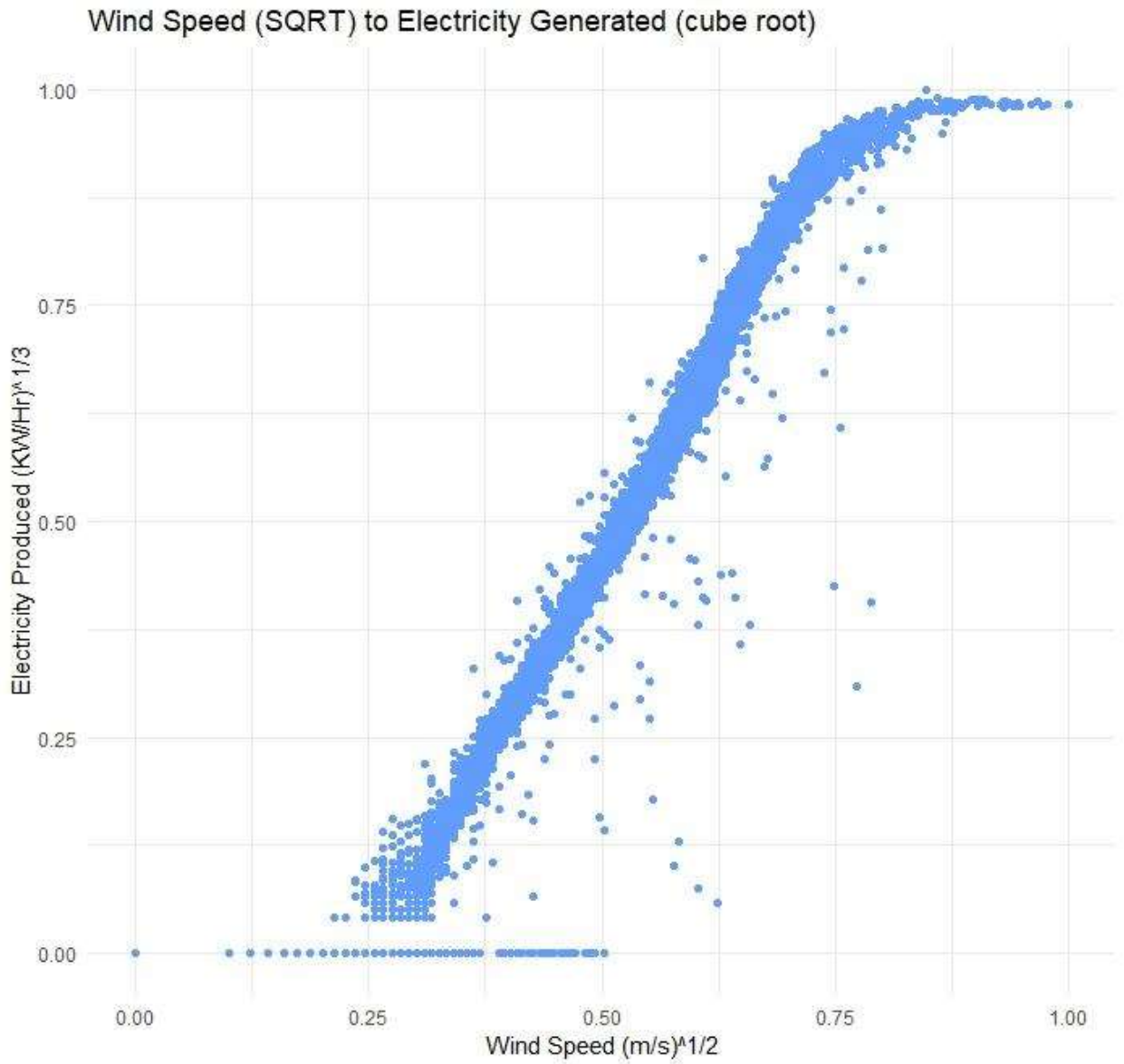


Table 6: Wind Train/Test Split for Model Analysis

	Wind Production Models					
	Training Data (80%)			Testing Data (20%)		
	RMSE	RMSE (Kw/Hr)	Explained Variance	RMSE	RMSE (Kw/Hr)	Explained Variance
Linear Regression	0.0422	2450	0.865	0.15	8707	0.721
Non-Linear Regression	0.031	1782	0.98	0.028	1637	0.984
Decision Tree	0.027	1567	0.984	0.035	2032	0.974
Decision Tree Bagging (10)	0.27	1567	0.984	0.031	1799	0.977
Decision Tree Bagging (250)	0.027	1567	0.984	0.032	1857	0.979
Decision Tree Bagging (1000)	0.027	1567	0.984	0.031	1799	0.979
Random Forest (100)	0.026	1509	0.985	0.033	1915	977
Random Forest (500)	0.027	1567	0.985	0.032	1857	0.977
Random Forest (1000)	0.03	1741	0.98	0.026	1509	0.986

Table 7: Wind, final model performance

	Wind Production Models					
	R^2	MAE	RMSE	RMSE (Kw/Hr)	Explained Variance	Performance Index
Linear Regression	0.865	0.077	0.091	5283	0.87 (+/- 0.05)	0.000
SVR (CV=10)	0.875	0.065	0.08	4683	0.87 (+/- 0.08)	0.096
Ada Boosting (CV=10)	0.872	0.03	0.049	2844	0.945 (+/- 0.031)	0.360
Neural Network (CV=10)	0.872	0.02	0.034	1974	0.966 (+/- 0.065)	0.458
Gradient Boosting (CV=10)	0.872	0.015	0.029	1683	0.979 (+/- 0.027)	0.498
Non-Linear (Sigmoid) Regression	0.975	0.017	0.029	1683	0.976 (+/- 0.05)	0.524
Decision Tree (1var)	0.979	0.014	0.031	1799	0.98 (+/- 0.02)	0.531
Decision Tree (1var, CV=10)	0.981	0.015	0.026	1509	0.98 (+/- 0.06)	0.546
Random Forest (CV=10)	0.979	0.014	0.026	1509	0.979 (+/- 0.027)	0.550
XGBoost (CV=10)	0.981	0.015	0.024	1393	0.98 (+/- 0.06)	0.553

Figure 18: Final Wind Model: XGBoost Performance

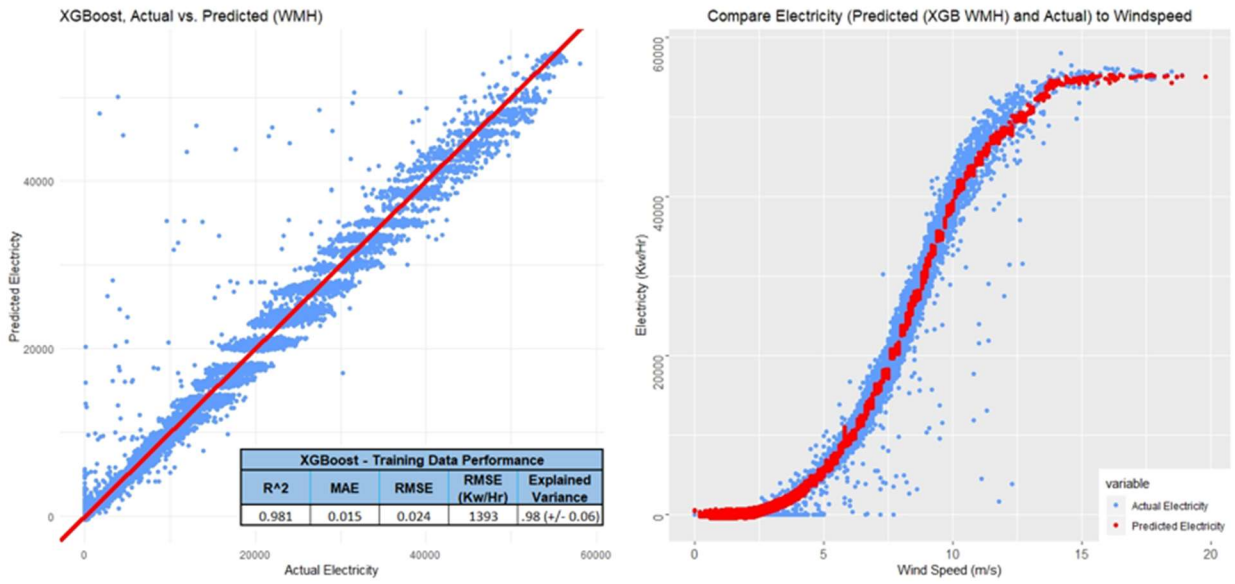


Figure 19: XGBoost vs. Random Forest for Wind Production

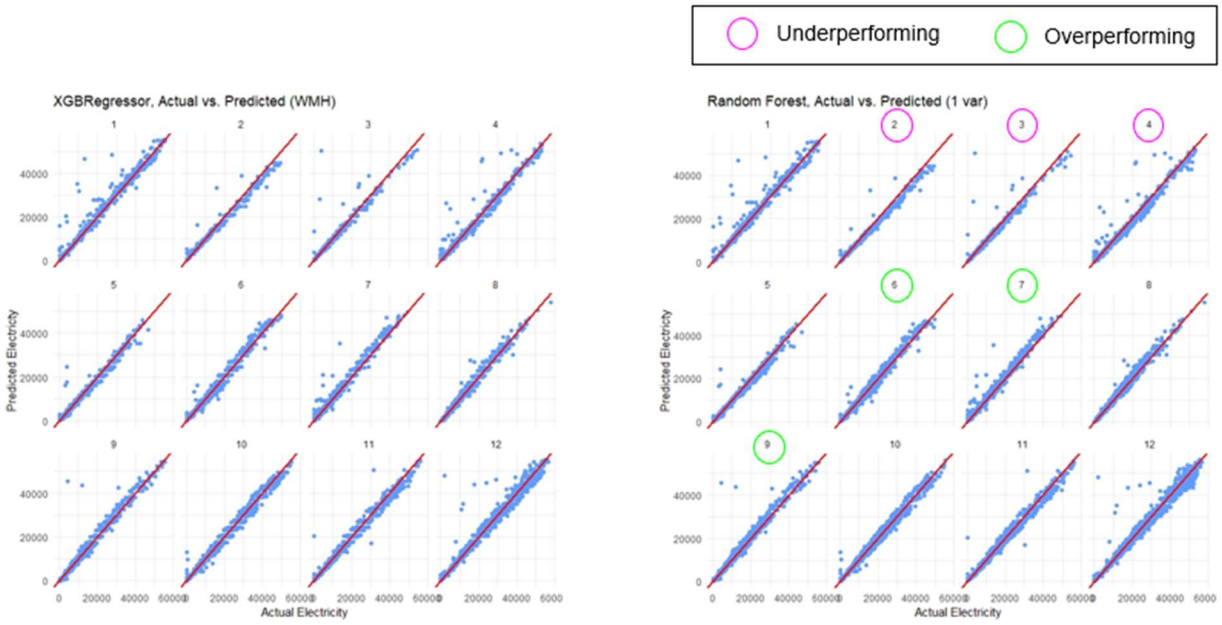


Figure 20: Visual explanation of method to impute wind target variable for the scenario dataset

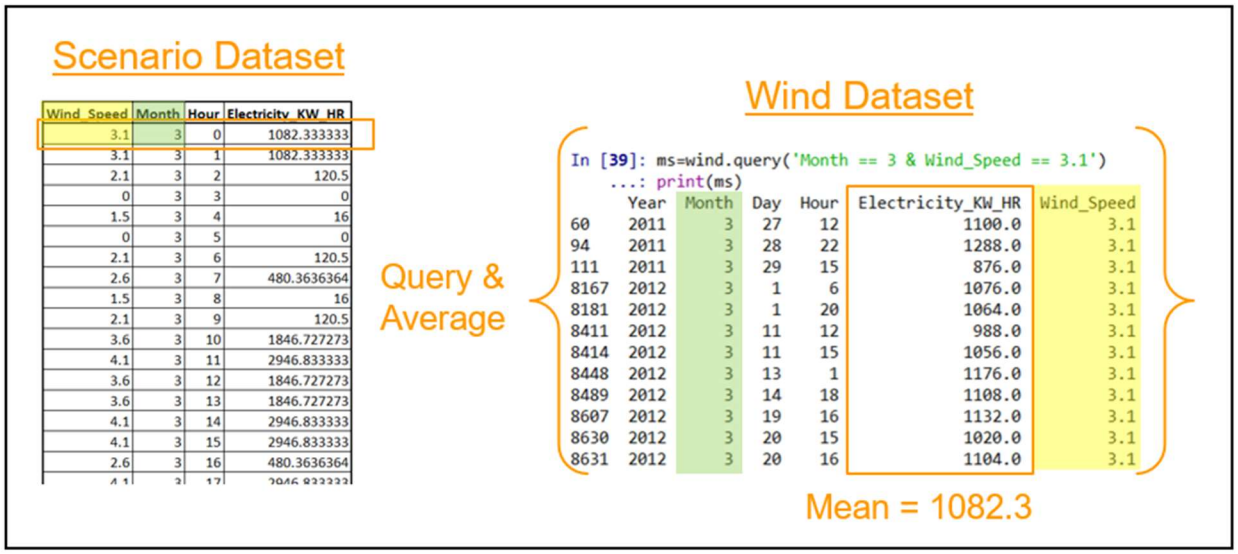
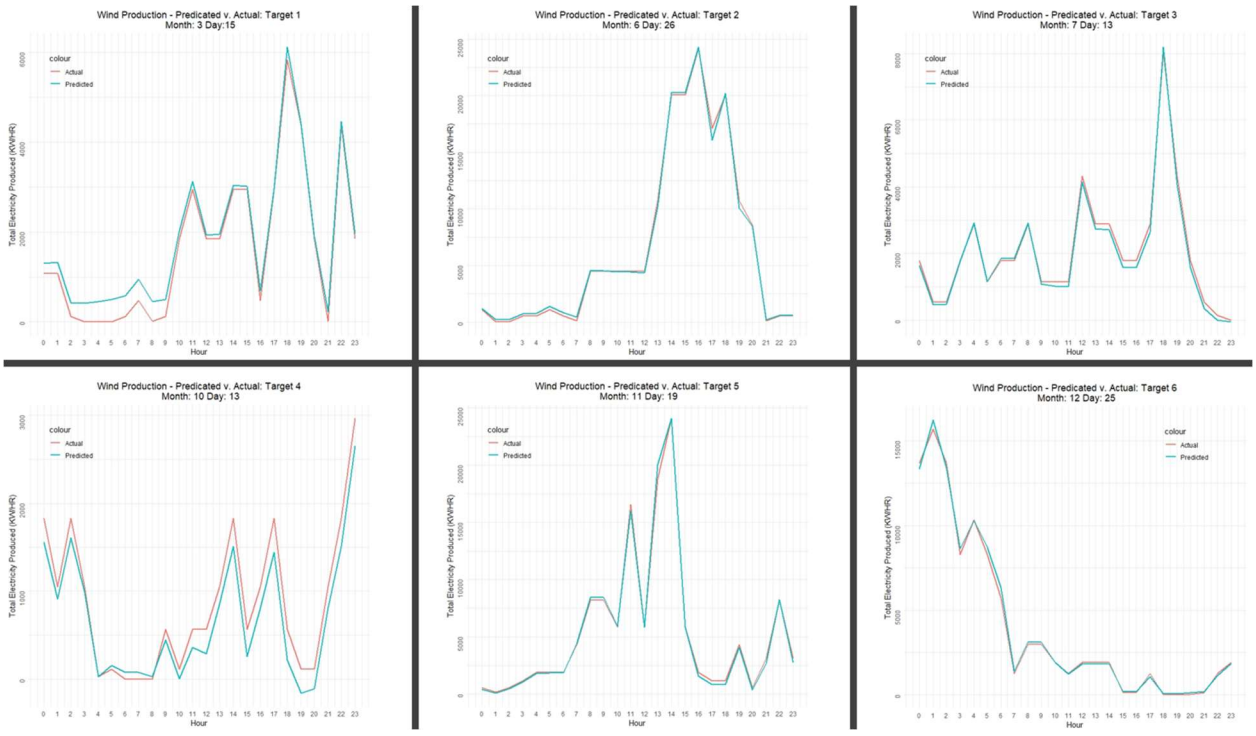
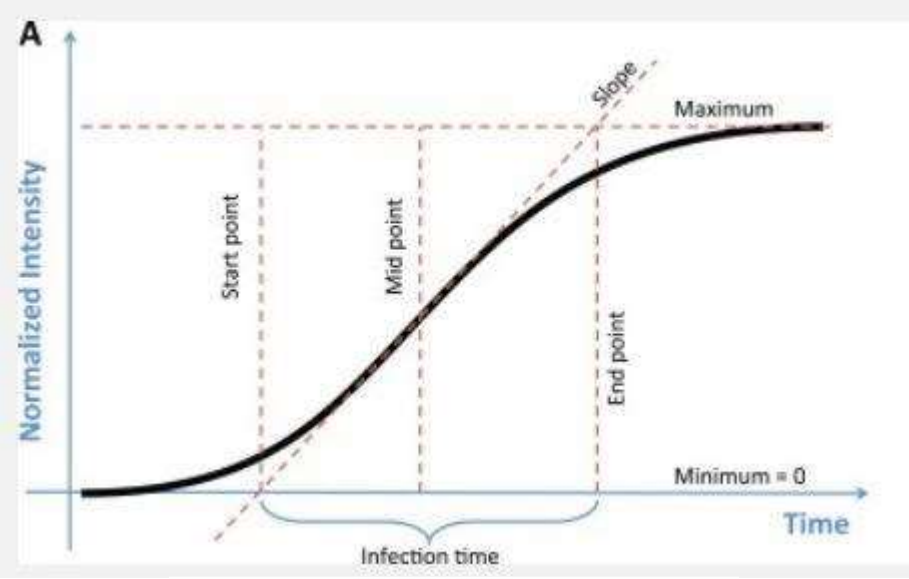


Figure 21: Wind Electricity Production Predictions on Scenario Dataset using XGBoost



MODELS/METRICS

Figure 22: Model: Sigmoid regression model



$$I(t) = f_{\text{sig}}(t) = \frac{I_{\max}}{1 + \exp(-a_1(t - t_{\text{mid}}))}$$

Here, I(t) is the intensity time course, given as a function of time t. The three parameters to be fitted are Imax, tmid, and a1. The parameter Imax represents the maximum intensity observed, the parameter tmid indicates the time at which intensity has reached half of its maximum, and the parameter a1 is related to the slope of I(t) at t = tmid via the formula d/dt $I(t)t = t_{mid} = a1I_{max}/4$.

Figure 23: Metrics: R-Squared

$$R^2 = 1 - \frac{\sum_{t=1}^{n} (y_t - x_t)^2}{\sum_{t=1}^{n} (y_t - \bar{y})^2}$$

(metrics 1) R-Squared:

The R^2 is defined as follows, where n is the sample size; x_t is the predicted value; y_t is the observed value; and y-bar is the mean of y_t.:

Figure 24: Metrics, RMSE

$$RMSE = \sqrt{\frac{1}{n} \times \sum_{t=1}^{n} (x_t - y_t)^2}$$

(Metrics 2) RMSE:

The mathematical formula for RMSE

Figure 25: Metrics, MAE

$$MAE = \frac{\sum_{t=1}^{n} |x_t - y_t|}{n}$$

(Metrics 3) MAE:

The mathematical formula for MAE

Figure 26: Metrics: Performance Index (PI)

$$PI_a = \frac{1}{3} * \frac{R^2_{min}}{R^2_a} + \frac{1}{3} * \frac{RMSE_a}{RMSE_{max}} + \frac{1}{3} * \frac{MAE_a}{MAE_{max}}$$

(Metrics 4) Performance Index (PI):

PI used in this paper is defined by the formula:

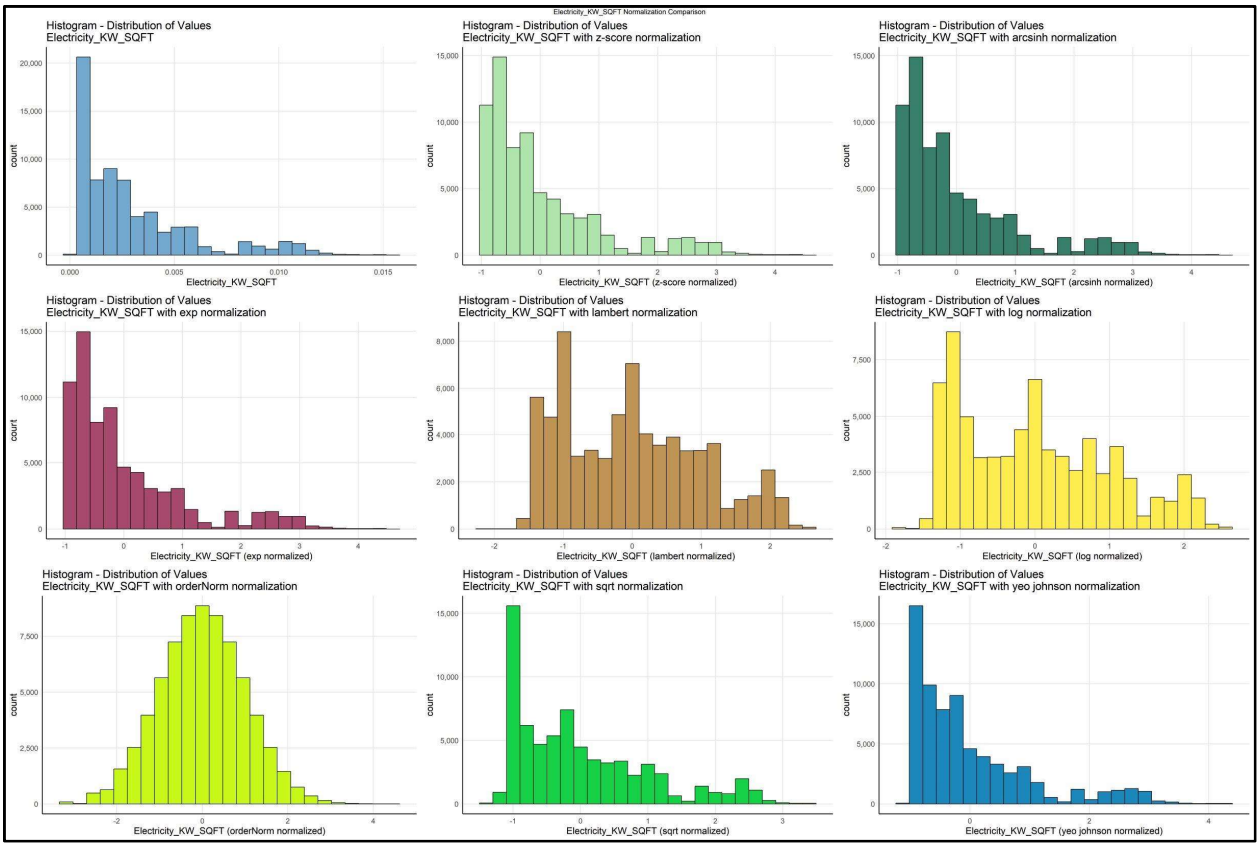
where PI_a, R^2_a, $RMSE_a$, and MAE_a are the PI, R^2, RMSE, and MAE of the prediction model a; R^2 min is the minimum R^2 in the comparison, and $RMSE_{max}$ and MAE_{max} are the maximum RMSE and MAE in the comparison, respectively. The PI applies standardization against the three evaluation indices to eliminate the magnitude of differences by comparing each model performance with the worst index performance among the comparison.

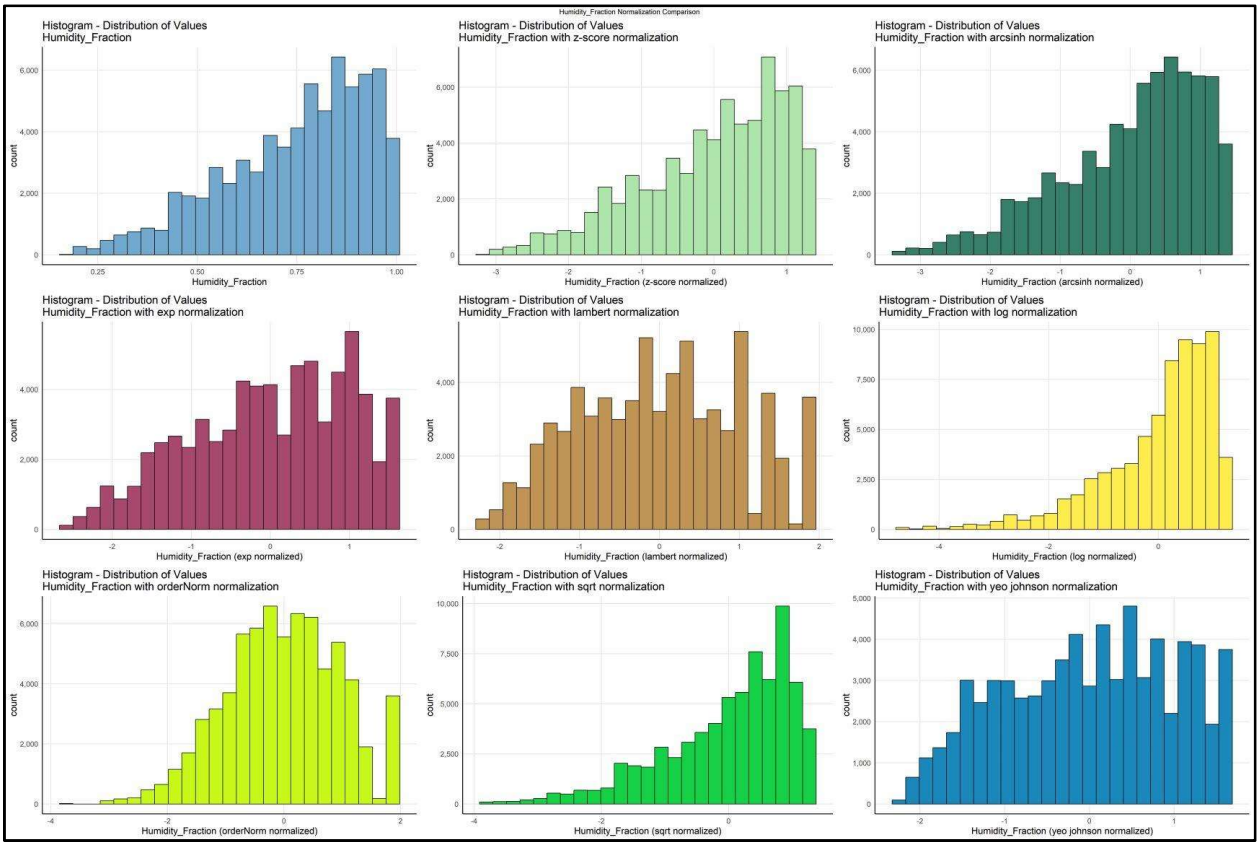
Actual versus Predicted Testing Energy Consumption Data
Consumption

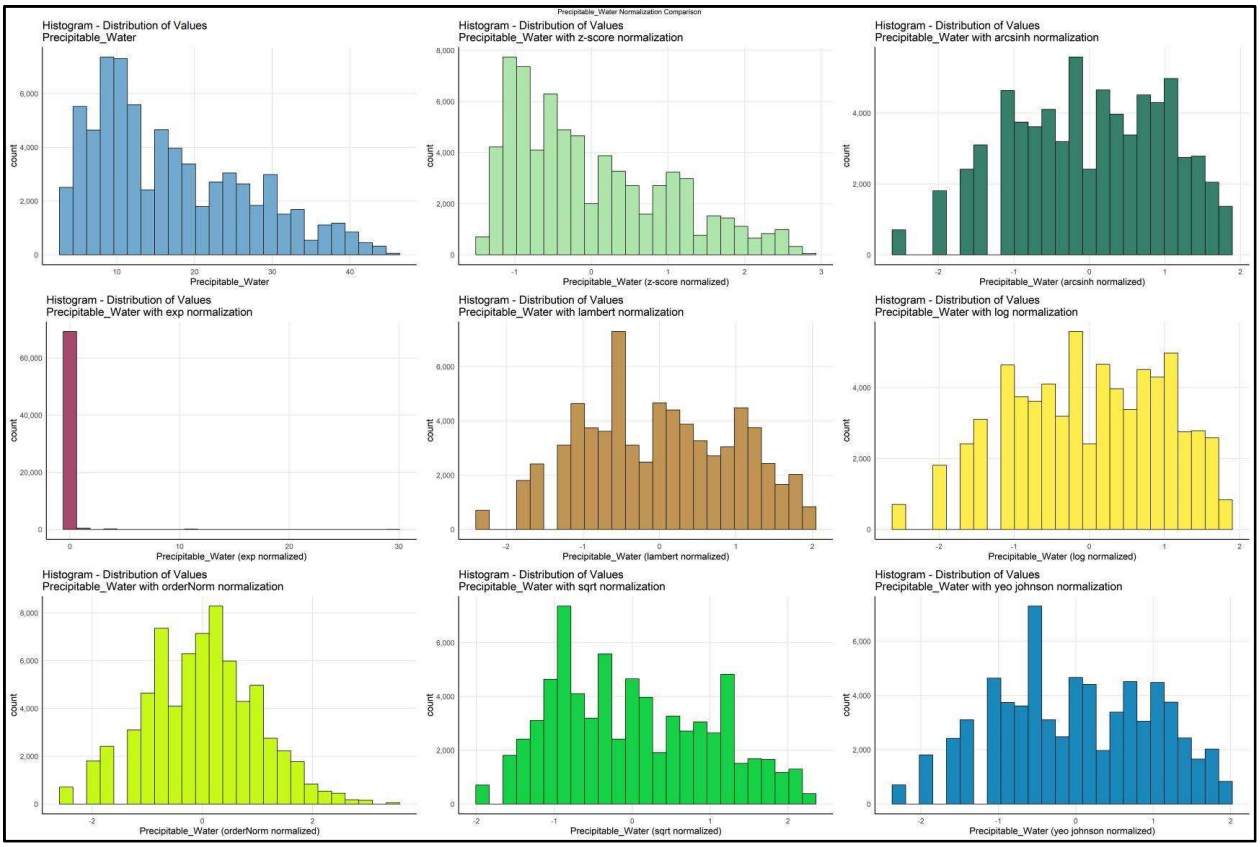
Table 8: Consumption, Summary Statistics

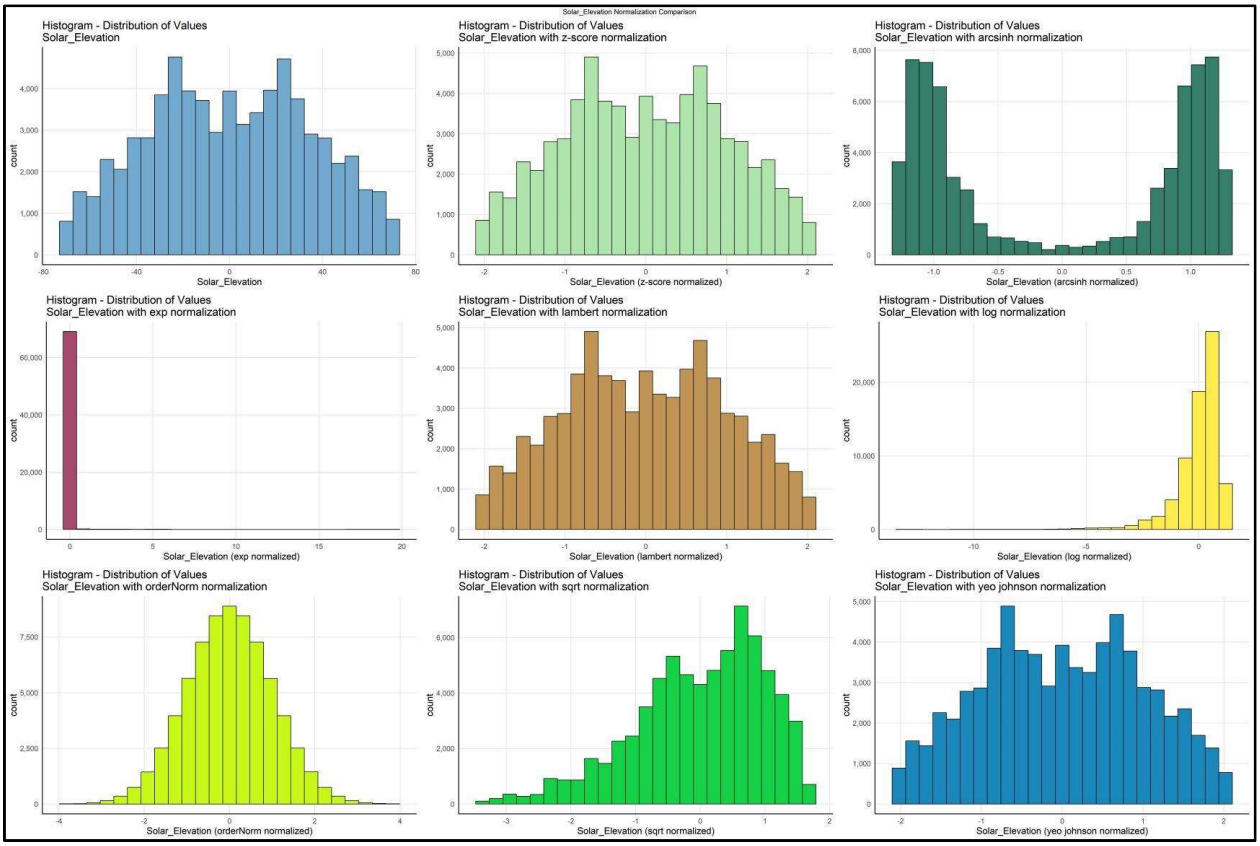
	Mean	Std.Dev	Min	Median	Max	N.Valid	Pct.Valid
Cloud_Cover_Fraction	0.64	0.40	0.00	0.80	1.00	70072	100.00
Dew_Point	3.94	10.60	-31.70	4.40	24.40	70072	100.00
Electricity_KW_SQFT	0.00	0.00	0.00	0.00	0.02	70072	100.00
Humidity_Fraction	0.75	0.18	0.18	0.78	1.00	70072	100.00
Precipitable_Water	16.90	9.86	3.00	14.00	45.00	70072	100.00
Solar_Elevation	0.29	34.75	-70.14	0.11	70.27	70072	100.00
Temperature	8.74	11.57	-27.20	8.90	35.60	70072	100.00
Visibility	14.38	6.92	0.00	16.00	32.20	70072	100.00

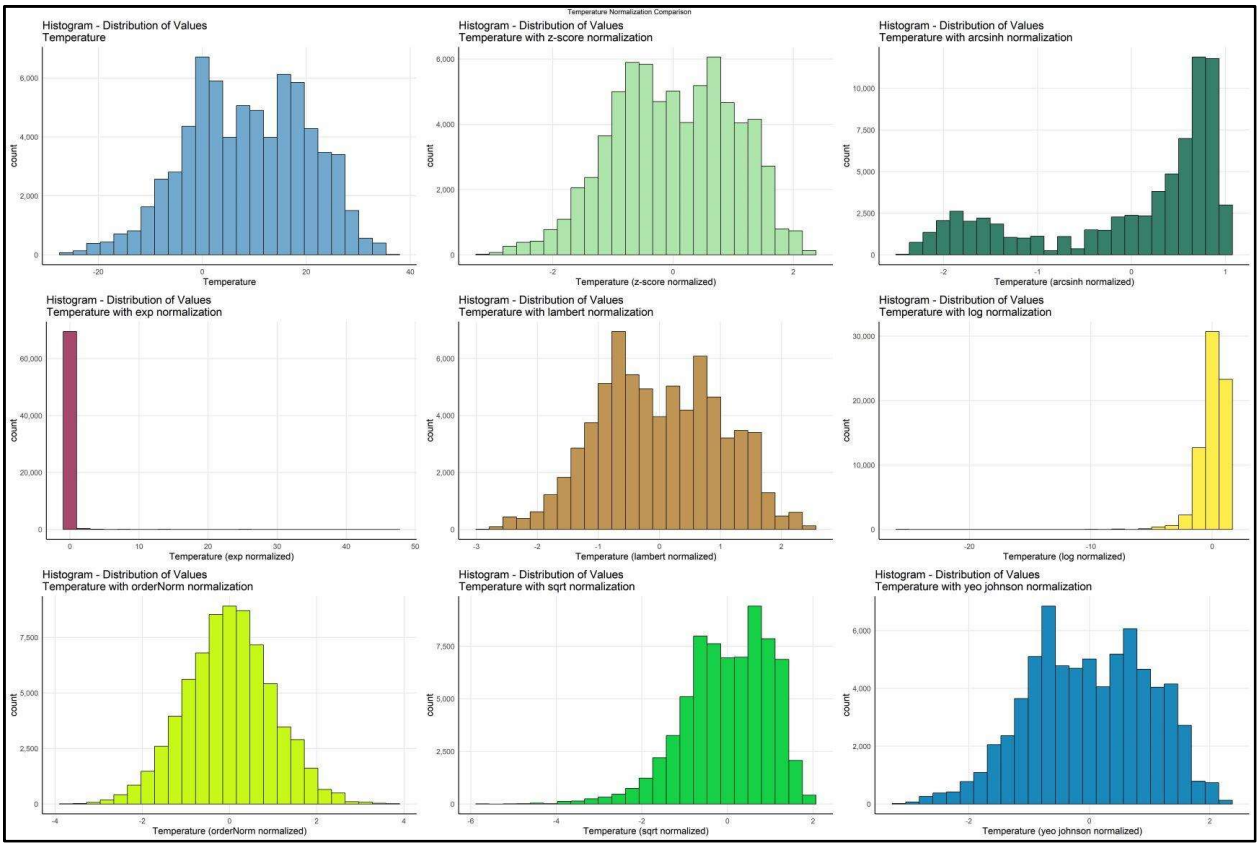
Figure 27: Consumption, Histogram Normalization Comparison

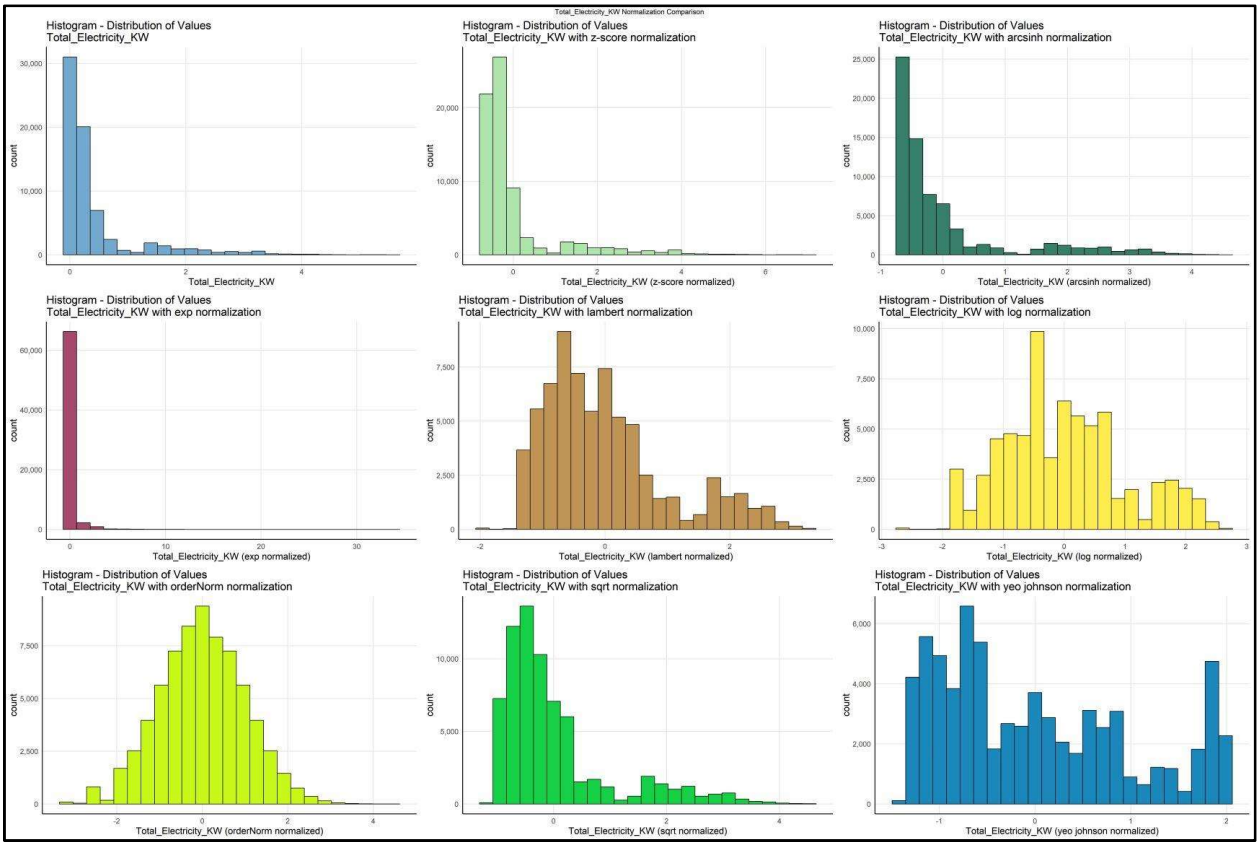

Electricity_KW_SQFT Normalization Comparison
Histogram - Distribution of Values
Electricity_KW_SQFT
Electricity_KW_SQFT
Histogram - Distribution of Values
Electricity_KW_SQFT with z-score normalization
Electricity_KW_SQFT (z-score normalized)
Histogram - Distribution of Values
Electricity_KW_SQFT with arcsinh normalization
Electricity_KW_SQFT (arcsinh normalized)
Histogram - Distribution of Values
Electricity_KW_SQFT with exp normalization
Electricity_KW_SQFT (exp normalized)
Histogram - Distribution of Values
Electricity_KW_SQFT with lambert normalization
Electricity_KW_SQFT (lambert normalized)
Histogram - Distribution of Values
Electricity_KW_SQFT with log normalization
Electricity_KW_SQFT (log normalized)
Histogram - Distribution of Values
Electricity_KW_SQFT with orderNorm normalization
Electricity_KW_SQFT (orderNorm normalized)
Histogram - Distribution of Values
Electricity_KW_SQFT with sqrt normalization
Electricity_KW_SQFT (sqrt normalized)
Histogram - Distribution of Values
Electricity_KW_SQFT with yeo johnson normalization
Electricity_KW_SQFT (yeo johnson normalized)
count


Humidity_Fraction Normalization Comparison
Histogram - Distribution of Values
Humidity_Fraction
Humidity_Fraction
Histogram - Distribution of Values
Humidity_Fraction with z-score normalization
Humidity_Fraction (z-score normalized)
Histogram - Distribution of Values
Humidity_Fraction with arcsinh normalization
Humidity_Fraction (arcsinh normalized)
Histogram - Distribution of Values
Humidity_Fraction with exp normalization
Humidity_Fraction (exp normalized)
Histogram - Distribution of Values
Humidity_Fraction with lambert normalization
Humidity_Fraction (lambert normalized)
Histogram - Distribution of Values
Humidity_Fraction with log normalization
Humidity_Fraction (log normalized)
Histogram - Distribution of Values
Humidity_Fraction with orderNorm normalization
Humidity_Fraction (orderNorm normalized)
Histogram - Distribution of Values
Humidity_Fraction with sqrt normalization
Humidity_Fraction (sqrt normalized)
Histogram - Distribution of Values
Humidity_Fraction with yeo johnson normalization
Humidity_Fraction (yeo johnson normalized)
count


Precipitable_Water Normalization Comparison
Histogram - Distribution of Values
Precipitable_Water
Precipitable_Water
Histogram - Distribution of Values
Precipitable_Water with z-score normalization
Precipitable_Water (z-score normalized)
Histogram - Distribution of Values
Precipitable_Water with arcsinh normalization
Precipitable_Water (arcsinh normalized)
Histogram - Distribution of Values
Precipitable_Water with exp normalization
Precipitable_Water (exp normalized)
Histogram - Distribution of Values
Precipitable_Water with lambert normalization
Precipitable_Water (lambert normalized)
Histogram - Distribution of Values
Precipitable_Water with log normalization
Precipitable_Water (log normalized)
Histogram - Distribution of Values
Precipitable_Water with orderNorm normalization
Precipitable_Water (orderNorm normalized)
Histogram - Distribution of Values
Precipitable_Water with sqrt normalization
Precipitable_Water (sqrt normalized)
Histogram - Distribution of Values
Precipitable_Water with yeo johnson normalization
Precipitable_Water (yeo johnson normalized)
count


Solar_Elevation Normalization Comparison
Histogram - Distribution of Values
Solar_Elevation
Solar_Elevation
Histogram - Distribution of Values
Solar_Elevation with z-score normalization
Solar_Elevation (z-score normalized)
Histogram - Distribution of Values
Solar_Elevation with arcsinh normalization
Solar_Elevation (arcsinh normalized)
Histogram - Distribution of Values
Solar_Elevation with exp normalization
Solar_Elevation (exp normalized)
Histogram - Distribution of Values
Solar_Elevation with lambert normalization
Solar_Elevation (lambert normalized)
Histogram - Distribution of Values
Solar_Elevation with log normalization
Solar_Elevation (log normalized)
Histogram - Distribution of Values
Solar_Elevation with orderNorm normalization
Solar_Elevation (orderNorm normalized)
Histogram - Distribution of Values
Solar_Elevation with sqrt normalization
Solar_Elevation (sqrt normalized)
Histogram - Distribution of Values
Solar_Elevation with yeo johnson normalization
Solar_Elevation (yeo johnson normalized)
count


Temperature Normalization Comparison
Histogram - Distribution of Values
Temperature
Temperature
Histogram - Distribution of Values
Temperature with z-score normalization
Temperature (z-score normalized)
Histogram - Distribution of Values
Temperature with arcsinh normalization
Temperature (arcsinh normalized)
Histogram - Distribution of Values
Temperature with exp normalization
Temperature (exp normalized)
Histogram - Distribution of Values
Temperature with lambert normalization
Temperature (lambert normalized)
Histogram - Distribution of Values
Temperature with log normalization
Temperature (log normalized)
Histogram - Distribution of Values
Temperature with orderNorm normalization
Temperature (orderNorm normalized)
Histogram - Distribution of Values
Temperature with sqrt normalization
Temperature (sqrt normalized)
Histogram - Distribution of Values
Temperature with yeo johnson normalization
Temperature (yeo johnson normalized)
count


Total_Electricity_KW Normalization Comparison
Histogram - Distribution of Values
Total_Electricity_KW
Total_Electricity_KW
Histogram - Distribution of Values
Total_Electricity_KW with z-score normalization
Total_Electricity_KW (z-score normalized)
Histogram - Distribution of Values
Total_Electricity_KW with arcsinh normalization
Total_Electricity_KW (arcsinh normalized)
Histogram - Distribution of Values
Total_Electricity_KW with exp normalization
Total_Electricity_KW (exp normalized)
Histogram - Distribution of Values
Total_Electricity_KW with lambert normalization
Total_Electricity_KW (lambert normalized)
Histogram - Distribution of Values
Total_Electricity_KW with log normalization
Total_Electricity_KW (log normalized)
Histogram - Distribution of Values
Total_Electricity_KW with orderNorm normalization
Total_Electricity_KW (orderNorm normalized)
Histogram - Distribution of Values
Total_Electricity_KW with sqrt normalization
Total_Electricity_KW (sqrt normalized)
Histogram - Distribution of Values
Total_Electricity_KW with yeo johnson normalization
Total_Electricity_KW (yeo johnson normalized)
count

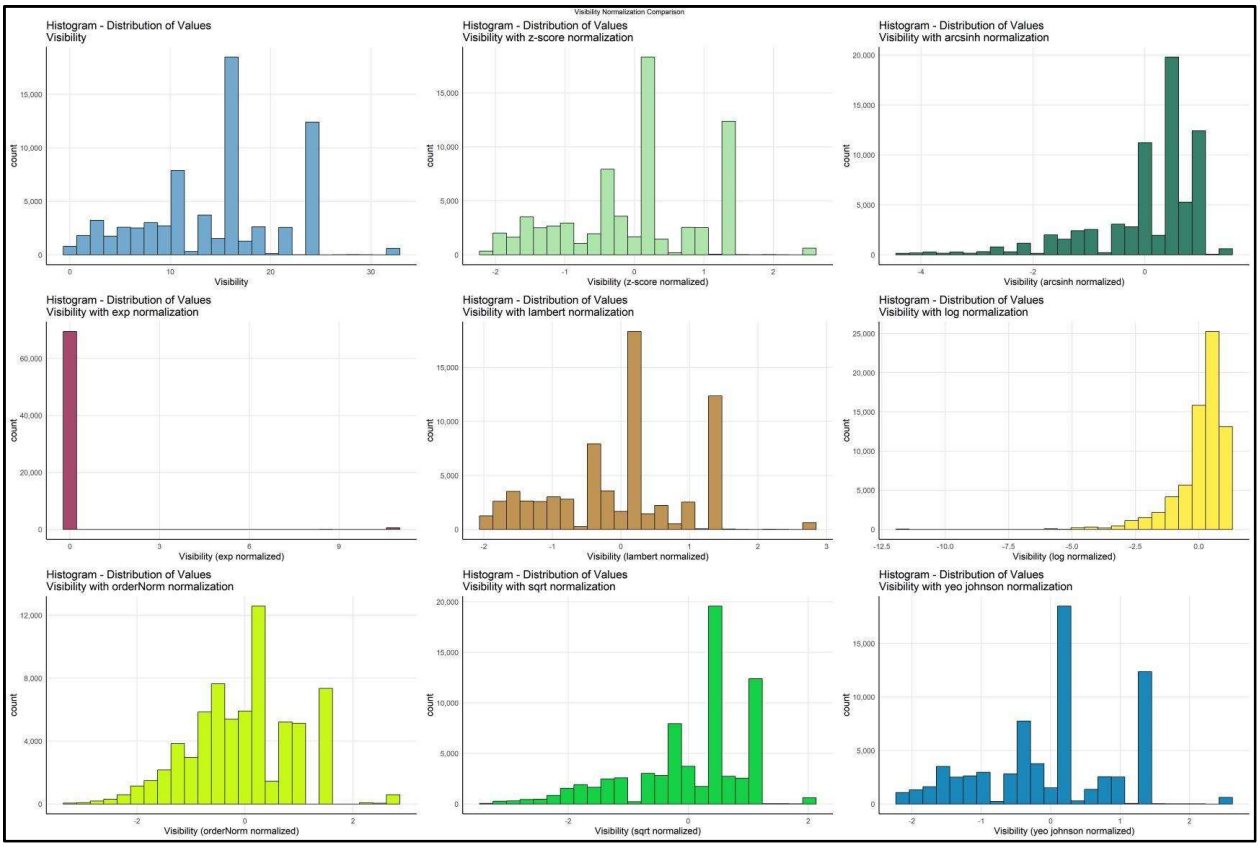
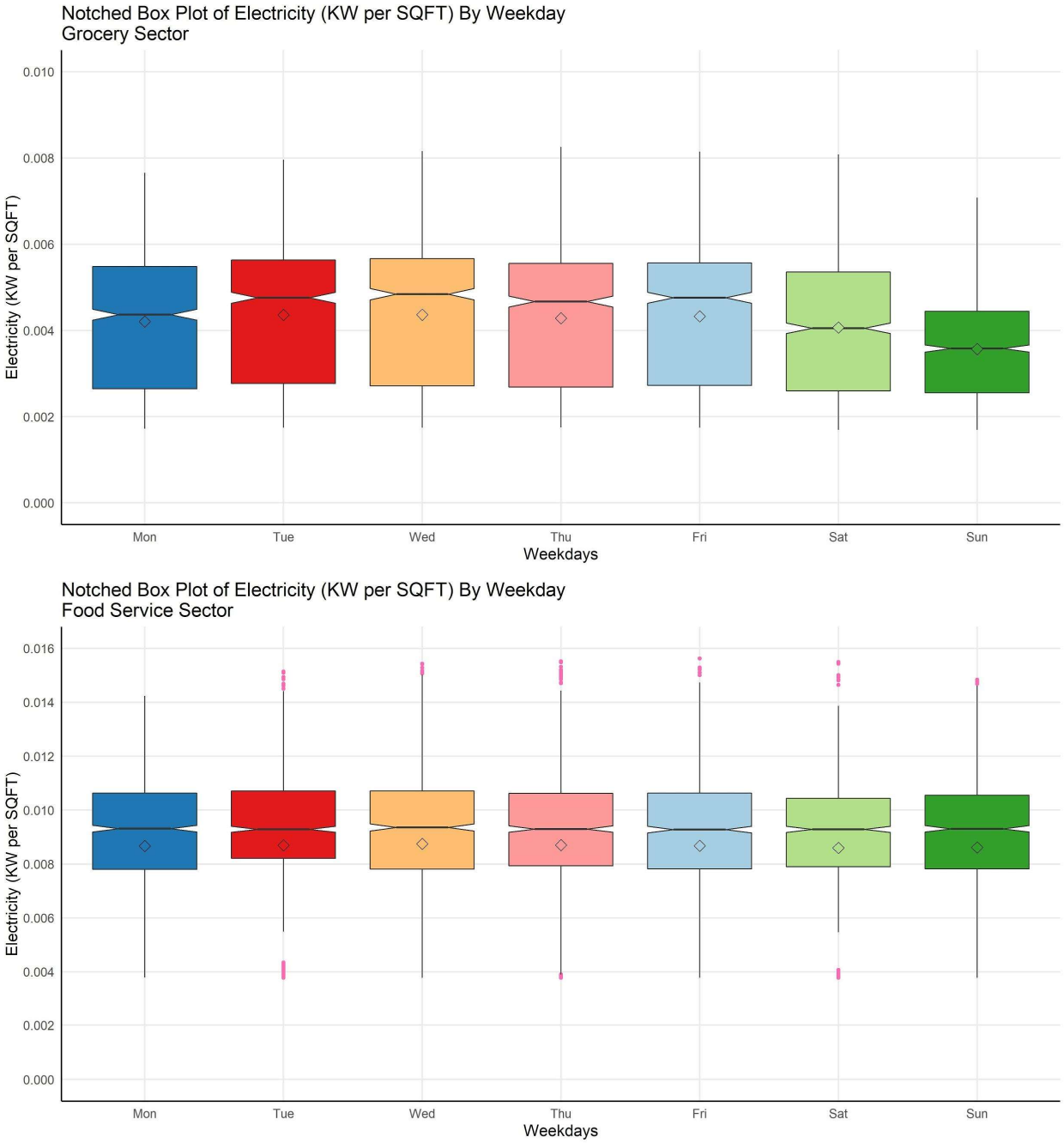
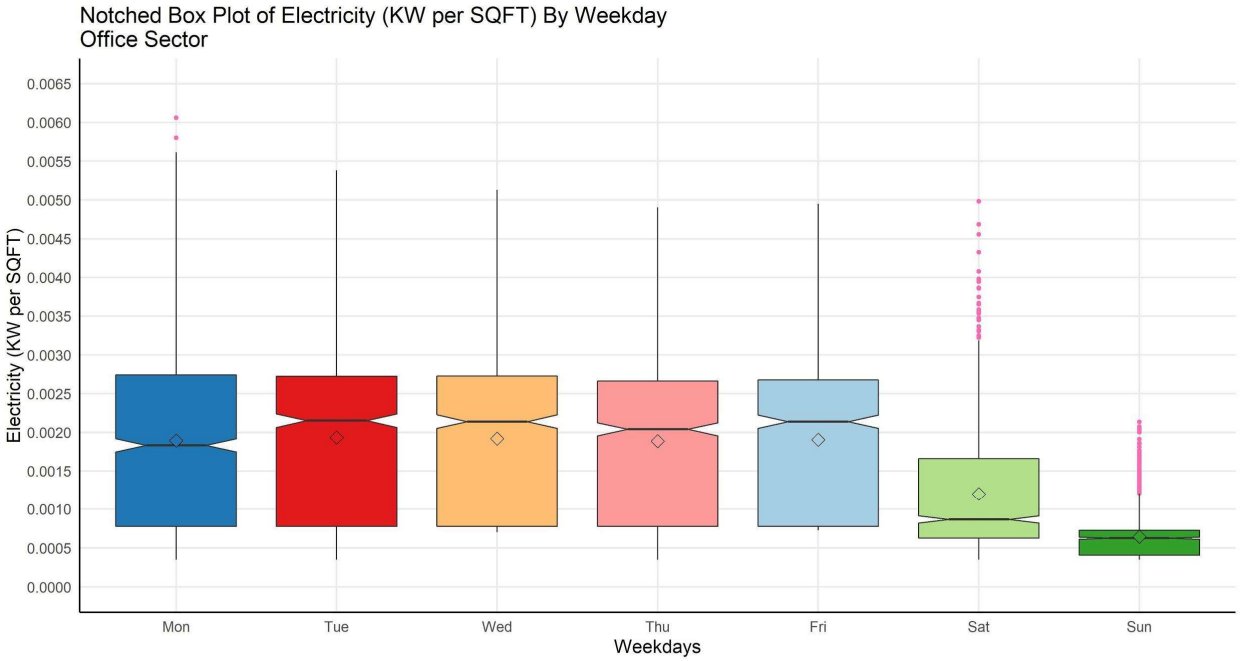
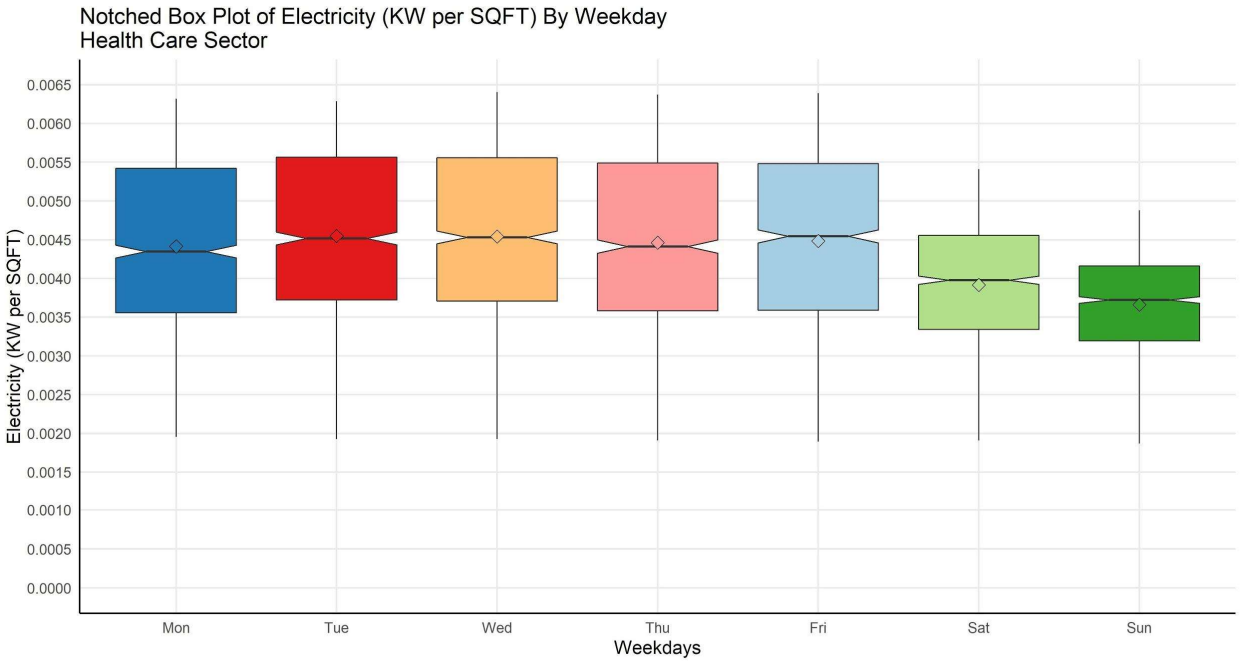


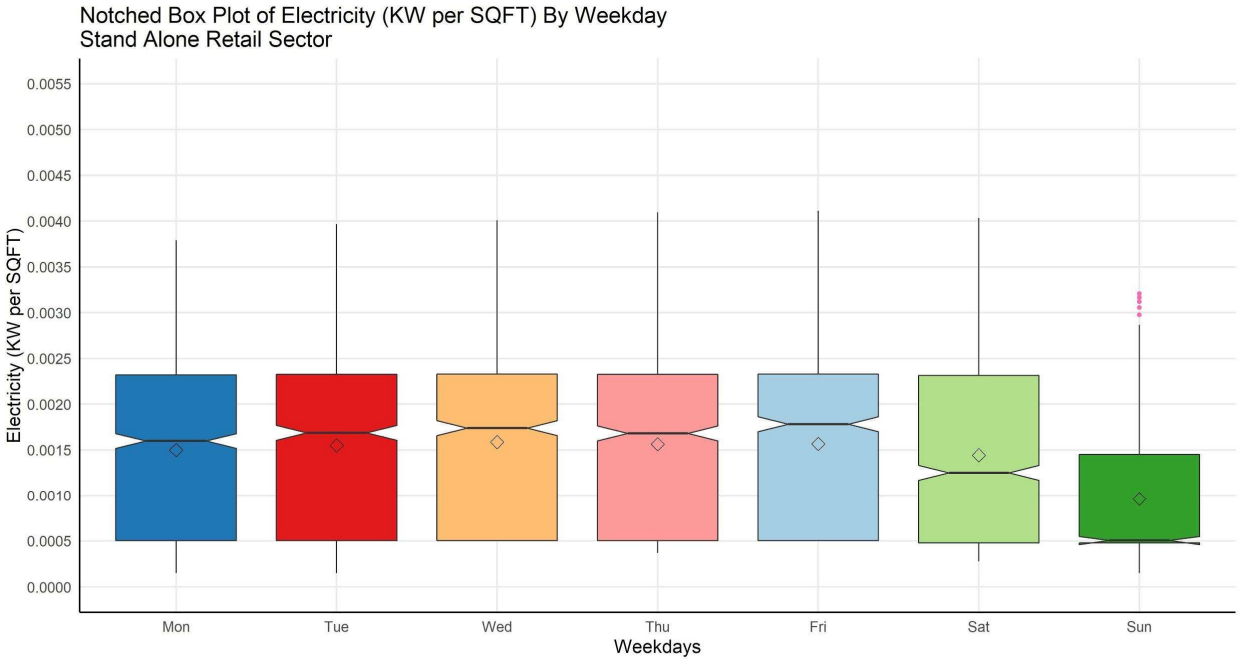
Figure 3 Notched Box Plots by Weekday by Sector

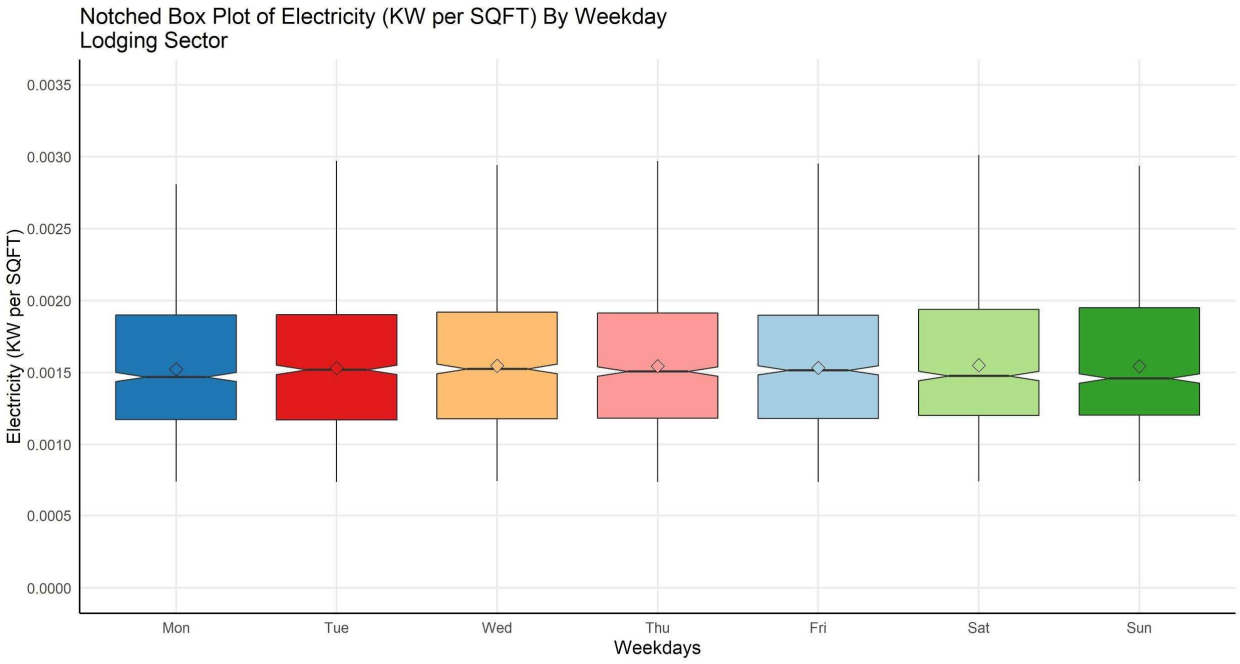
Figure 28: Consumption, Box Plots - Electricity Consumption by Sector and Weekday

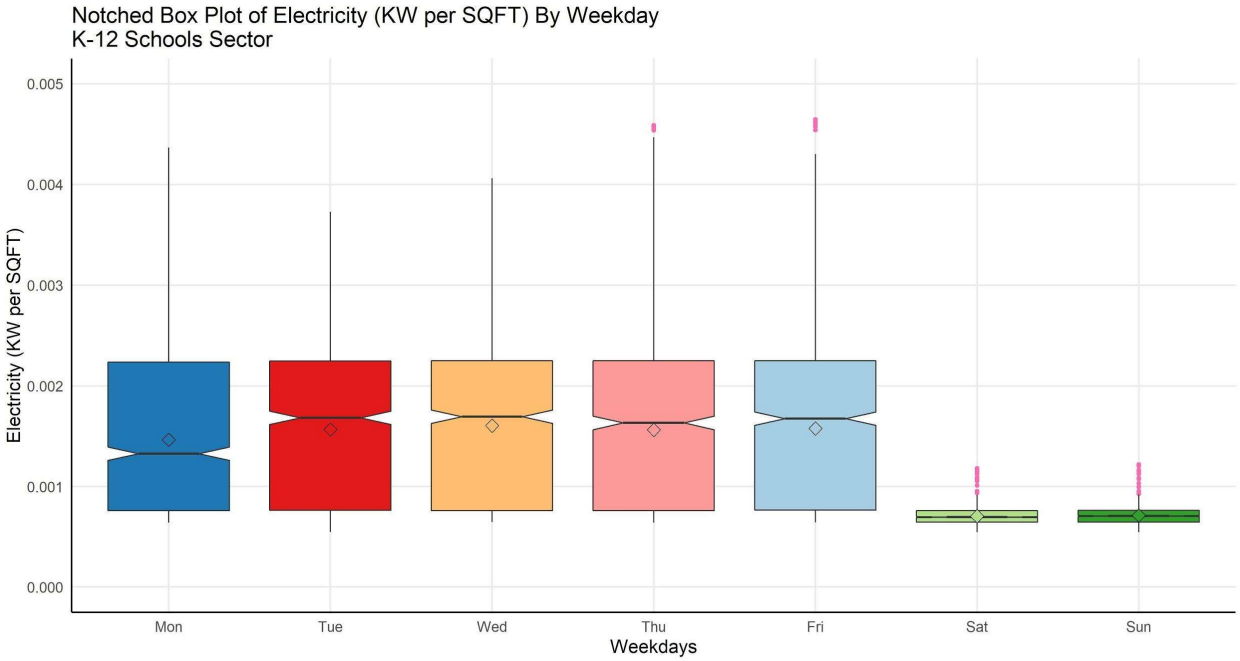



Notched Box Plot of Electricity (KW per SQFT) By Weekday
Office Sector
Electricity (KW per SQFT)
0.0065
0.0060
0.0055
0.0050
0.0045
0.0040
0.0035
0.0030
0.0025
0.0020
0.0015
0.0010
0.0005
0.0000
Mon
Tue
Wed
Thu
Fri
Sat
Sun
Weekdays


Notched Box Plot of Electricity (KW per SQFT) By Weekday
Health Care Sector
Electricity (KW per SQFT)
0.0065
0.0060
0.0055
0.0050
0.0045
0.0040
0.0035
0.0030
0.0025
0.0020
0.0015
0.0010
0.0005
0.0000
Mon
Tue
Wed
Thu
Fri
Sat
Sun
Weekdays


Notched Box Plot of Electricity (KW per SQFT) By Weekday
Stand Alone Retail Sector
Electricity (KW per SQFT)
0.0055
0.0050
0.0045
0.0040
0.0035
0.0030
0.0025
0.0020
0.0015
0.0010
0.0005
0.0000
Mon
Tue
Wed
Thu
Fri
Sat
Sun
Weekdays


Notched Box Plot of Electricity (KW per SQFT) By Weekday
Lodging Sector
Electricity (KW per SQFT)
0.0035
0.0030
0.0025
0.0020
0.0015
0.0010
0.0005
0.0000
Mon
Tue
Wed
Thu
Fri
Sat
Sun
Weekdays


Notched Box Plot of Electricity (KW per SQFT) By Weekday
K-12 Schools Sector
Electricity (KW per SQFT)
0.005
0.004
0.003
0.002
0.001
0.000
Mon
Tue
Wed
Thu
Fri
Sat
Sun
Weekdays

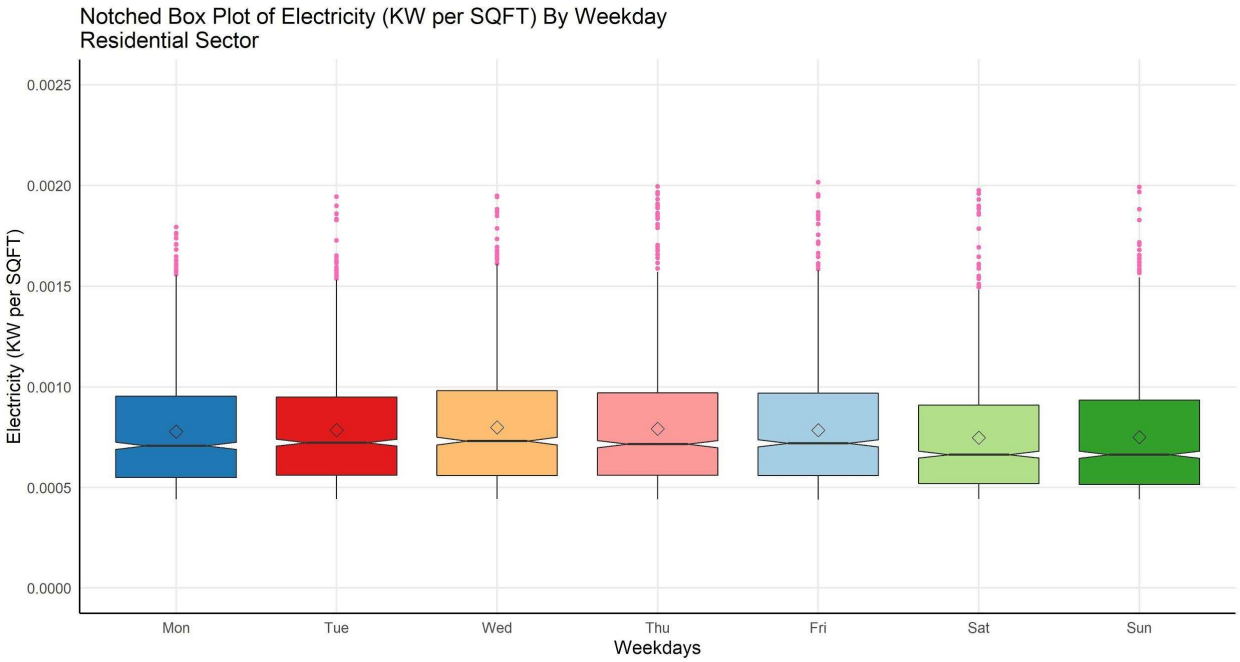
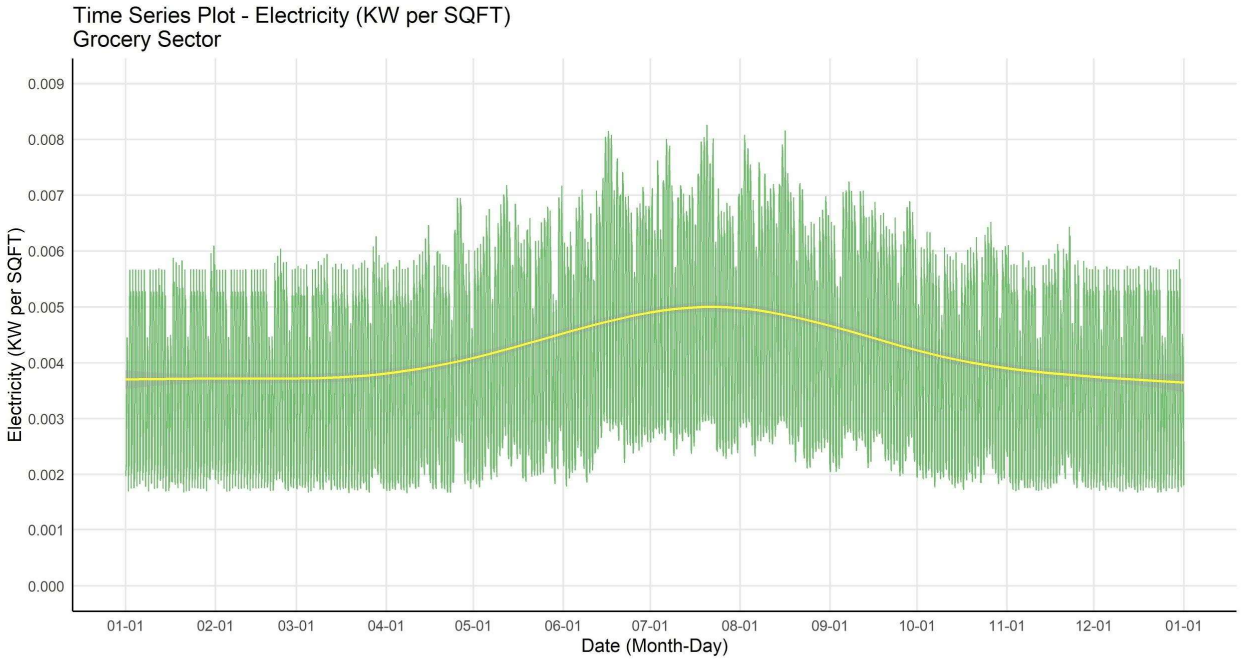
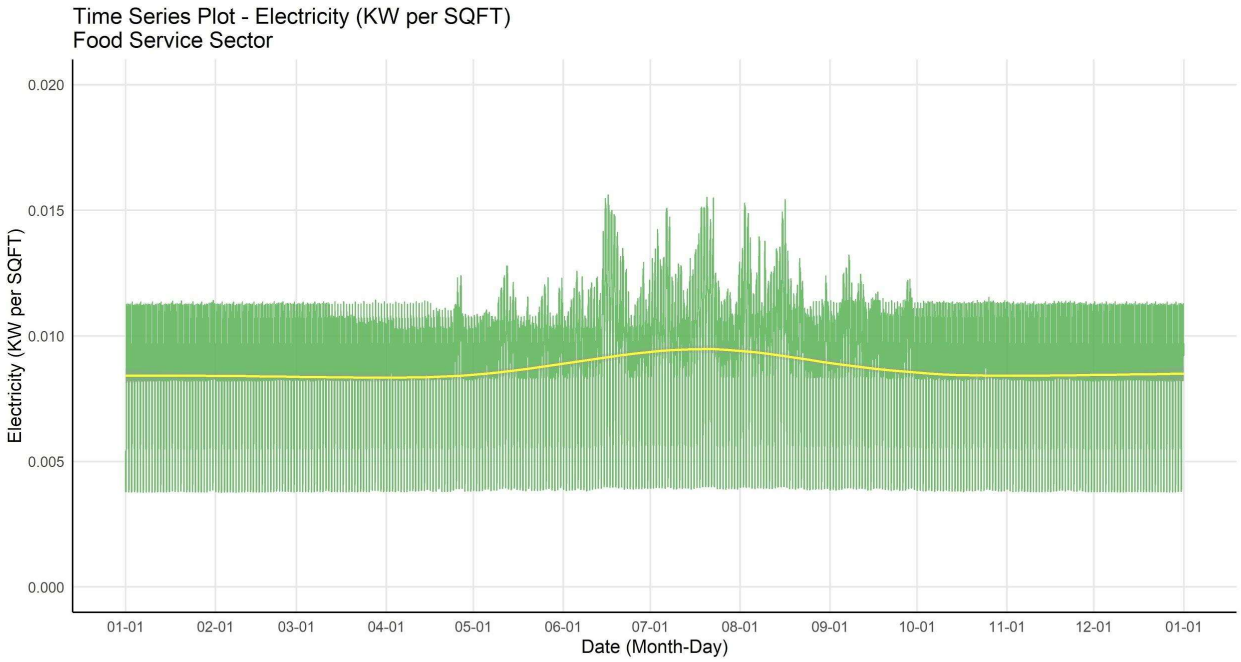
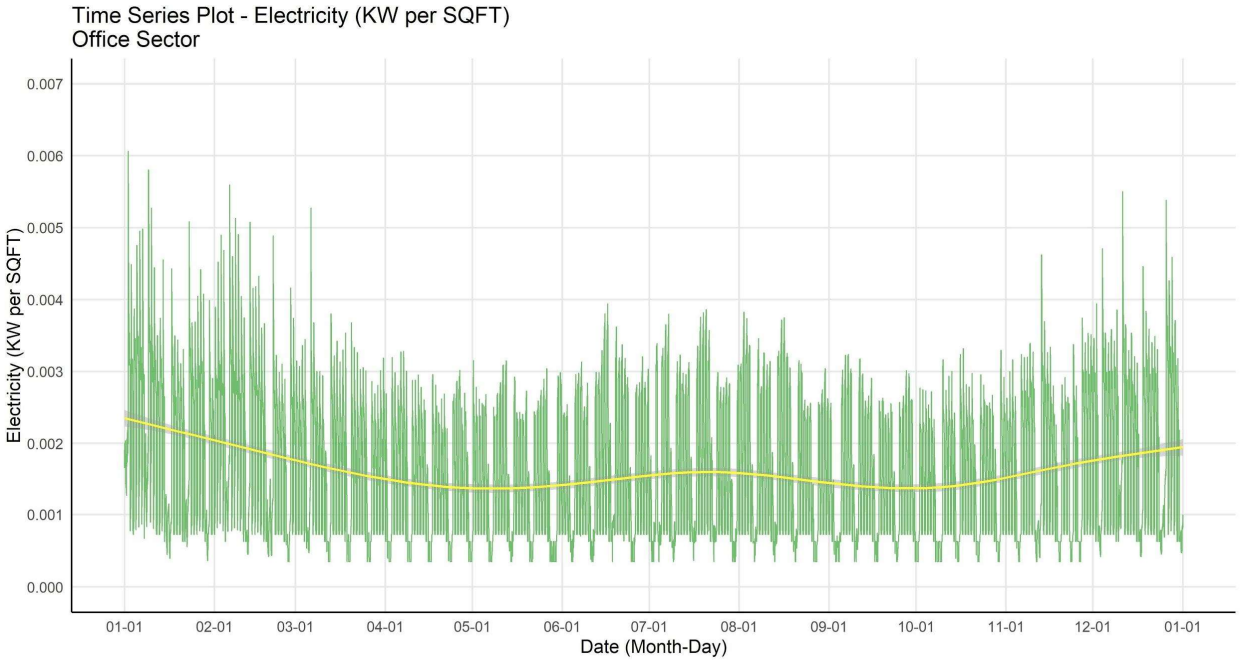
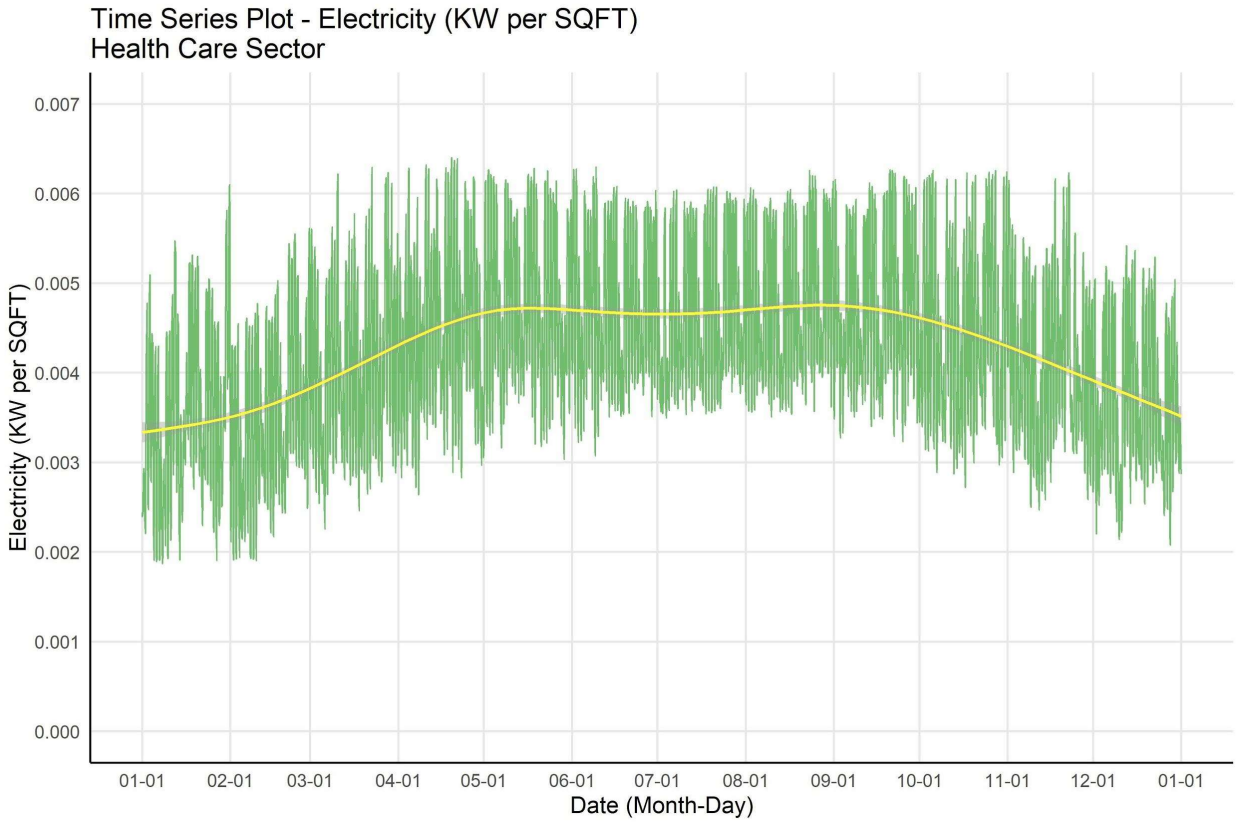

Notched Box Plot of Electricity (KW per SQFT) By Weekday
Residential Sector
Electricity (KW per SQFT)
0.0025
0.0020
0.0015
0.0010
0.0005
0.0000
Mon
Tue
Wed
Thu
Fri
Sat
Sun
Weekdays

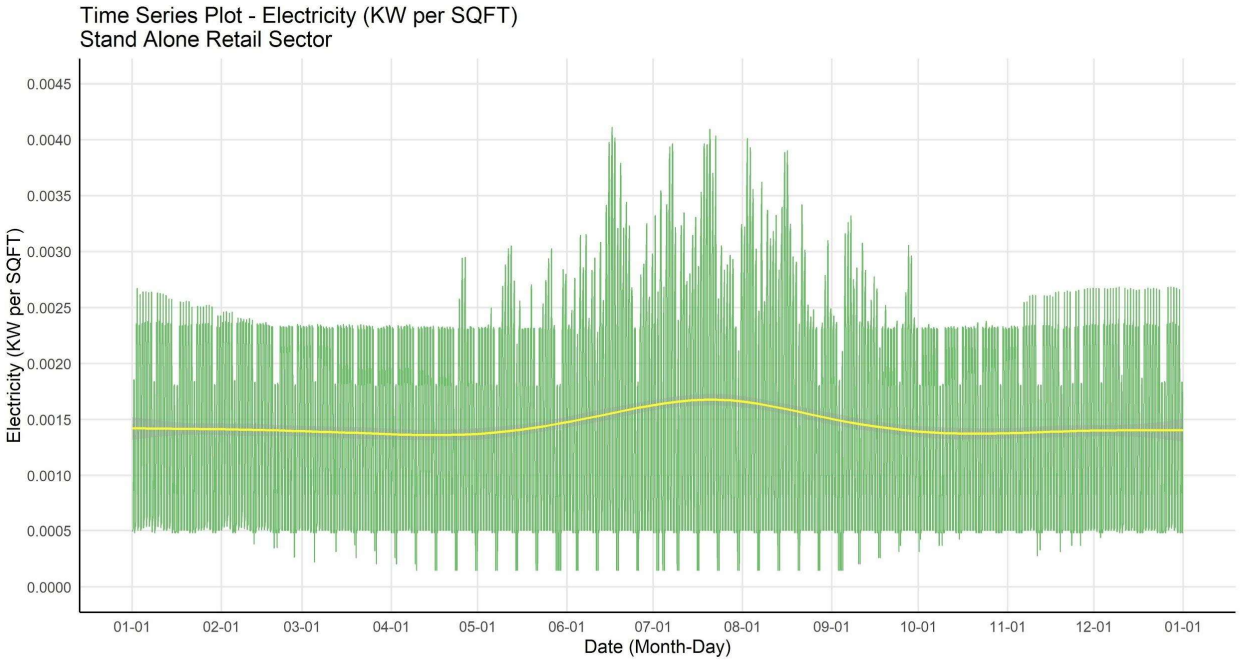
Figure 29: Consumption, Time Series Plots by Sector

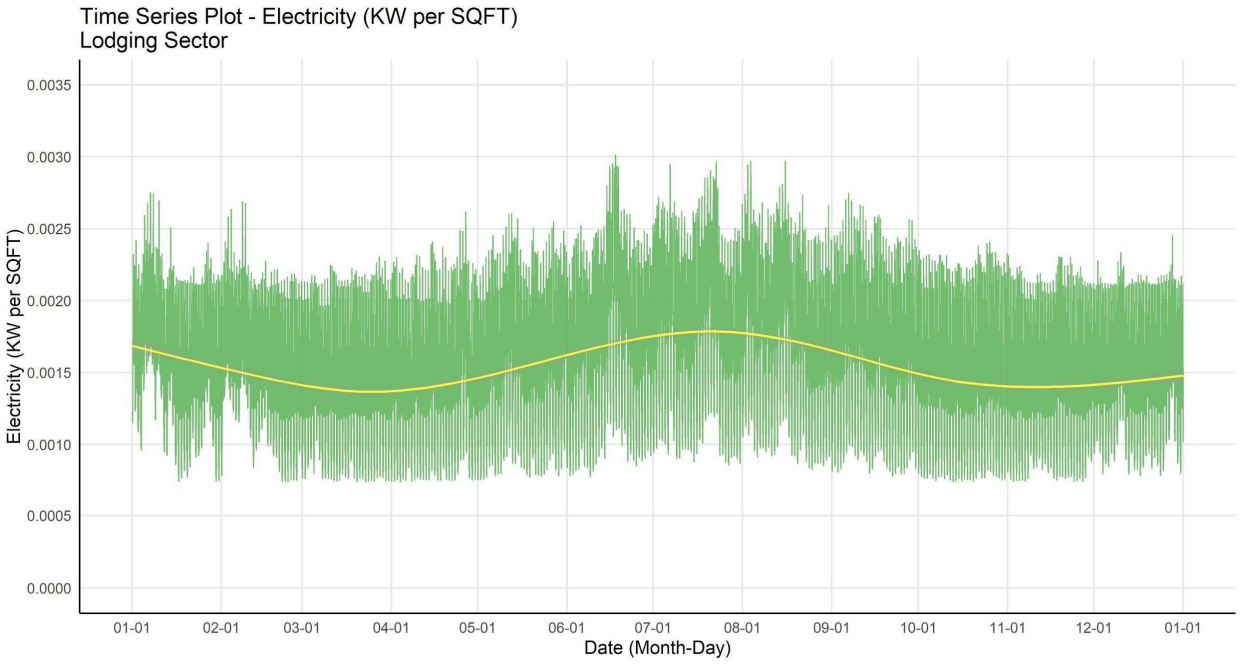


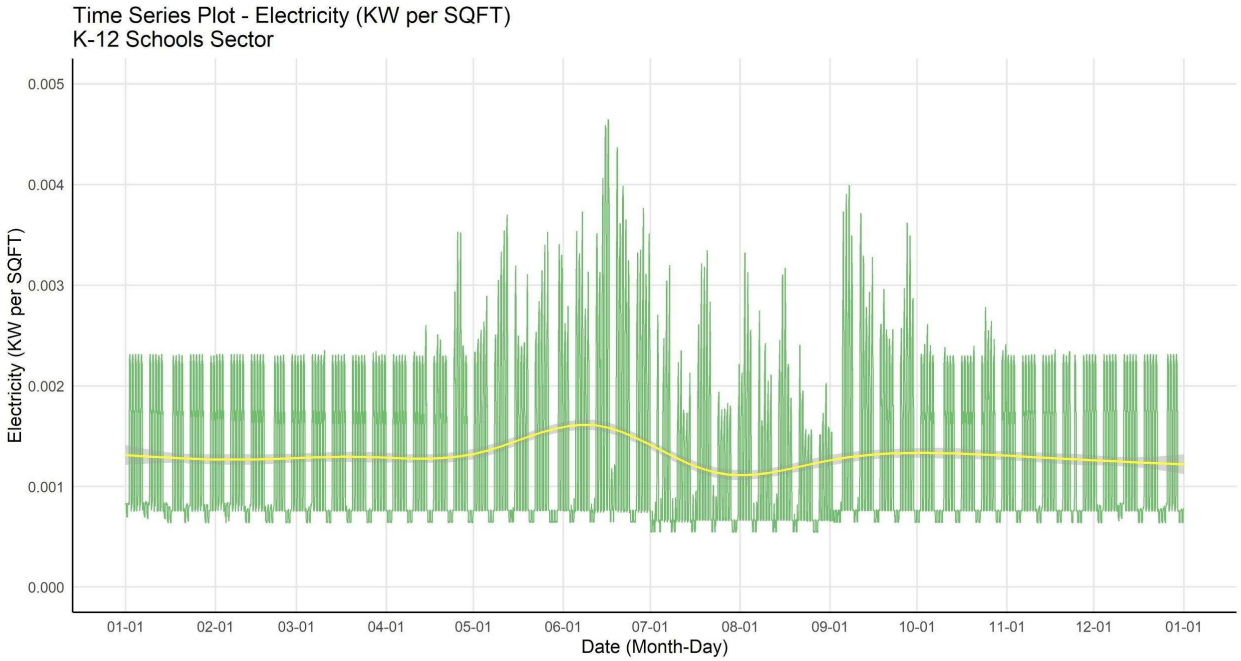


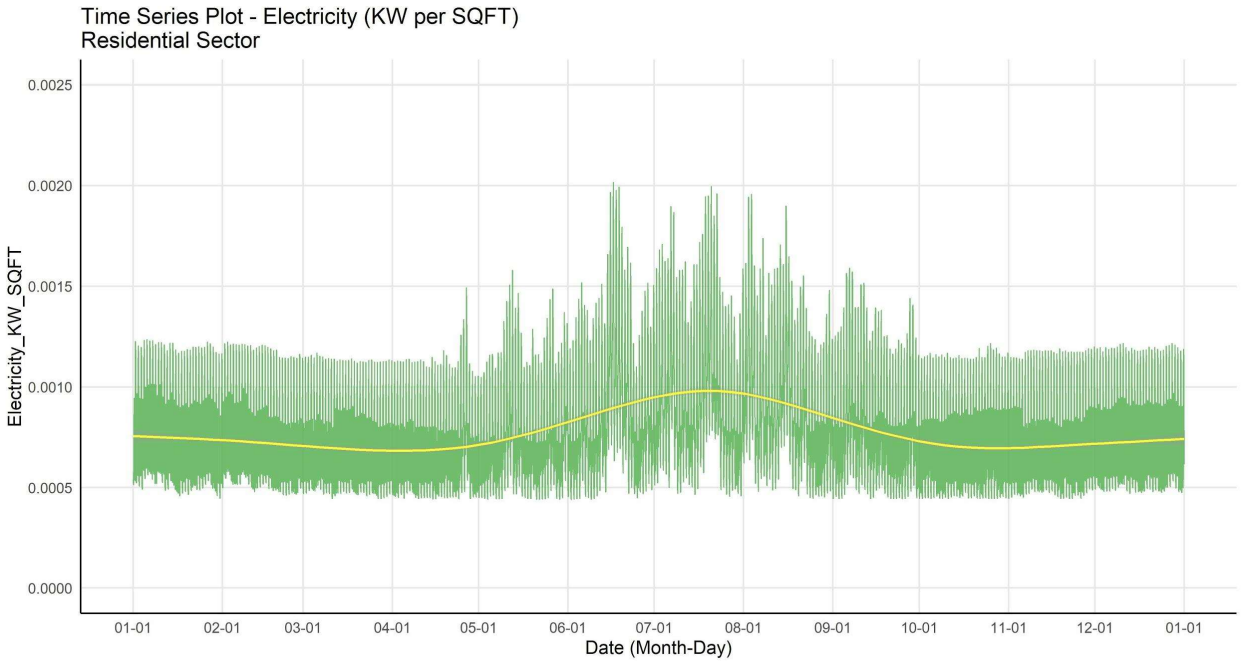

Time Series Plot - Electricity (KW per SQFT)
Office Sector
Electricity (KW per SQFT)
0.007
0.006
0.005
0.004
0.003
0.002
0.001
0.000
01-01
02-01
03-01
04-01
05-01
06-01
07-01
08-01
09-01
10-01
11-01
12-01
01-01
Date (Month-Day)


Time Series Plot - Electricity (KW per SQFT)
Health Care Sector
Electricity (KW per SQFT)
0.007
0.006
0.005
0.004
0.003
0.002
0.001
0.000
01-01
02-01
03-01
04-01
05-01
06-01
07-01
08-01
09-01
10-01
11-01
12-01
01-01
Date (Month-Day)


Time Series Plot - Electricity (KW per SQFT)
Stand Alone Retail Sector
Electricity (KW per SQFT)
0.0045
0.0040
0.0035
0.0030
0.0025
0.0020
0.0015
0.0010
0.0005
0.0000
01-01
02-01
03-01
04-01
05-01
06-01
07-01
08-01
09-01
10-01
11-01
12-01
01-01
Date (Month-Day)

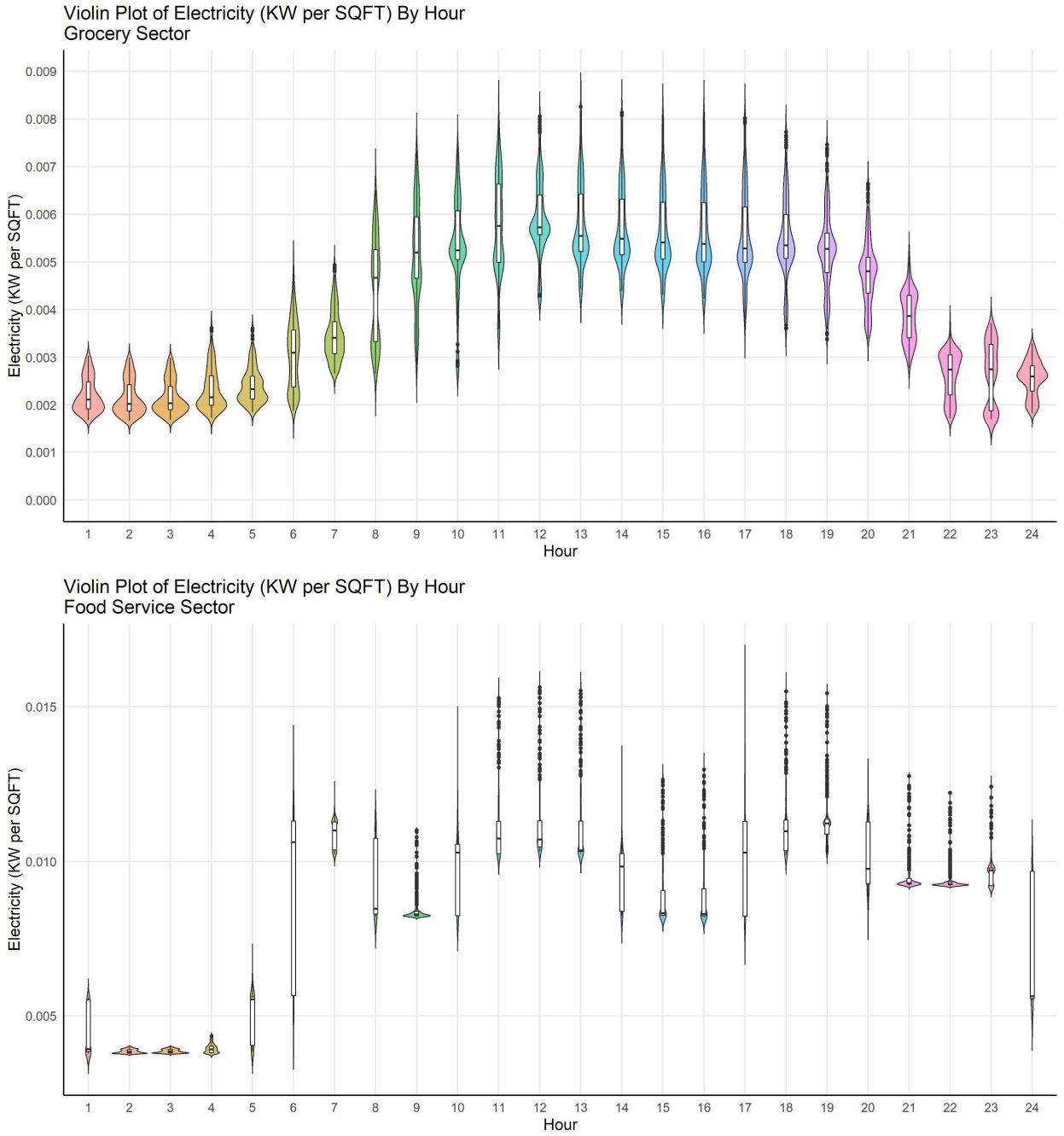

Time Series Plot - Electricity (KW per SQFT)
Lodging Sector
Electricity (KW per SQFT)
0.0035
0.0030
0.0025
0.0020
0.0015
0.0010
0.0005
0.0000
01-01
02-01
03-01
04-01
05-01
06-01
07-01
08-01
09-01
10-01
11-01
12-01
01-01
Date (Month-Day)

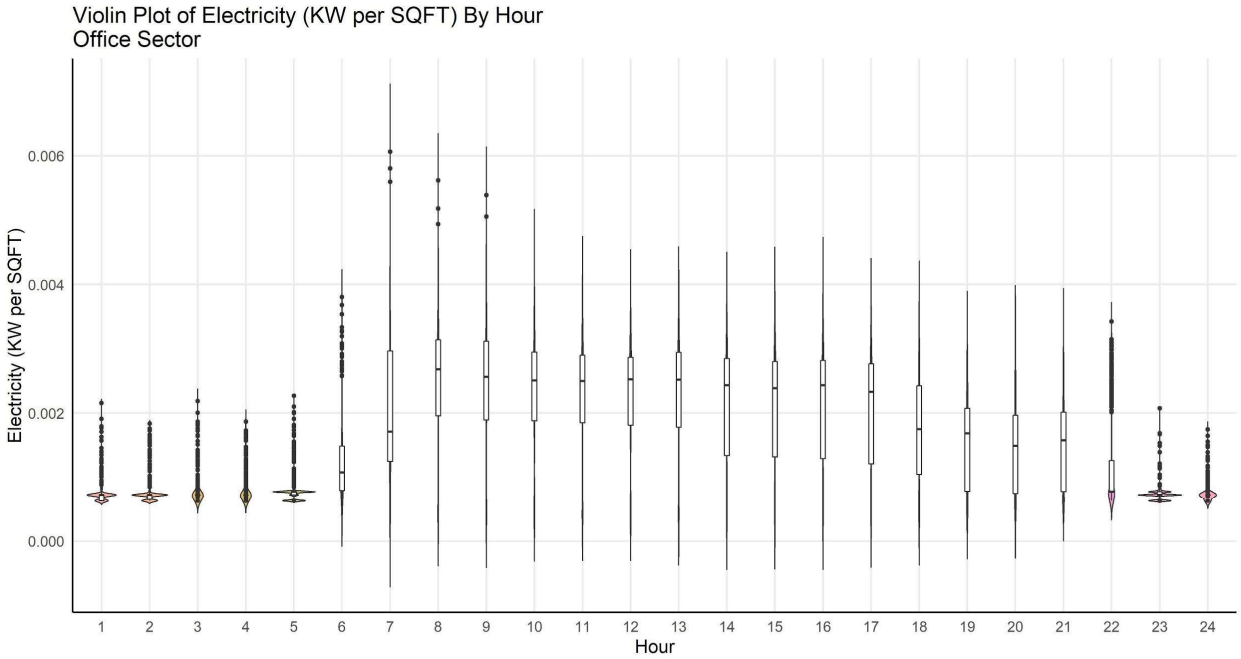


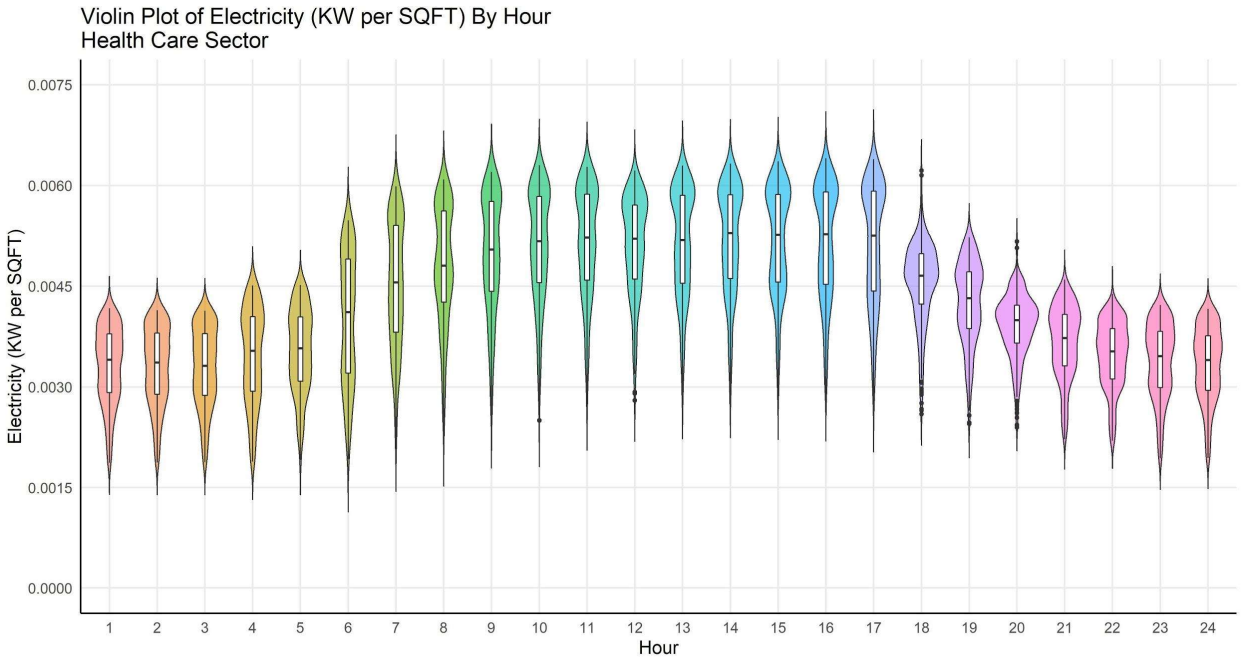


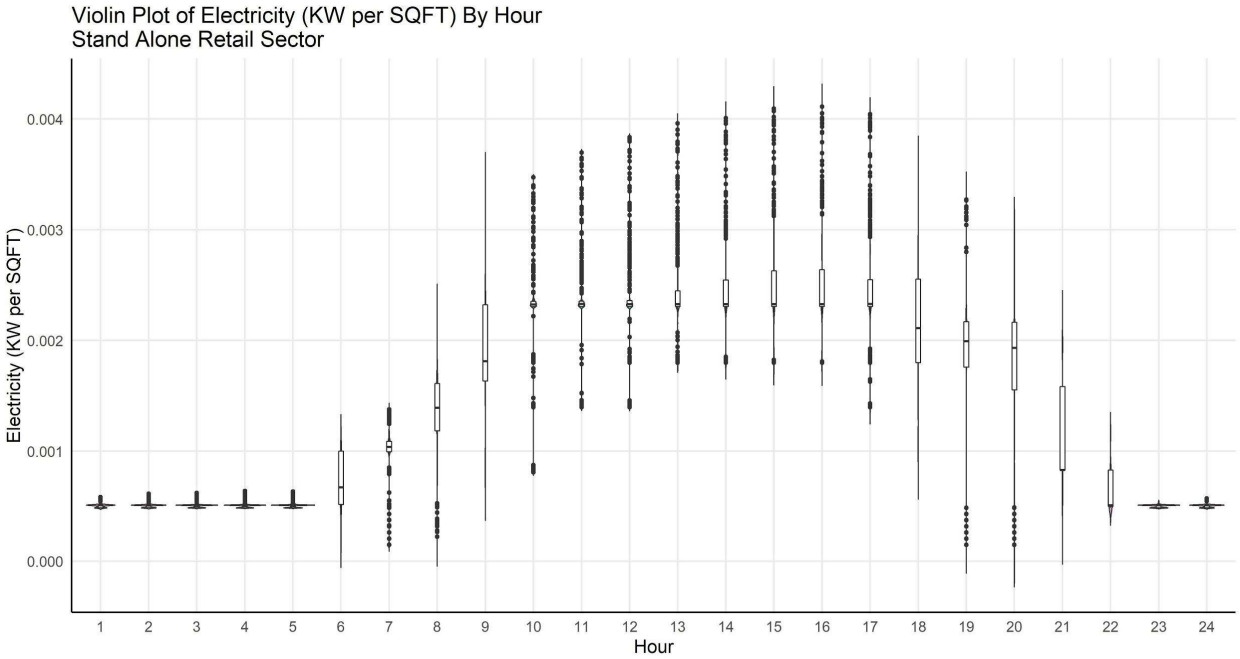
Violin Plot - Electricity Consumption by Hour and Sector

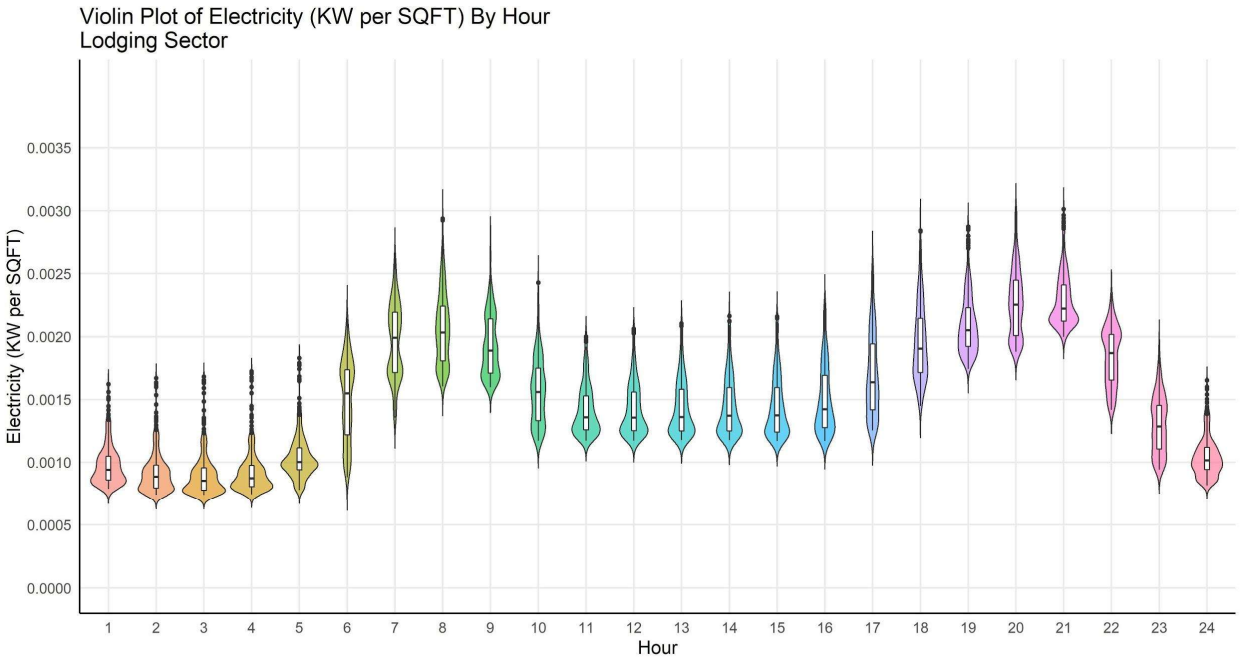
Figure 30: Consumption, Violin Plot of Electricity Usage by Sector by hour

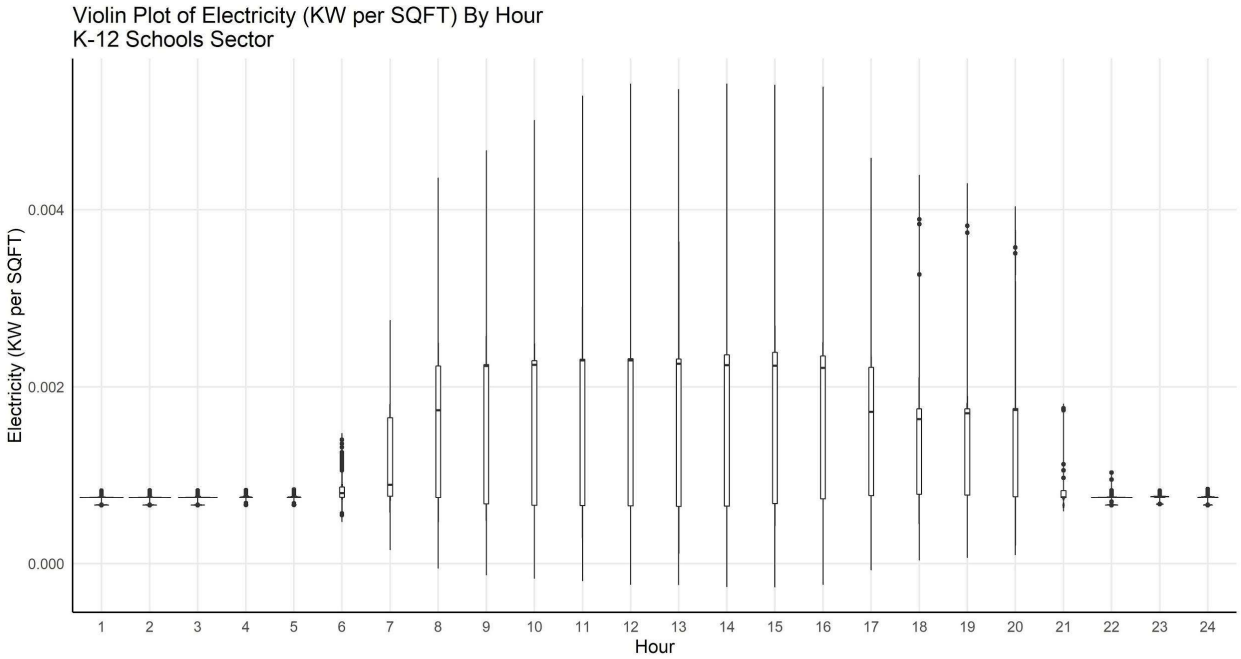



Violin Plot of Electricity (KW per SQFT) By Hour
Office Sector
Electricity (KW per SQFT)
0.006
0.004
0.002
0.000
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
Hour


Violin Plot of Electricity (KW per SQFT) By Hour
Health Care Sector
Electricity (KW per SQFT)
0.0075
0.0060
0.0045
0.0030
0.0015
0.0000
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
Hour


Violin Plot of Electricity (KW per SQFT) By Hour
Stand Alone Retail Sector
Electricity (KW per SQFT)
0.004
0.003
0.002
0.001
0.000
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
Hour


Violin Plot of Electricity (KW per SQFT) By Hour
Lodging Sector
Electricity (KW per SQFT)
0.0035
0.0030
0.0025
0.0020
0.0015
0.0010
0.0005
0.0000
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
Hour


Violin Plot of Electricity (KW per SQFT) By Hour
K-12 Schools Sector
Electricity (KW per SQFT)
0.004
0.002
0.000
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
Hour

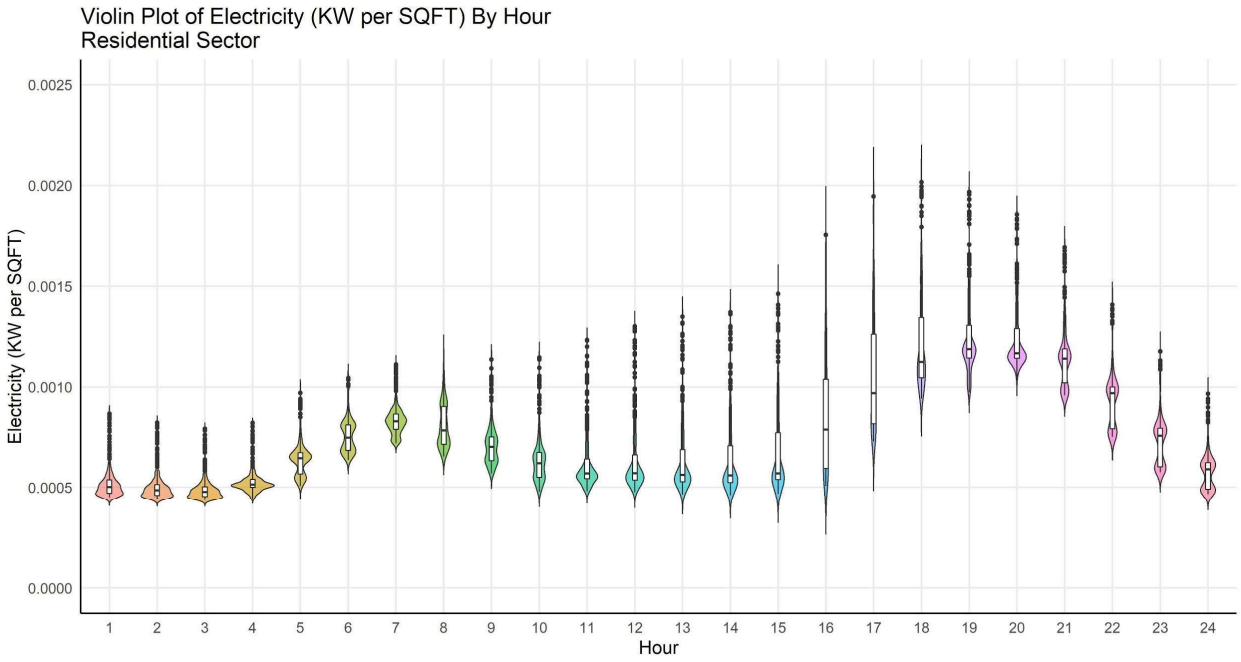

Violin Plot of Electricity (KW per SQFT) By Hour
Residential Sector
Electricity (KW per SQFT)
0.0025
0.0020
0.0015
0.0010
0.0005
0.0000
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
Hour

Figure 31: Consumption, Ridgeline Plot of Electricity Consumption by Sector

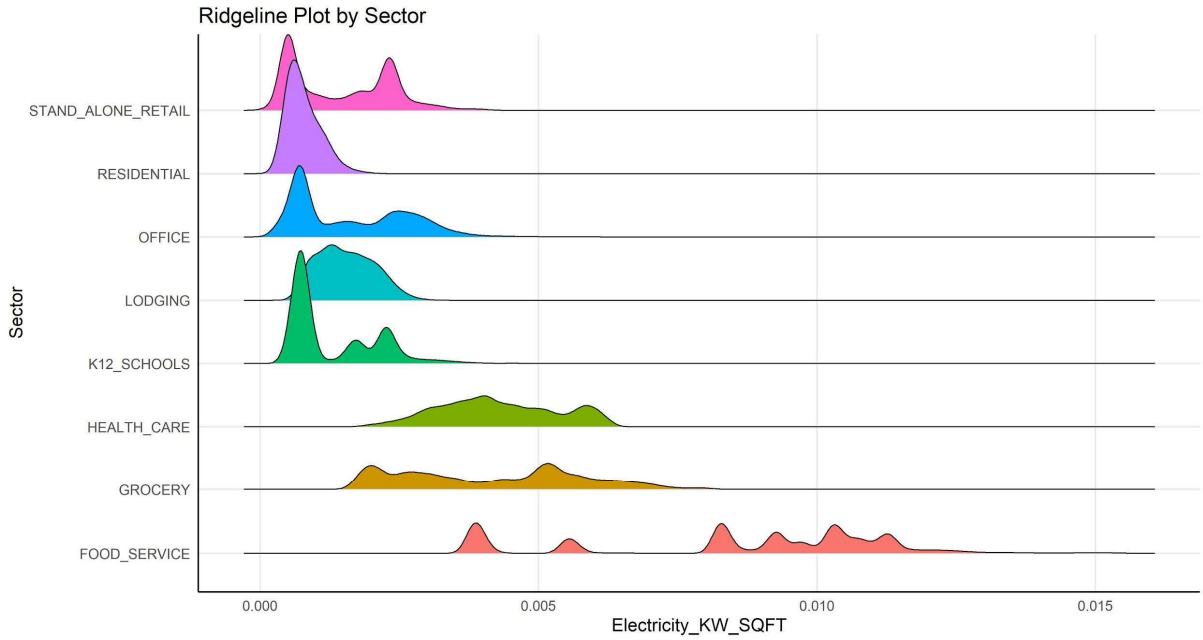


Figure 32: Consumption, Actual versus Predicted Testing Energy Consumption Data

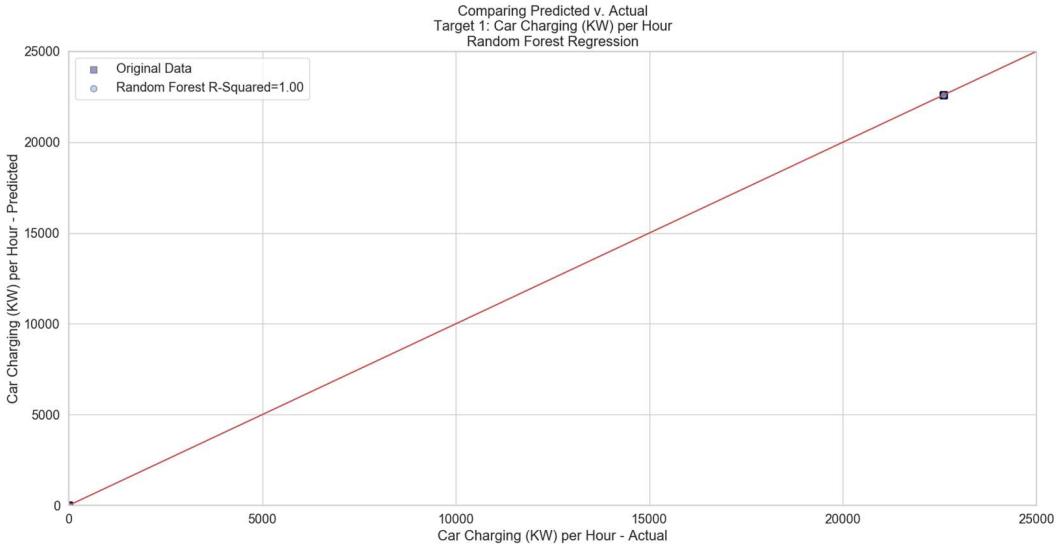


Figure 33: Consumption, Actual versus Predicted Energy Consumption Data - Validation

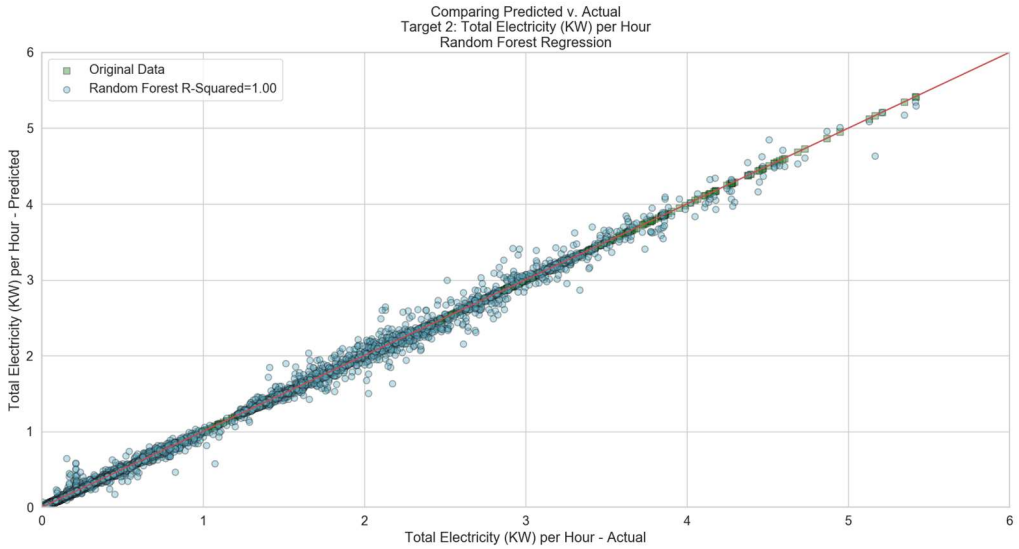


Table 9: Consumption, Model Performance Summary Table (CV, Testing, Training)

Model	R^2	Runtime (seconds)	MAE	RMSE	Explained Variance
Linear Regression (Quantile Transformation, ohe)	61.60%	2.81	0.08	0.13 (+/- 0.00)	0.62 (+/- 0.01)
Lasso Regression (Robust Scaler, ohe, alpha value = 0.1)	55.50%	31.45	925.75	2461.96 (+/- 13.12)	0.56 (+/- 0.03)
Ridge Regression (Quantile Transformation, ohe, alpha value=0.1)	61.60%	1.61	0.08	0.13 (+/- 0.00)	0.62 (+/- 0.01)
Elastic Net Regression (Robust scaler, ohe, alpha value=0.1, l1 ratio value=1)	55.50%	29.23	925.75	2461.96 (+/- 13.12)	0.56 (+/- 0.03)
Decision Tree (Quantile Transformation, non-ohe, criterion='mse')	99.40%	10.46	0.01	0.02 (+/- 0.01)	0.99 (+/- 0.01)
Random Forest (no transformation, non-ohe, n estimators=100, criterion='friedman_mse', min sample split=2)	99.70%	691.77	0.01	0.04 (+/- 0.01)	1.00 (+/- 0.00)
Neural Network (Standard Scaler, ohe, solver='adam')	96.40%	1010.67	0.08	0.18 (+/- 0.11)	0.96 (+/- 0.05)
Gradient Boosting (no transformation, ohe/dummy variables, loss function='ls', n estimators=200)	99.20%	565.10	0.03	0.06 (+/- 0.01)	0.99 (+/- 0.00)
Ada Boosting (no transformation, ohe/dummy variables, loss function='square', n estimators=50)	95.40%	271.49	21.10	166.34 (+/- 997.30)	0.95 (+/- 0.10)
XGBoost Regressor (Power Transformer, non-ohe)	98.90%	699.43	0.04	0.11 (+/- 0.01)	0.99 (+/- 0.01)

Model	R^2	MAE	MSE	RMSE	Explained Variance
Linear Regression (Quantile Transformation, ohe)	61.40%	0.08	0.02	0.12	61.40%
Lasso Regression (Robust Scaler, ohe, alpha value = 0.1)	55.03%	917.03	5916402.39	2432.37	55.03%
Ridge Regression (Quantile Transformation, ohe, alpha value=0.1)	61.34%	0.08	0.02	0.12	61.34%
Elastic Net Regression (Robust scaler, ohe, alpha value=0.1, l1 ratio value=1)	55.80%	933.53	6216348.96	2493.26	55.80%
Decision Tree (Quantile Transformation, non-ohe, criterion='mse')	99.78%	0.00	0.00	0.01	99.78%
Random Forest (no transformation, non-ohe, n estimators=100, criterion='friedman_mse', min sample split=2)	99.90%	0.01	0.00	0.03	99.90%
Neural Network (Standard Scaler, ohe, solver='adam')	97.32%	0.07	0.03	0.16	97.41%
Gradient Boosting (no transformation, ohe/dummy variables, loss function='ls', n	99.33%	0.03	0.00	0.06	99.33%
Ada Boosting (no transformation, ohe/dummy variables, loss function='square',	97.25%	0.06	0.01	0.12	97.26%
XGBoost Regressor (Power Transformer, non-ohe)	99.02%	0.04	0.01	0.11	99.02%

Model	R^2	MSE	MAE	RMSE	Explained Variance
Linear Regression (Quantile Transformation, ohe)	61.95%	0.02	0.07	0.12	61.95%
Lasso Regression (Robust Scaler, ohe, alpha value = 0.1)	55.93%	6083087.57	927.98	2466.39	55.93%
Ridge Regression (Quantile Transformation, ohe, alpha value=0.1)	61.96%	0.02	0.08	0.12	61.96%
Elastic Net Regression (Robust scaler, ohe, alpha value=0.1, l1 ratio value=1)	55.70%	6008112.82	917.52	2451.15	55.74%
Decision Tree (Quantile Transformation, non-ohe, criterion='mse')	100.00%	0.00	0.00	0.00	100.00%
Random Forest (no transformation, non-ohe, n estimators=100, criterion='friedman_mse', min sample split=2)	100.00%	0.00	0.00	0.01	100.00%
Neural Network (Standard Scaler, ohe, solver='adam')	97.65%	0.02	0.07	0.15	97.74%
Gradient Boosting (no transformation, ohe/dummy variables, loss function='ls', n estimators=200)	99.35%	0.00	0.03	0.06	99.35%
Ada Boosting (no transformation, ohe/dummy variables, loss function='square', n estimators=50)	97.18%	0.01	0.06	0.12	97.20%
XGBoost Regressor (Power Transformer, non-ohe)	98.99%	0.01	0.04	0.11	98.99%

Figure 34: Predicted Production - Validation Dataset, March 15

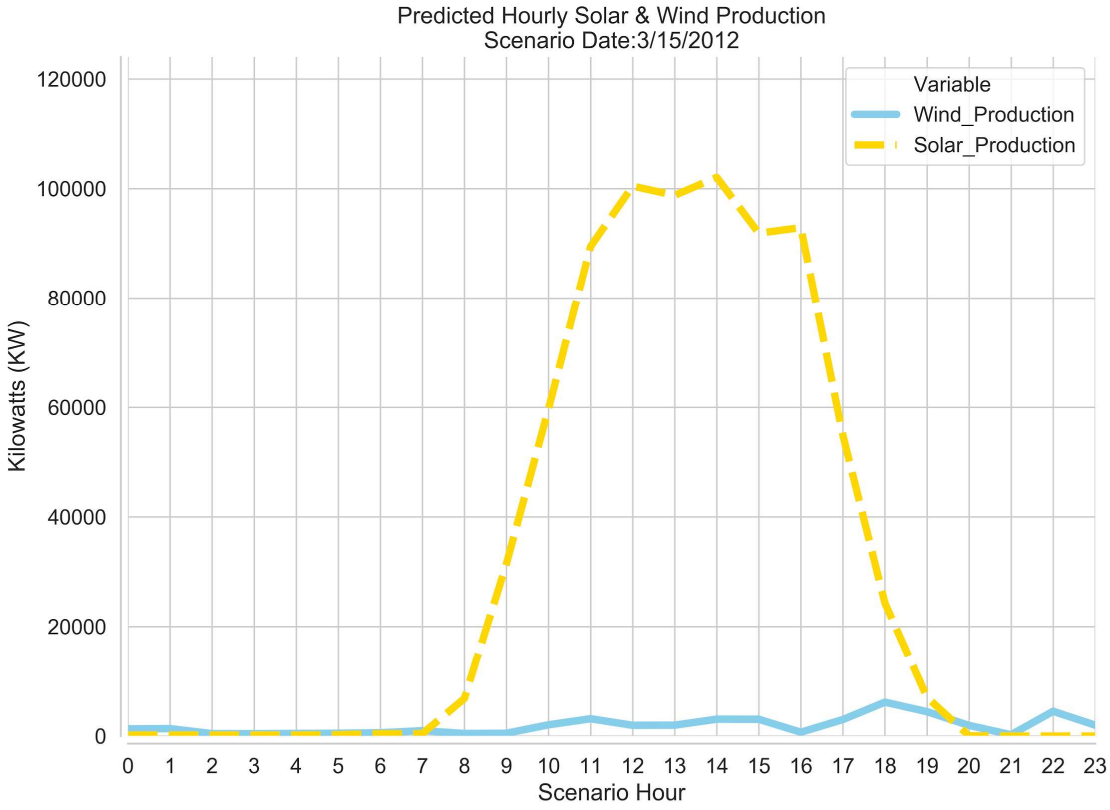


Figure 35: Predicted Production - Validation Dataset, June 26

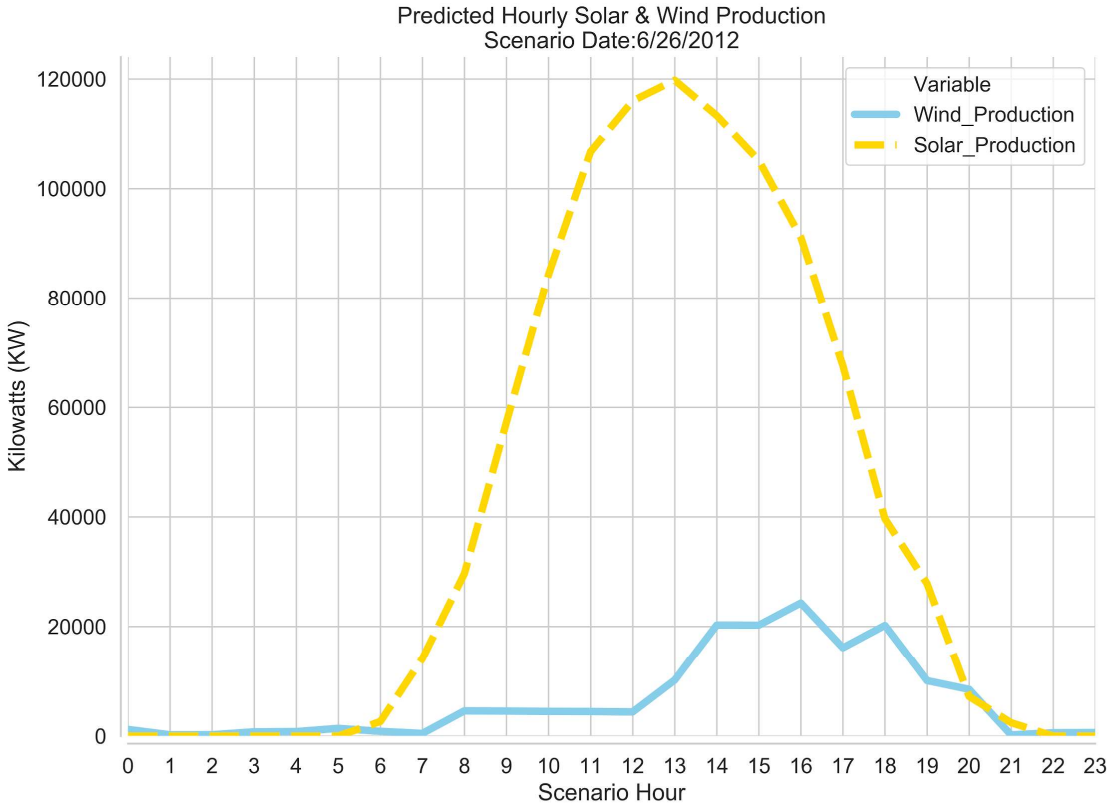


Figure 36: Predicted Production - Validation Dataset, July 3

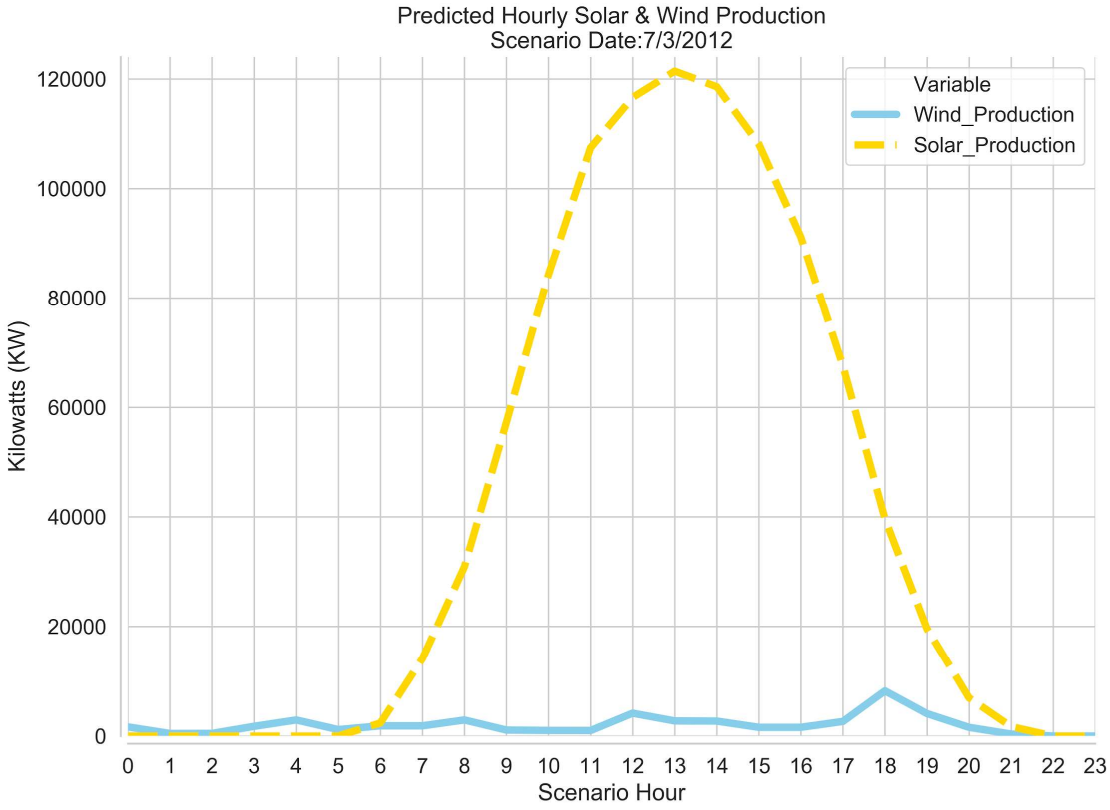


Figure 37: Predicted Production - Validation Dataset, October 13

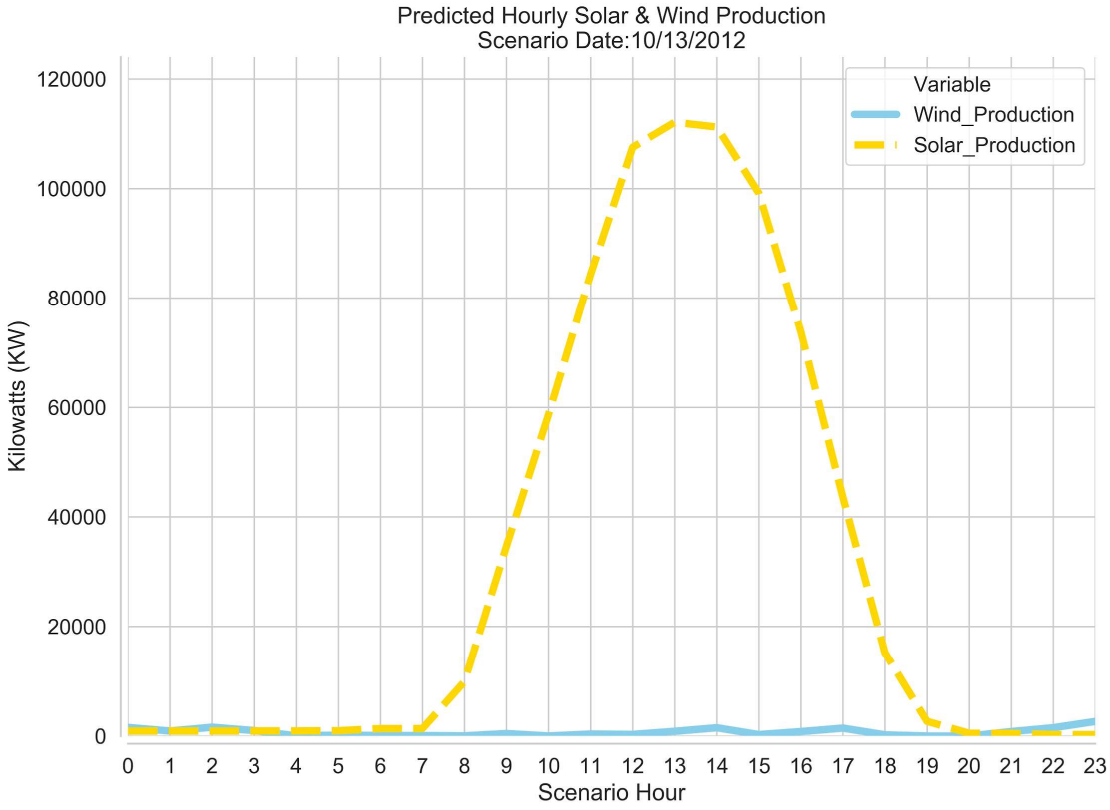


Figure 38: Predicted Production - Validation Dataset, November 19

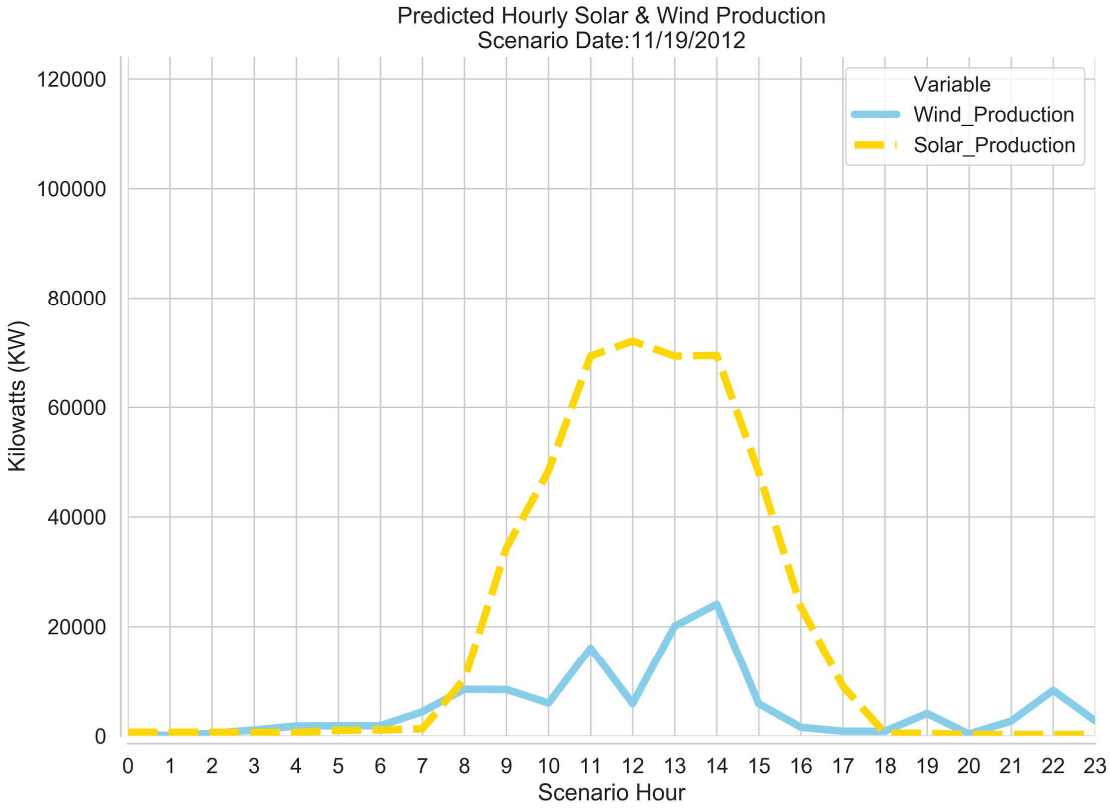


Figure 39: Predicted Production - Validation Dataset, December 25

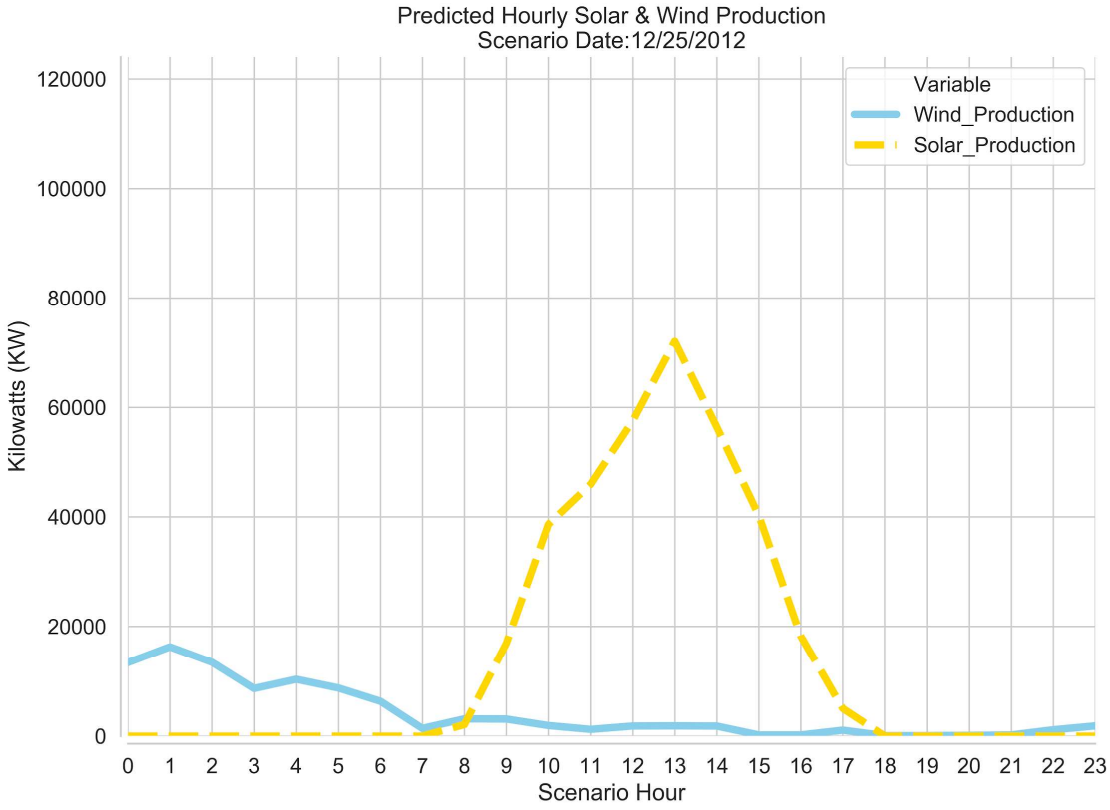


Figure 40: Predicted Production v. Consumption - Validation Dataset, March 15

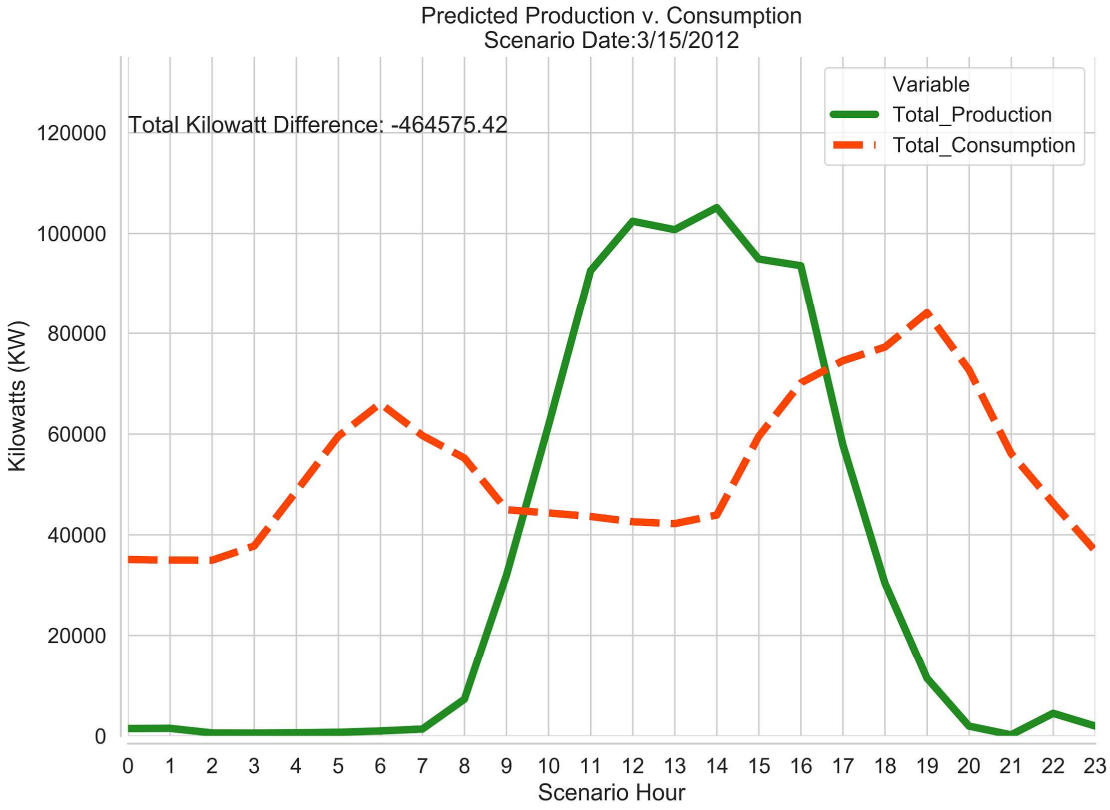


Figure 41: Predicted Production v. Consumption - Validation Dataset, June 26

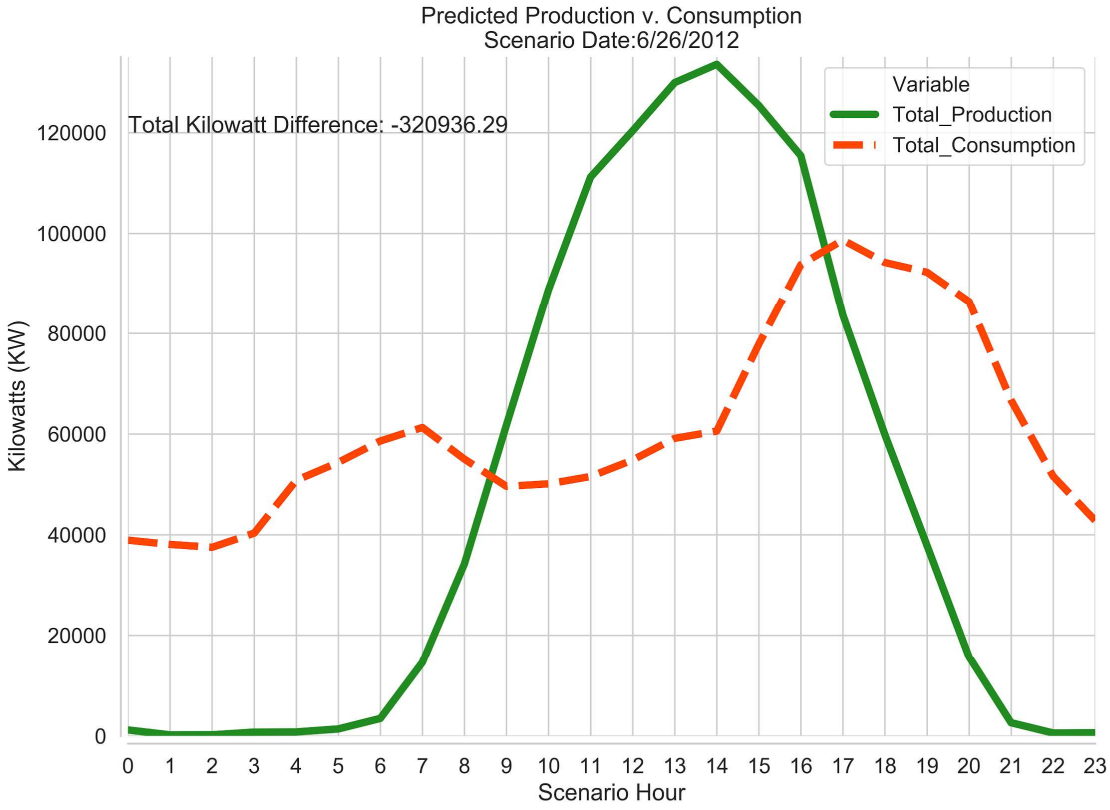


Figure 42: Predicted Production v. Consumption - Validation Dataset, July 3

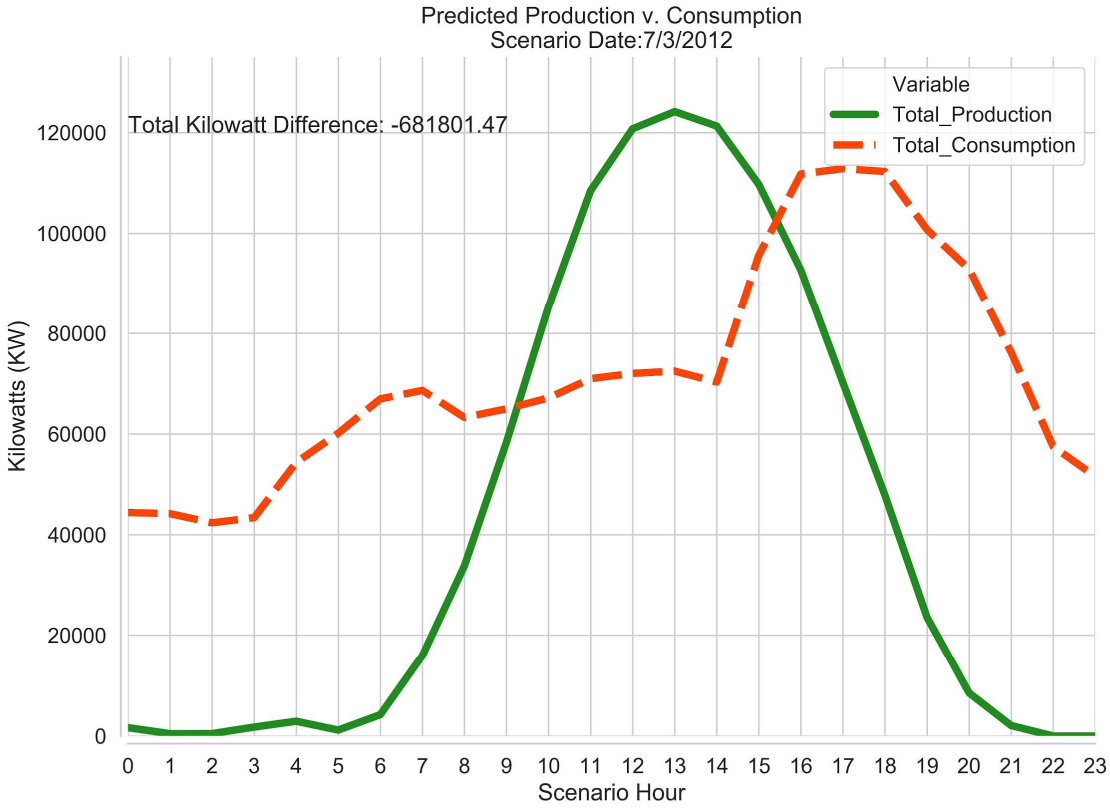


Figure 43: Predicted Production v. Consumption - Validation Dataset, October 13

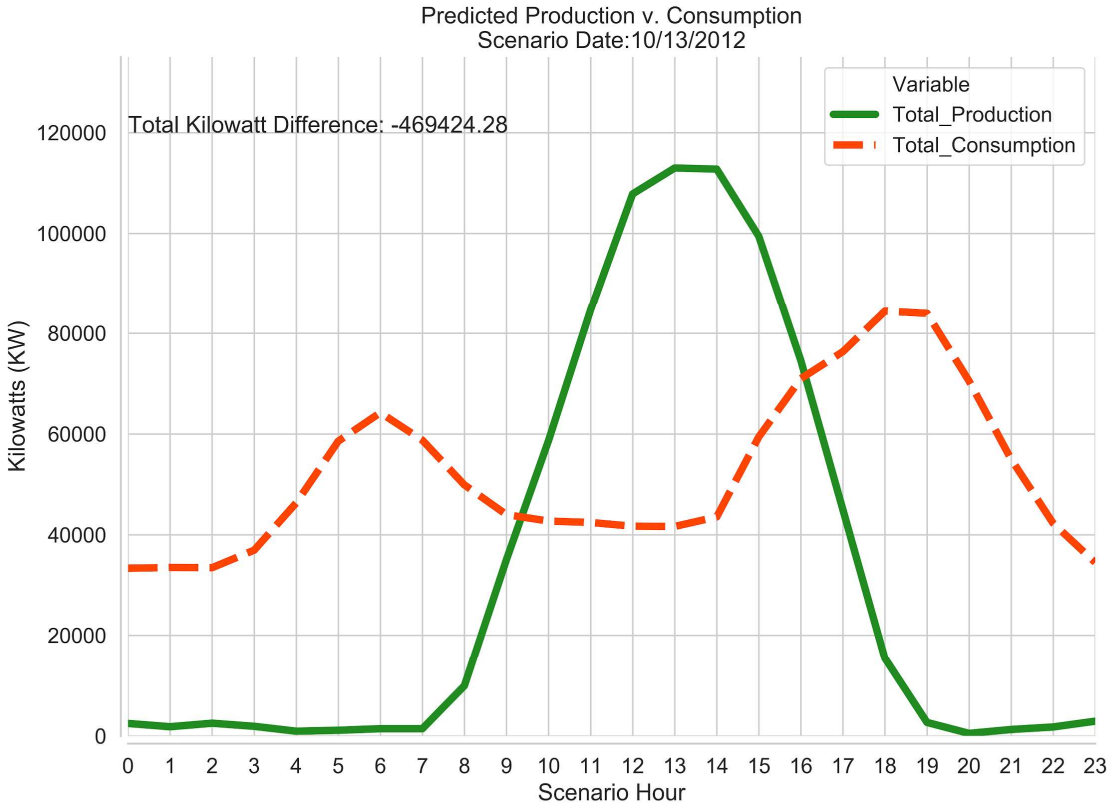


Figure 44: Predicted Production v. Consumption - Validation Dataset, November 19

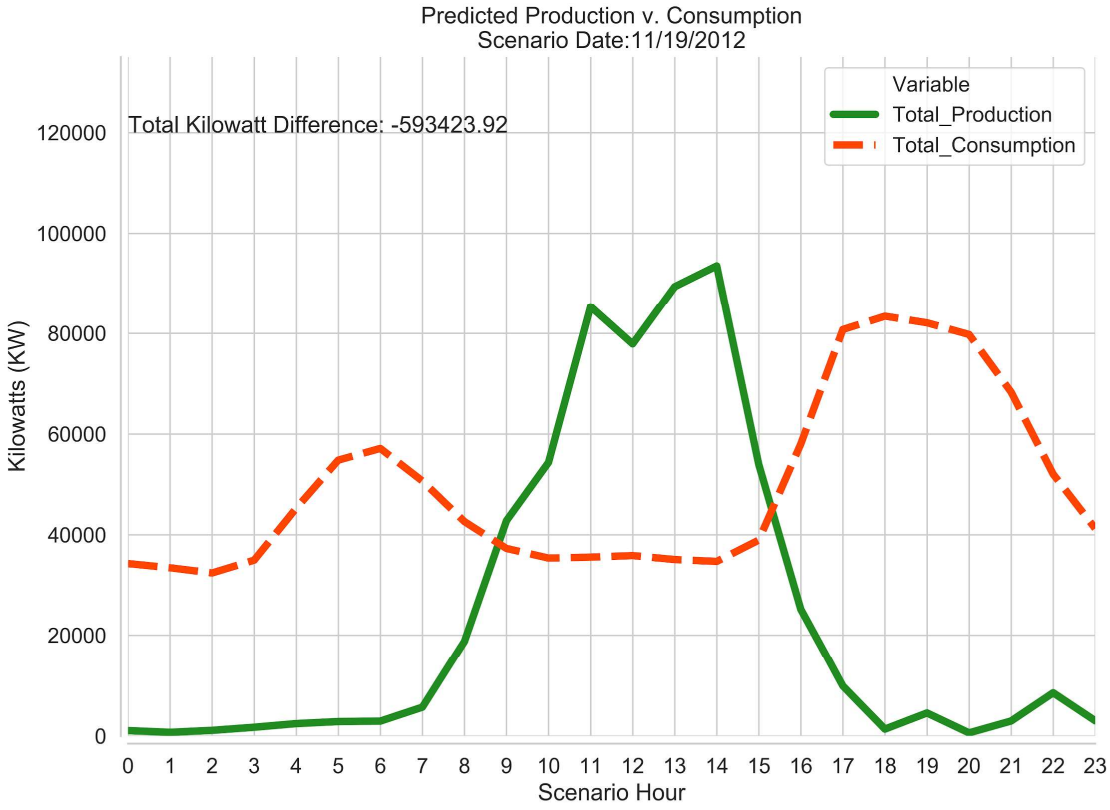


Figure 45: Predicted Production v. Consumption - Validation Dataset, December 25