Data-Driven Solar Irradiance Prediction

Christopher Foo[1]

Advisor: Lipyeow Lim[1]

Co-Advisor: Duane Stevens[2]

[1]*Information & Computer Sciences Department, University of Hawaii at Manoa* [2]*Meteorology Department, University of Hawaii at Manoa*

1. Introduction

The recent surge of renewable energy generation systems on both the residential and commercial level as spurred on by programs such as the Hawaii Clean Energy Initiative, which aims to convert 70% of our energy generation to clean and renewable sources by 2030, has introduced a variety of problems for Hawaii's grid operators. Unlike traditional generator based systems which are fully controlled by the grid's operators, the energy generation from renewable resources is largely beyond the control of the operators as they are dependent on a myriad of factors like the weather. This adds a level of unpredictability to the net energy generation on the grid and further exacerbates the delicate balancing act that takes place between energy consumers and generators in order to ensure the stability of the grid. To alleviate some of this uncertainty, energy operators can (and do) use various dynamic models such as the Weather Research and Forecast (WRF) model for predicting the evolution of local weather. These dynamic models utilize observations of many different types, including surface sensors that we focus on, as initial conditions to project forward in time a prediction of the evolving weather using the laws of physics. The laws of physics are typically represented as nonlinear partial differential equations in space and time, but require further approximations such as discretization of differential equations, data assimilation, and subgrid scale parameterizations of physical processes. The errors inherent in these approximations lead us to ask whether statistical methods alone (such as data mining) without explicit dynamics might be useful in characterizing relationships among sensor data and solar irradiance, both in space and time. Here we have leveraged this sensor data using simple machine learning and data mining techniques (linear regression, cubist trees, a conditional probability classifier, and a naive Bayes classifier) to aid with the prediction of the generative capabilities of one of the most popular renewable energy sources in Hawaii: solar energy. In this paper we will discuss the data sets that we found, our attempt at using those datasets to predict the solar irradiance, and the various patterns that we discovered in through our data analysis.

2. Related Works

Many different data mining techniques have been applied to the problem of solar irradiance forecasting in the past. One approach, as presented in [1], is to apply these techniques to the output of the dynamic models as a post-processing step to improve the resulting forecasts. This differs from our approach as we only use sensor data as the input for our models. Another approach proposed by [2] is to use machine learning techniques to predict the cloud situation in the future using cloud motion vector fields and use the predicted cloud coverage to predict the solar irradiance. This also differs from our approach as we directly predict the surface solar irradiance values instead of inferring them from the predicted value of a different variable.

To directly predict solar irradiance values there are two prevalent techniques. The first techniques is the use of adaptive or autoregressive linear time series models which are linear functions that are updated regularly as new data is obtained with the latest values having a larger weight to better capture the most recent trends. [3] examines the effectiveness of a variation of this model called the ARIMA model while [4] presents a variation of this model called autoregressive with exogenous input or ARX which incorporates the use of the output of the clear sky model into its input. The other technique is the use of an artificial neural network. Several variations of the artificial neural network have been tested. [5] tested a typical feed forward neural network using a genetic algorithm to select the features to be used as input. In contrast, [6] proposes the use of certain statistical feature parameters such as the third derivative of the solar irradiance, normalized discrete difference between the extraterrestrial and surface irradiance, average ambient temperature and surface irradiance, and day of the year as the input for the neural networks. Some network variations that have been used include the wavelet neural network variant ([7] and [8]), the adaptive-network-based fuzzy inference system (ANFIS, [9] and [10]), and multilayer perceptron networks [11]. The techniques used in our experiments are much less complex than both the autoregressive linear time series and artificial neural network models and hence do not fall into either category.

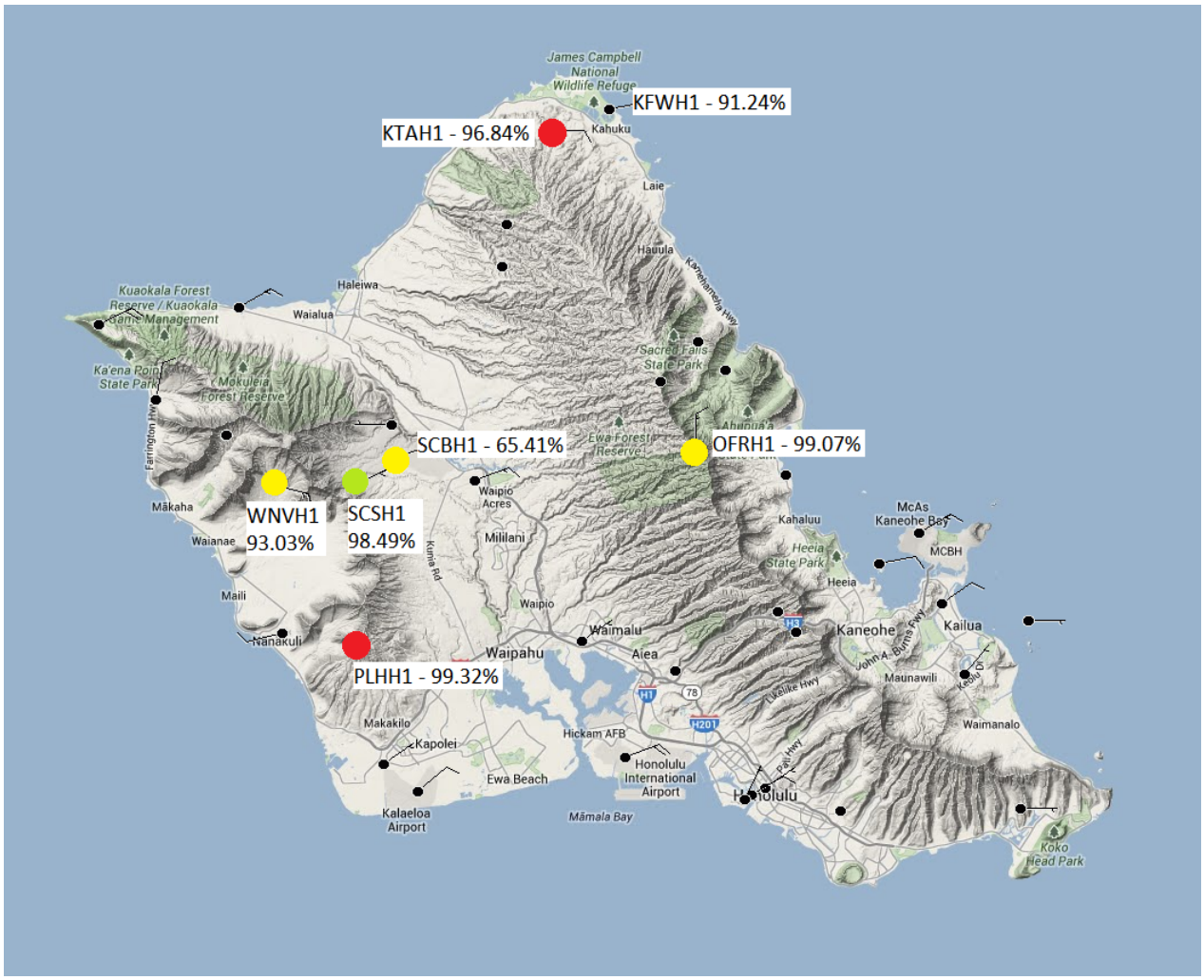
3. Datasets

3.1 Mesowest Dataset

For the majority of our tests, we used 4 years (from January 1, 2010 to December 31, 2013) of the publicly available data for 21 sensor stations on the island of O'ahu that was obtained from MesoWest website.[1] The sensors at these stations provide different types of data such as the temperature, wind direction and speed, and relative humidity and each station is equipped with a different set of sensors. Hence, some readings are missing for some stations while being present in others. Unfortunately, only 7 of these stations provided solar irradiance readings which limited our prediction attempts to these

[1] MesoWest website: http://mesowest.utah.edu/

stations as the actual solar irradiance observed is required to establish a ground truth for our results. These stations and the percent of solar irradiance observations that were present in the data set may be seen in the map below. The SCSH1 station used in the experiments for sections 3 and 4 is indicated by the green circle. The PLHH1 and KTAH1 stations used in the experiments for sections 5, 6, and 7 are indicated by the red circles. All other stations with at least 50% of their solar irradiance observations intact are marked by a yellow circle. Nonetheless, we are still able to incorporate the sensor data from the other stations as additional predictors.



Preprocessing

While each of the 21 stations were active throughout the 4 year period, there are periods where data is missing (perhaps due to maintenance or sensor failure). This poses a problem for our prediction algorithms as we cannot run the algorithm for a particular time if any of the predictor values are missing. We accounted for this by filling in the missing values using a climate mean for that value at that particular hour over our 4 years of

data. Ideally, we would be able to establish a climate by only looking at that particular day and time, but doing so would only allow us to average 4 data points (for the 4 years of data) which is not enough to establish a robust climate. As a result, we calculate the climate mean using a 5 day running window for that hour (the missing day and 2 days before and after the missing day) to give us 20 data points (4 years times 5 days) and a much more robust climate. We forego this step when running our clustering tests though as introducing the filler values would artificially influence the clusters.

Furthermore, each of the stations collect data at different rates with some collecting data as often as every 10 minutes while others only collect data once per hour which would make joining sensor readings across stations difficult. This is further exacerbated by the non-uniformity of the polling time (e.g. every hour 15 minutes after the hour vs. every hour 30 minutes after the hour). To avoid this issue, we normalized the readings for every station to once per hour on the hour. For stations that collect data hourly, this is achieved by considering the reading closest to the top of the hour as being on the top of the hour. Collection rates faster than an hour are handled by binning the readings based on the nearest hour and taking the average of the binned readings. The result is a uniformed data set which makes joining the sensor readings across the stations very simple.

In addition to normalizing the times, we also performed a few adjustments to the raw data. First, we adjusted the wind directions to have them relative to the West and have a range of -180 degrees (exclusive) to 180 degrees (inclusive). In this coordinate system, winds coming from the West would have a wind direction of zero degrees, winds coming from the North would have a direction of 90 degrees, winds from the East would have a direction of 180 degrees, and winds from the South would have a direction of -90 degrees. We also added several calculated values if the prerequisite features were available in the raw data. If the data contains both the temperature and the dew point temperature, we can calculate the *TTD* measure which is the dew point depression, temperature minus dew point temperature. We also used the temperature and the dew point temperature to calculate the relevant vapor pressure: actual vapor pressure *e_actual* is computed from saturation vapor pressure at temperature *TD*, while saturation vapor pressure *e_sat* is the saturation vapor computed at the temperature *T*. Finally, we used the wind direction and speed to separate the wind into its North and East components.

3.2 WRF Solar Irradiance Forecasts

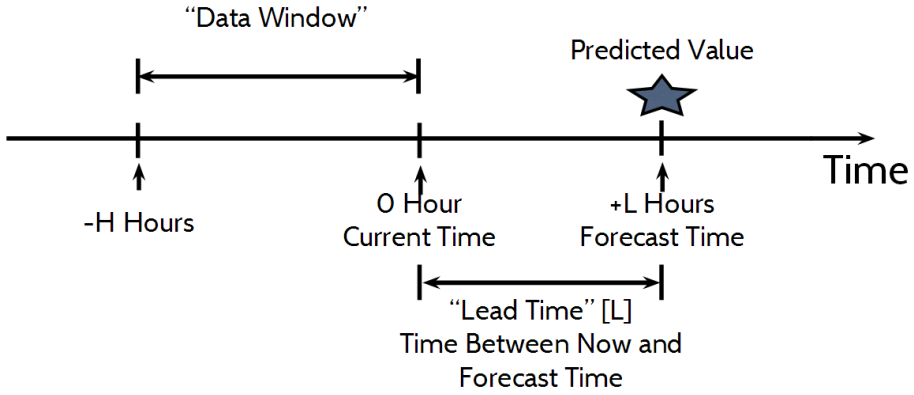
The WRF solar irradiation forecasts used for comparison were obtained from the publicly available WRF forecast archives.[2] These are 3.5 day forecasts run by Prof. Yi-Leng Chen of the Meteorology Department of the University of Hawaii at Manoa's School of Ocean and Earth Science and Technology (SOEST) that start from June 22, 2010 at

[2] WRF Dataset: http://oos.soest.hawaii.edu/erddap/griddap/WRF_OA_Best.html

2:00 am. These WRF forecasts are generated for a grid covering the island of O‘ahu with a resolution of about 1.5 km (i.e. there are forecasts for every 1.5 km by 1.5 km square in the grid). To match these grid-based forecasts with the stations from the MesoWest website, we used the latitude and longitude listed on the MesoWest stations page[3] and used the square that contains the station in question.

4. Prediction of Hourly Solar Irradiance Values

Our first focus was to predict future solar irradiance values using simple machine learning techniques. Here, our goal was to use the data available from *H* hours ago until the current time to predict a solar irradiance value that is *L* hours in the future. Henceforth, we shall refer to the interval from *H* hours ago to the current time as the data window, the time *L* hours in the future as the forecast time, and the interval between the current time and the forecast time as the lead time.



4.1 Feature Selection

Once the lead time and data window has been determined, we can select the top n features based on the pairwise correlation between each possible feature and the solar irradiance at the forecast time using a set of archived data. Here, our pool of features was generated by considering all sensor observations at a particular station at a particular number of hours before the forecast time and we could perform the ranking using either Pearson’s correlation coefficient, Spearman’s rank correlation coefficient, or Kendall’s tau. These correlation coefficients are then sorted in descending order and the top n features are selected.

4.2 Modeling Options

4.2.1 Model Type

[3] MesoWest Station Details: http://mesowest.utah.edu/cgi-bin/droman/stn_state.cgi?state=HI

After the features have been selected, we can then use the archived data to train our models using the selected features. While there are a myriad of different machine learning techniques that can be used to train the models, we decided to set a minimal baseline by choosing two basic machine learning techniques.

1. **Linear Regression:** Linear regression attempts to model the relationship between our selected features and the forecasted solar irradiance as a straight line. It does this by finding a line of best fit for the provided training data and outputting a linear equation in the form $y = a_0x_0 + a_1x_1 + \ldots + a_nx_n + c$ where y is the value that we want to predict (solar irradiance in our case); $x_{0 \ldots n}$ are the observed values for our selected features; and $a_{0 \ldots n}$ and c, etc. are the weights determined by the algorithm to fit the line.
2. **Cubist Tree:** A cubist tree is a decision tree that chooses between several linear models. Here, the algorithm not only uses the provided training data to find the weights of the linear models (created using linear regression), but it also can partition the data based on any significant features that it detects. As a result, the cubist tree can have multiple, more specialized linear regression models based on the criteria it discerns from the training dataset hence introducing one kind of non-linearity into the model.

Since both of these techniques attempt to find linear relationships (although Cubist trees are not strictly linear, it still uses linear regression at the leaves) between the features and the forecasted irradiance, we decided to use the Pearson's correlation coefficient, which measures a purely linear correlation between variables, for all feature selection tasks.

4.2.2 Temporal Partitioning

Due to the cyclic nature of the solar irradiance, we decided to explore whether training and using different models for different temporal intervals would affect the accuracy of our predictions. The most obvious pattern for solar irradiance is the daily pattern as solar irradiance is zero during the night, rises during the morning, peaks around noon, and then falls until sunset. However, there is also a seasonal cycle as the irradiance from the sun tend to be less intense in the Winter compared to the Summer. Furthermore, weather patterns tend to differ throughout the seasons which can affect the cloud coverage and consequently affect the solar irradiance observed at ground level. Hence, we propose two temporal partitioning methods to capture these patterns.

1. **Hourly Partitioning:** To capture the daily pattern, we can create twenty four hourly models, one for each hour of the day, with the hopes that each individual model would be more accurate due to the smaller variations between observations with the same hour. However, creating models for

the night time hours is redundant as the predicted solar irradiance for those hours should always be zero. Hence, we only create models for the hours where we have observed non-zero solar irradiance values in practice.

2. **Monthly-Hourly Partitioning:** We can then further the idea of using multiple models to capture temporal patterns to the seasonal cycle by creating sets of hourly models for each season. This concept can be generalized as having a set of hourly models for every interval of a given number of days within a year. Here, we decided to use 30 day intervals to create a set of models for approximately every month to get 13 sets with at most 24 models in each set.

Since each interval could have different characteristics, the relationships between the features and the solar irradiance could also vary. Therefore, we perform feature selection for every interval to ensure that we are selecting the top features for every partition.

4.3.3 Deseasonalized Data

Another option we explored to handle the daily and seasonal cycles was to deseasonalize the solar irradiance. This was accomplished by taking multiple years of data and finding the average solar irradiance for every hour of every day to obtain the seasonal signal. For example, if we have solar irradiance observations from 2010 to 2013 and we want to find the value of the seasonal signal for January 1 at 12pm (noon), we take the average of the solar irradiance values observed at 12pm on January 1 in the years 2010, 2011, 2012, and 2013. We can then apply this to every single hour of the day and every single day of the year to obtain the entire seasonal signal. To perform the deseasonalization we subtract this seasonal signal from the observed solar irradiance values to obtain the differences from the seasonal signal (i.e. the deseasonalized signal) and substitute it for the observed solar irradiance values. In tests using the deseasonalized data set, we predict the deseasonalized value and then add the seasonal signal to our prediction to reconstruct the solar irradiance.

As the deseasonalization of the data would account for the cycles that we are interested in capturing, using a single model for all hours and days of the year should be sufficient in theory. Nevertheless, creating more specialized models could still be beneficial and we also explore effects of using the aforementioned multi-model schemes in conjunction with the deseasonalized data.

4.3.4 Partitioning Based On Wind Speed

The final option we explored was to split the data into two regimes based on the strength of the observed surface winds at the current time. The intuition here is that the wind

speed affects the speed at which the clouds are moving so splitting the winds into these two regimes would separate the data into the cases where there is either little cloud movement and consequently rather stable solar irradiance or where the clouds are moving quickly and the solar irradiance would sporadically change as the clouds pass by. Here, we chose 3 knots (~1.543 m/s) as the minimum wind speed (inclusive) for the high wind regime as inspired by the Beaufort scale.[4] Once the data has been segregated into the two regimes, we can then apply the same techniques as previously mentioned to each regime.

4.2 Experiments

To test the effects of applying these options, we tested each of the setups using the same testing and training data sets using four fold validation on the four years of sensor data obtained from the MesoWest website as described in the data sets section. In our four fold validation, we ran the tests four times, withholding a single year of data as the testing set and using the other three years as the training set. We then calculated two criteria that we used to evaluate the accuracy of the models:

1. **Absolute Error:** |predicted - actual|
2. **Relative Error**: |predicted - actual| / actual

To obtain an overall evaluation of the model, we take the average of every test from all four folds. The tests were run for the SCSH1 (Schofield Barracks South Range) station due to its location in the center of O'ahu. All tests were implemented using R with linear regression being handled with the built-in stats package and cubist trees being handled by the Cubist package.[5] All models were trained using the default settings and with a lead time of one hour unless otherwise stated.

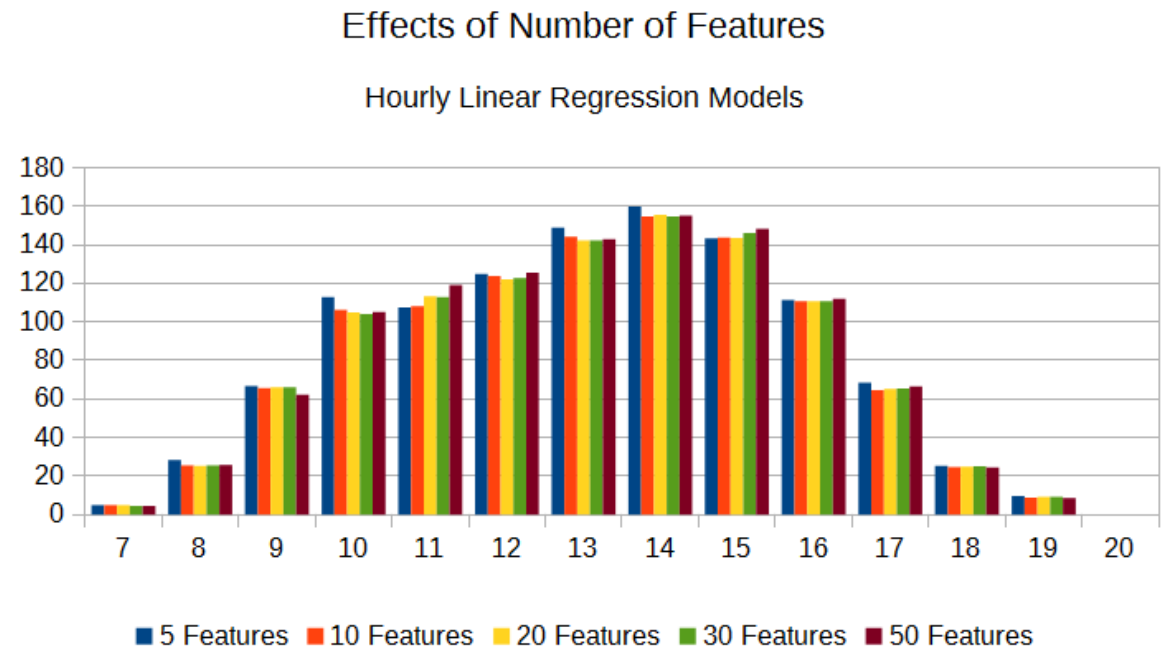
4.2.1 Number of Features

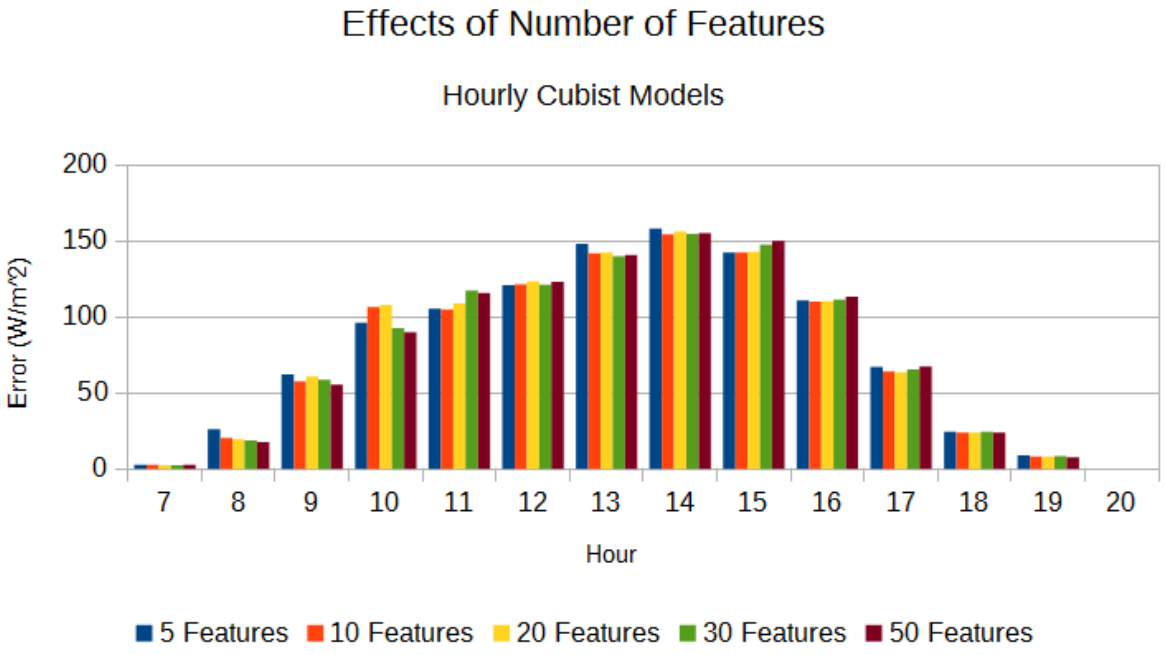
Goal: Since our feature selection method takes the number of features as a parameter, we wanted to determine how much of an effect the number of features selected had on the resulting predictions. Having too few features would cause inaccuracies as the models would be missing some of the significant relationships between the observed sensor data and the solar irradiance while having too many features can also be detrimental to the accuracy of the models because having an excess of insignificant features would introduce noise that could confuse the model. As a result, we must find a number of features that does not fall into either extreme and produces the best results.

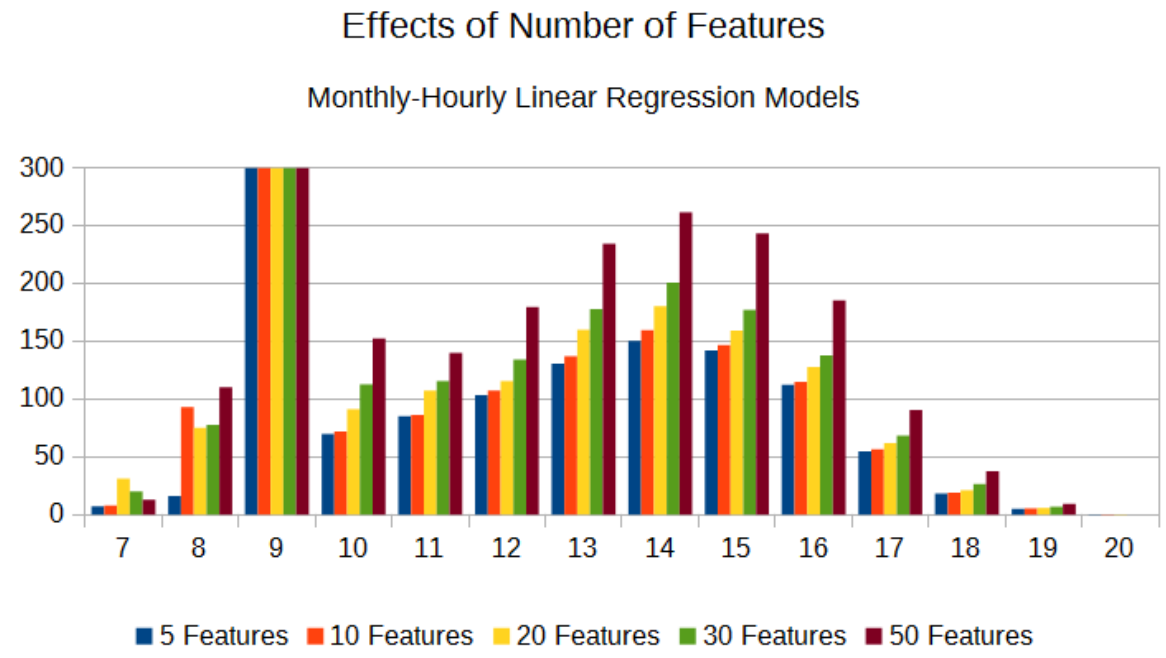
[4] Beaufort scale source: http://www.spc.noaa.gov/faq/tornado/beaufort.html
[5] The Cubist package may be found here: http://cran.r-project.org/web/packages/Cubist/index.html

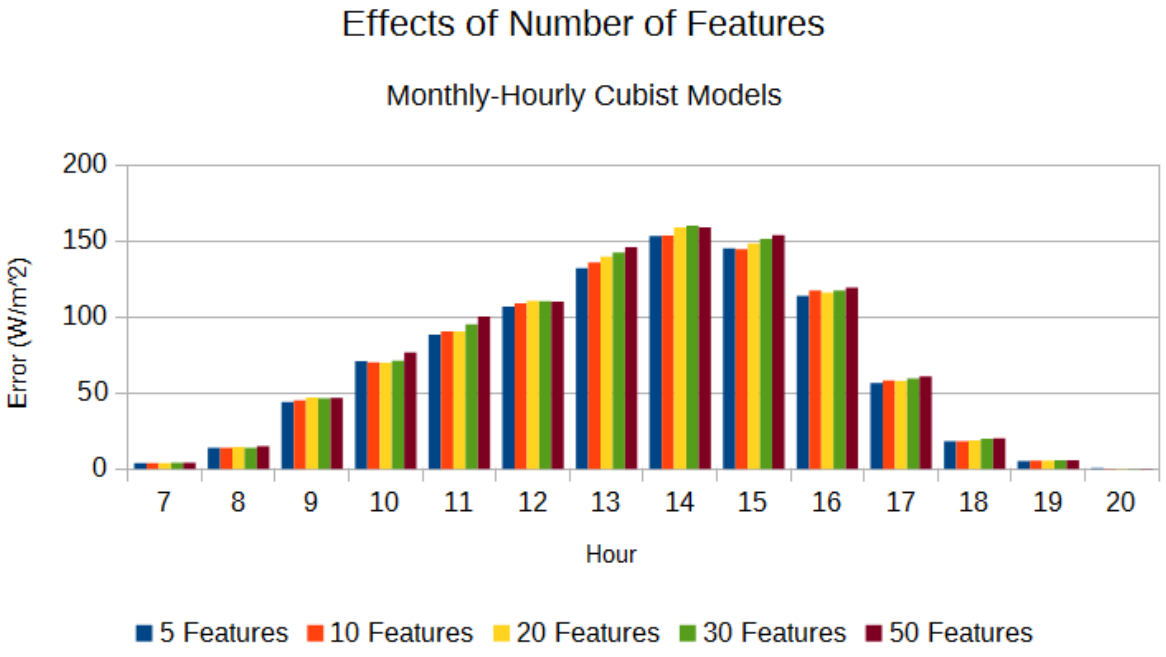
Method: In this experiment, we ran our four fold validation tests on the normal (i.e. non-deseasonalized) data with one hour of lead time for both of the temporal partitionings that we proposed and both of the model types. We ran these tests five times for each partitioning / model type pairing with differing numbers of selected features each time. The number of selected features ranged from 5 to 50 features. We then compared the overall accuracy of our predictions within each partitioning / model type pairing to determine the number of features that lead to the most accurate predictions and to observe any trends.

Results: Changing the number of features seemed to have a relatively small effect in most cases as the errors were comparable between all numbers of features. Even so, there were two discernable trends depending on the partitioning method used. For hourly partitions (a single model for each hour for a maximum of 24 partitions), increasing the number of features from 5 to 10 or 20 improved the accuracy by a small margin. The monthly-hourly partitioning (a single model for every hour in a 30 day interval for a maximum of 24 * 13 partitions) experience the opposite as increasing the number of features tended to decrease the accuracy. This implies that the number of significant features is rather small since increasing the number of features quickly confused the models when the models were more specialized. However, it seems that the best features may vary based on the season because the hourly models, where the models are used for the entire year, needed more features to fully capture all of the significant features.


Effects of Number of Features
Hourly Linear Regression Models
180
160
140
120
100
80
60
40
20
0
7
8
9
10
11
12
13
14
15
16
17
18
19
20
5 Features
10 Features
20 Features
30 Features
50 Features


Effects of Number of Features
Hourly Cubist Models
Error (W/m^2)
200
150
100
50
0
7
8
9
10
11
12
13
14
15
16
17
18
19
20
Hour
5 Features
10 Features
20 Features
30 Features
50 Features





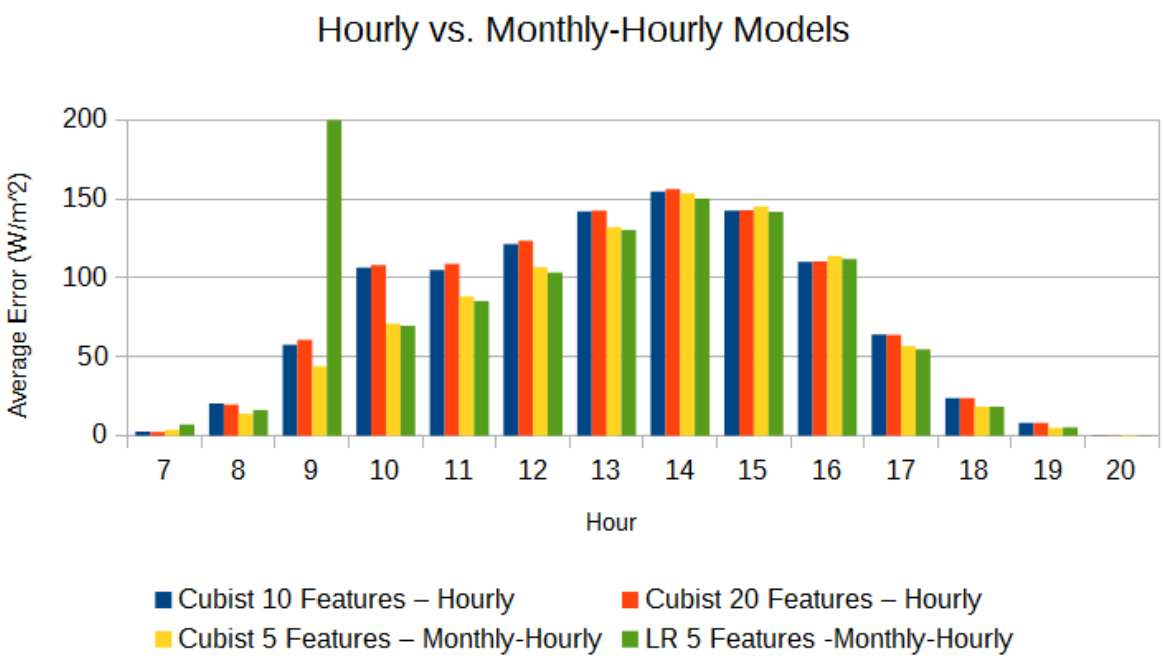
4.2.2 Hourly vs. Monthly-Hourly Partitioning

Goal: In this experiment, the aim was to determine which of the two temporal partitioning methods that were proposed performs better in practice. While the monthly-hourly partitioning would capture both the daily and seasonal trends, it is possible that the models could become overly specialized and become inaccurate when used to

predict values outside of the training set. This is especially true due to our relatively small dataset as each model would have a maximum of 90 data points (30 data points / year * 3 years) in its training set. Hence, we ran these tests to determine if such over specialization would become an issue.

Method: This experiment was run using the exact same methodology as our number of features experiment. However, this time we compared results between partitioning / model type pairings by selecting the models that had the lowest overall error from each partitioning method and compared the results of the selected models.

Results: A chart comparing the best two hourly models and the best two monthly-hourly models may be seen below. Overall, the monthly-hourly models tended to have a lower error and hence were more accurate. The hourly models were more accurate in only two cases. At hour 9, the linear regression monthly-hourly model had a very large outlier. Similarly, the Cubist monthly-hourly model performed the worst at hour 15. Nonetheless, the other model type performed comparably or better at those hours which implies that it was more of an issue with the type of model created rather than the partitioning. It is also possible to use different model type for each hour or month so these outliers can be eliminated by simply using the other model type for that particular interval. As a result, it is safe to assume that it would be better to use monthly-hourly models instead of hourly models as the prediction results would be either comparable or better.

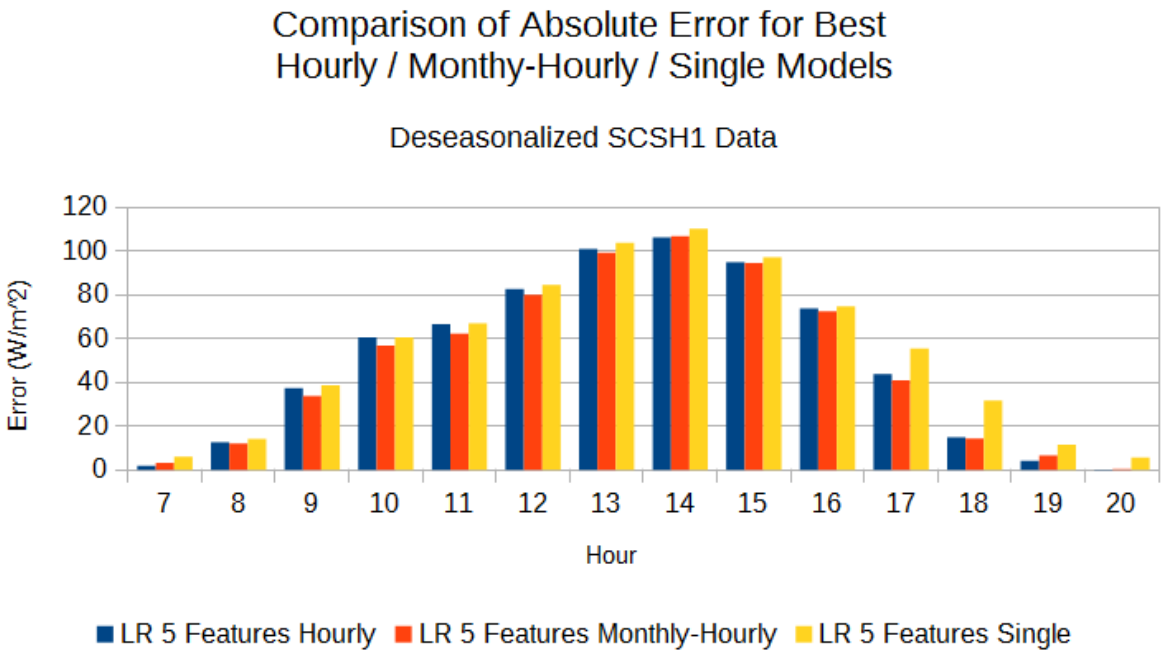


4.3.3 Single vs. Hourly vs. Monthly-Hourly for Deseasonalized

Goal: Deseasonalizing the data removes the daily and seasonal trends from the solar irradiance signal that we wish to predict and stores them in the seasonal signal that is added to our predictions to reconstruct an actual solar irradiance value. Since the seasonal trend describes the daily and seasonal solar irradiance trends, it is not necessary to perform the temporal partitioning to ensure that these patterns are accounted for and a single model for all hours and days of the year should be sufficient. Nevertheless, it is still possible that having more specialized models could be beneficial. Hence, we will test our models using both of our proposed temporal partitioning methods to see if the partitioning is an improvement over having just a single model, regardless of the time or day, for all predictions.

Method: For this experiment, we ran our four fold validation tests on the deseasonalized data set with the three partitioning methods. For each partitioning method, we also tested both linear regression and cubist trees and varied the number of features from 5 to 50 to find the optimal number of features. We then chose the model with the lowest overall error from each partitioning method and compared the results of those best models.

Results: While the differences between partitioning methods are much less pronounced than with the non-deseasonalized data, it is clear that using the monthly-hourly partitioning does improve the accuracy slightly compared to both the hourly and single partitioning methods. This suggests that the deseasonalization removes some of the seasonal trend, but not all of it as the partitioning captures the remnants of that trend to slightly improve the accuracy.

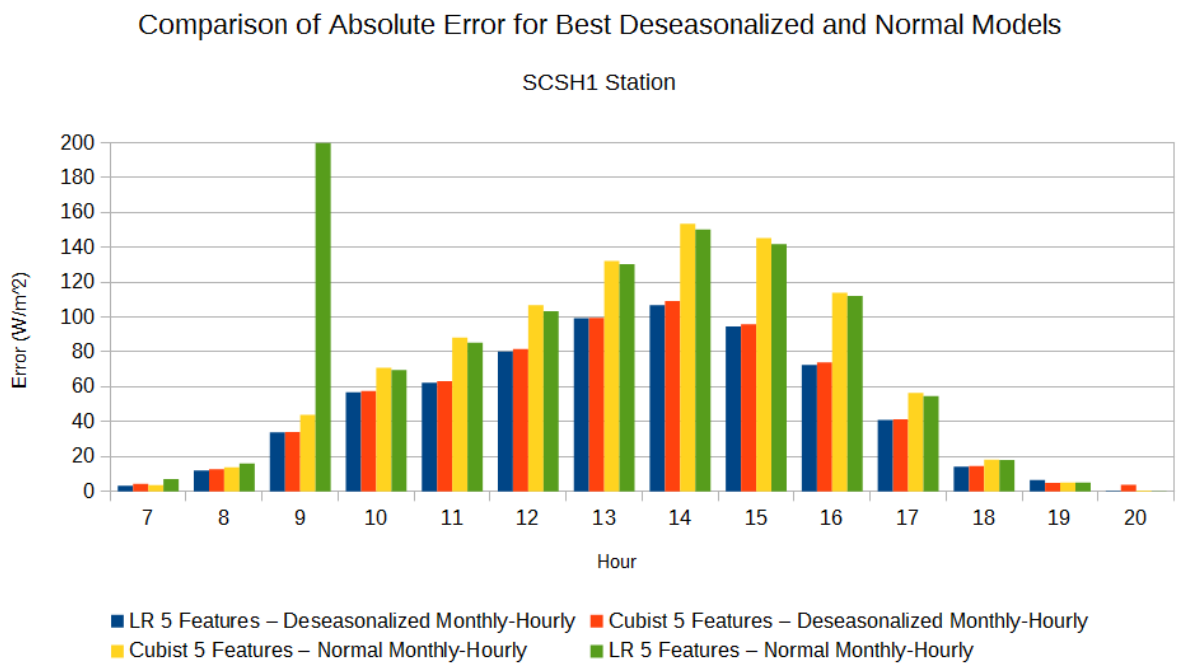


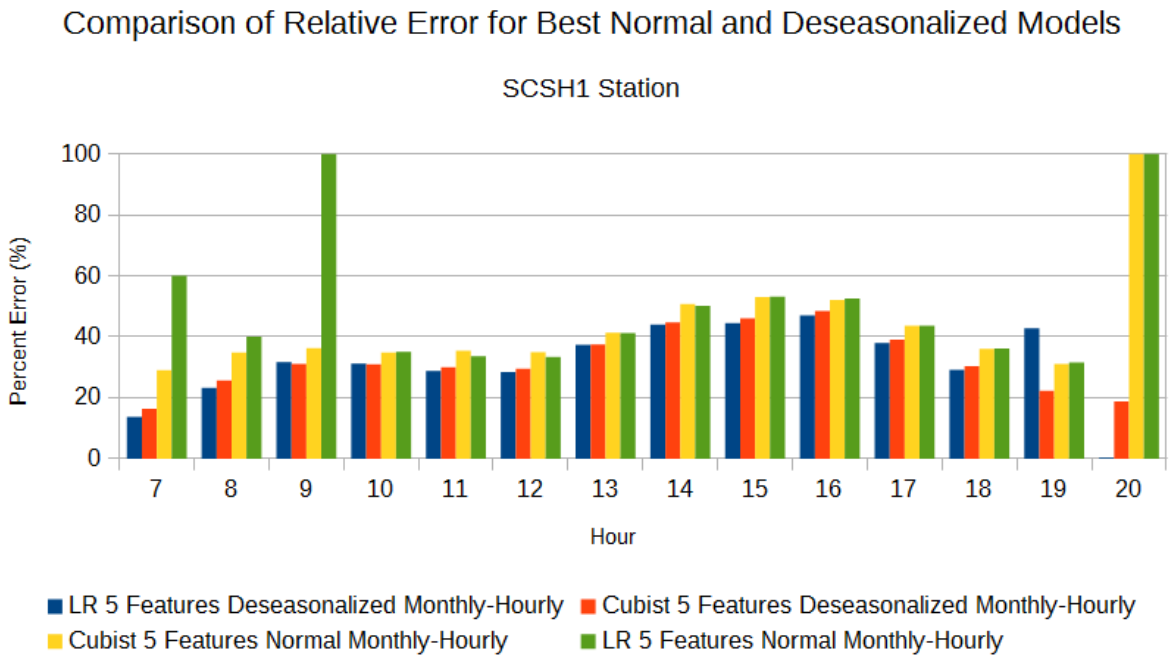
4.3.4 Normal vs. Deseasonalized

Goal: Here we aimed to determine whether deseasonalizing the data improves the accuracy of our predictions or not.

Method: The data for this comparison was obtained using the same method as the previous tests for both the deseasonalized and non-deseasonalized models. We then chose the models with the lowest overall error from the deseasonalized models and the non-deseasonalized models and compared the selected models models.

Results: The charts below compare the prediction accuracy of the deseasonalized models and the non-deseasonalized models. It appears that the deseasonalizing the solar irradiance results in a rather noticeable improvement to the prediction accuracy for nearly all hours. This difference is especially pronounced during the high solar irradiance times while both setups are comparable during the low solar irradiance hours. Since deseasonalizing the data results in much more accurate predictions, we shall be using it as our baseline moving forward. Even so, our predictions are still differ by about 30% from the actual irradiance on average as seen in the relative error chart.



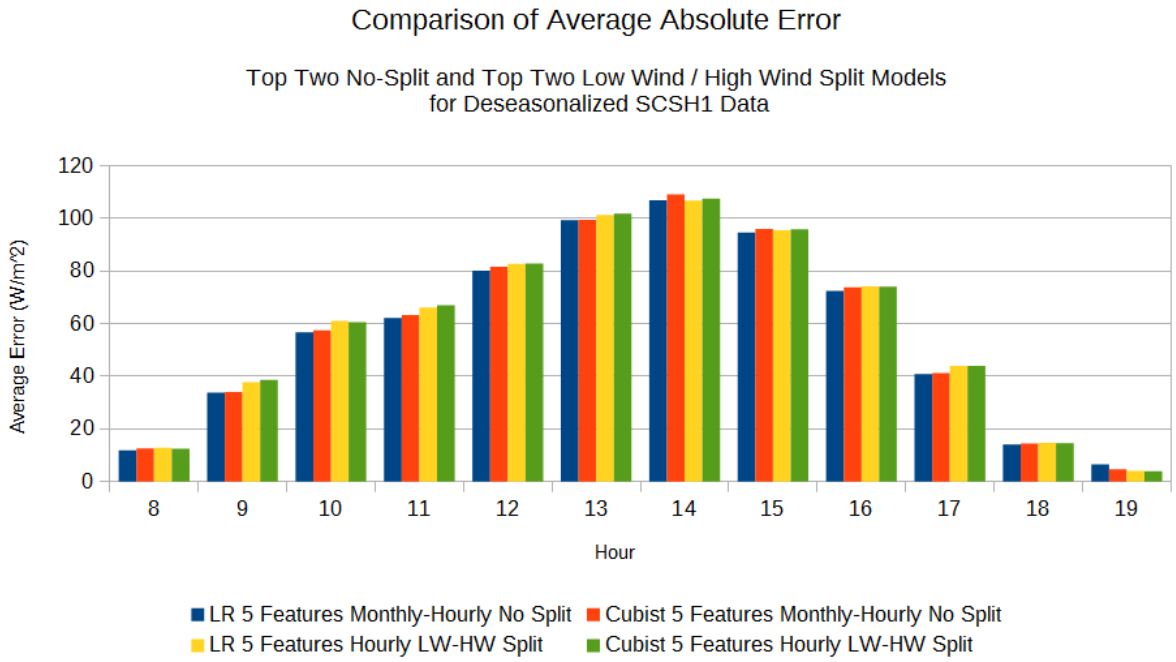


4.3.5 Wind Partitions vs. No Partitions

Goal: In these experiments, we tested our final partitioning option which creates partitions based on the wind speed. This is based on our intuition that the wind speed has a strong effect on how clouds behave and hence should be a good predictor. If we see an improvement in the accuracy here, then our intuition is confirmed. Otherwise, our models were unable to capture the relationship between the wind speed and the solar irradiance.

Method: To perform this experiment we first partitioned deseasonalized data into the low wind and high wind partitions based on the wind speed. We can then run four fold validation on both of these partitions. Once again, we tested both model types and all three partitioning types, but only used 5 features since we observed that was the optimal number of features from the previous tests. To obtain an overall error for the wind partitioned models, we can average the errors obtained from both the low wind and high wind partition. These overall errors are then used to select the best models for our comparison.

Results: The results compared to our baseline (No-Split) may be seen below. Overall, the accuracy of these setups are comparable with the baseline being slightly better in a few cases. This implies that our models were unable to capture the relationship between the solar irradiance and the wind speed.

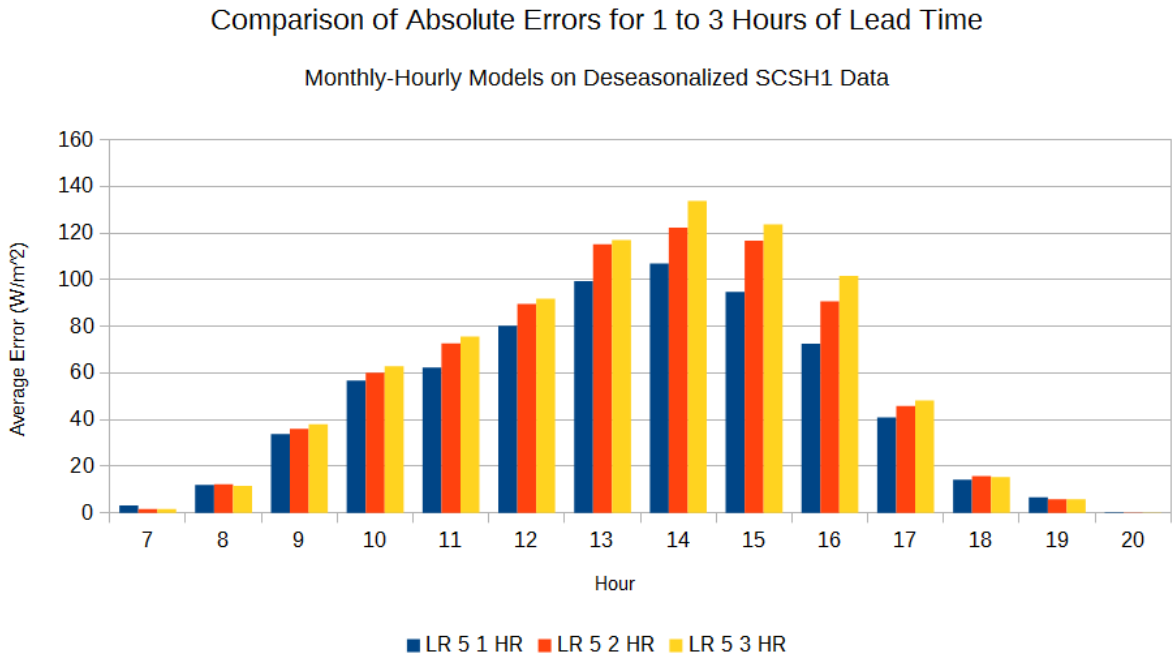


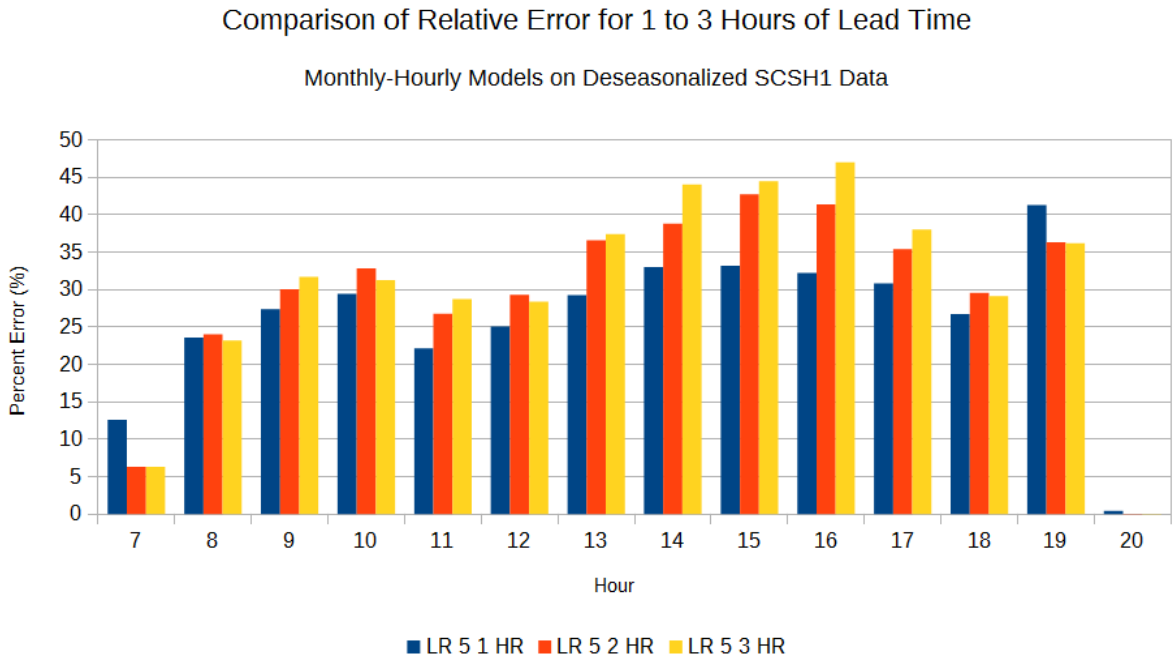
4.3.6 Lead Time (1 hr vs. 2 hrs vs. 3 hrs)

Goal: All of our previous tests used a lead time of one hour, but an hour may not be enough of a lead time for some applications. As a result, we tested our predictions with lead times of two hours and three hours to see how the accuracy changed. In general, we expect the accuracy of the predictions to decrease as the effects of persistence becomes weaker due to the increased temporal distance of our latest observed sensor data and the forecast time.

Method: To perform this experiment, we created monthly-hourly linear regression or cubist tree models using 5 features and deseasonalized data. As seen from the previous experiments, this set up should produce the best results in most cases. To simulate lead times of more than an hour, we omitted data points that would fall within the lead time period. For example, if we used a lead time of 2 hours, we ignored all feature values that were one hour before the forecast time. Otherwise, the tests were identical to the previous deseasonalized tests.

Results: As expected, increasing the lead time decreased the accuracy of the predictions as persistence becomes less significant as the distance between the current time and the forecast time increases. However, the error does not increase as drastically as expected as the differences between one hour and two hours of lead time is larger than the increase from two hours to three hours.





4.2.7 Correlation Analysis

Goal: To select the features for our models, we used pairwise correlations, calculated using Pearson's correlation coefficient, to rank our features as the correlation coefficient is a measure of the strength of the linear relationship between the feature and the solar

irradiance. This is useful as features with a strong relationship with solar irradiance would also be good predictors for solar irradiance. Here we aimed to analyze the best predictors of deseasonalized solar irradiance by examining the features selected by our models.

Method: Since we used multiple models we aggregated the selected features using the measures of selection rate and average correlation.

1. **Selection Rate:** The percent of models that a particular feature was selected, i.e., how frequent was a feature picked.
    - (number of times the feature was selected / number of models) * 100
2. **Average Correlation:** The average of the correlation coefficients observed when the feature was selected by a model, i.e., how strong was the dependence.
    - $(correlation_0 + correlation_1 + \ldots + correlation_i) / i$
    - Where i is the number of times the feature was selected

The features are named using the following convention. For three part names, the first part is the code for the station that the feature was observed at, the second part is the feature type (SOLR is the solar irradiance, RELH is the relative humidity, and TTD is the difference between the temperature and the dewpoint temperature), and the last part is the number of hours before the time of prediction. Two part names omit the station code as they are at the station we are performing the prediction at (SCSH1 in this case).

Results: Overall, the same features were selected for most scenarios. In all cases, the solar irradiance in the previous two hours were the top selected features and hence are the most strongly related with the solar irradiance at the target time. This is no surprise as this is the effect of persistence which is also supported by the other features as they are all within 3 hours of the target time. Other prevalent features are the TTD and solar irradiance of PLHH1 and the relative humidity of WNVH1. TTD and RELH both indicate the humidity at the surface; when humidity is high, cloudiness tends to be high and irradiance is low, so it follows that humidity should be correlated with irradiance. In addition, PLHH1 and WNVH1 are two of the closest stations to the SCSH1 station that we are examining which implies that the spatial distance also has an effect on correlation between the features and the solar irradiance of our target station.

Single Model - Top 5 Features		
Feature	Selection Rate	Average Correlation
SOLR_1	100%	0.613
SOLR_2	100%	0.380
PLHH1_TTD_1	100%	0.220

D3665_TTD_1	100%	0.219
SOLR_3	100%	0.213

Hourly Models - Top 5 Features		
Feature	Selection Rate	Average Correlation
SOLR_1	92.86%	0.521
SOLR_2	64.29%	0.397
PLHH1_SOLR_1	57.14%	0.390
WNVH1_TTD_1	28.57%	0.440
PHNL_TTD_1	21.43%	0.313
WNVH1_RELH_1	21.43%	-0.456

Hourly Models - Top 10 Features		
Feature	Selection Rate	Average Correlation
SOLR_1	92.86%	0.521
SOLR_2	64.29%	0.397
PLHH1_SOLR_1	57.14%	0.390
WNVH1_TTD_1	50%	0.382
WNVH1_RELH_1	50%	-0.378

Monthly-Hourly Models - Top 5 Features		
Feature	Selection Rate	Average Correlation
SOLR_1	75.16%	0.610
SOLR_2	27.33%	0.523
WNVH1_TTD_1	17.39%	0.516
PLHH1_SOLR_1	16.77%	0.535
WNVH1_RELH_1	13.66%	-0.515

Monthly-Hourly Models - Top 10 Features		
Feature	Selection Rate	Average Correlation
SOLR_1	77.02%	0.603
SOLR_2	33.54%	0.512
WNVH1_TTD_1	27.33%	0.481
PLHH1_SOLR_1	26.71%	0.515
WNVH1_RELH_1	25.47%	-0.500

	Monthly-Hourly Models - Top 5 Features		
	Feature	Selection Rate	Average Correlation
Low Winds	SOLR_1	71.43%	0.611
	SOLR_2	21.74%	0.527
	PLHH1_SOLR_1	15.53%	0.522
High Winds	SOLR_1	47.26%	0.693
	SOLR_2	13.01%	0.614
	WNVH1_RELH_1	8.90%	-0.658

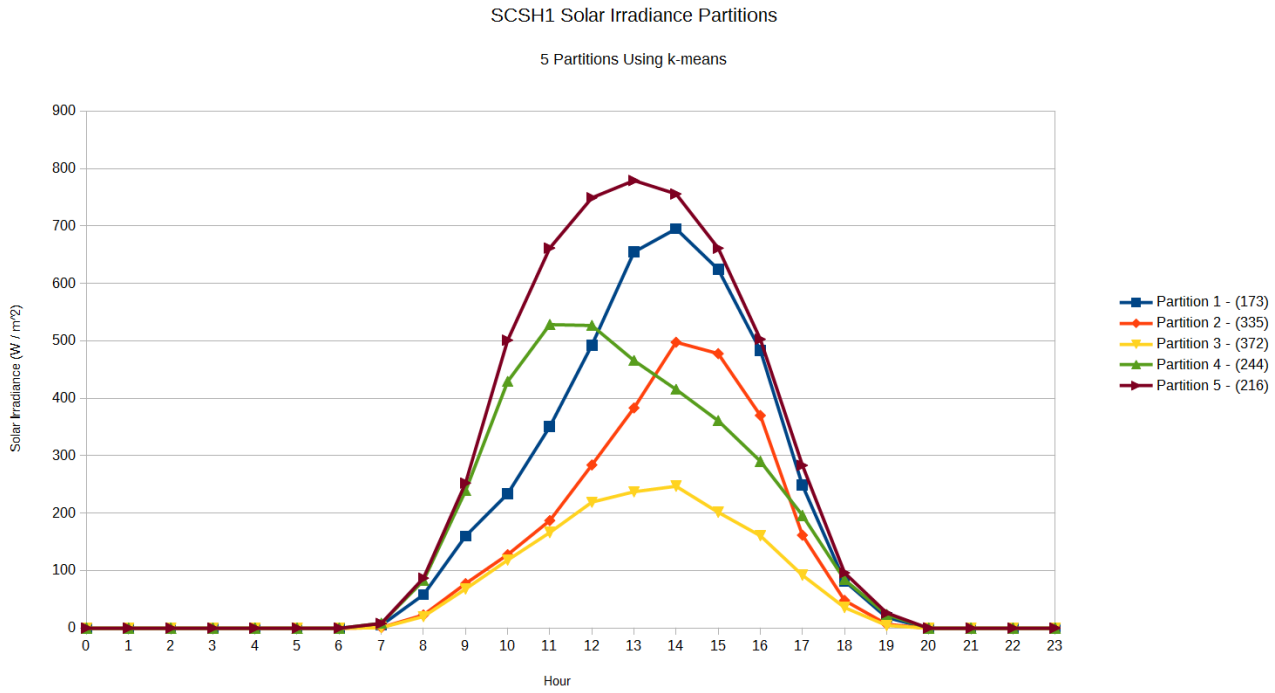
5. Clustering of Daily Solar Irradiance Curves

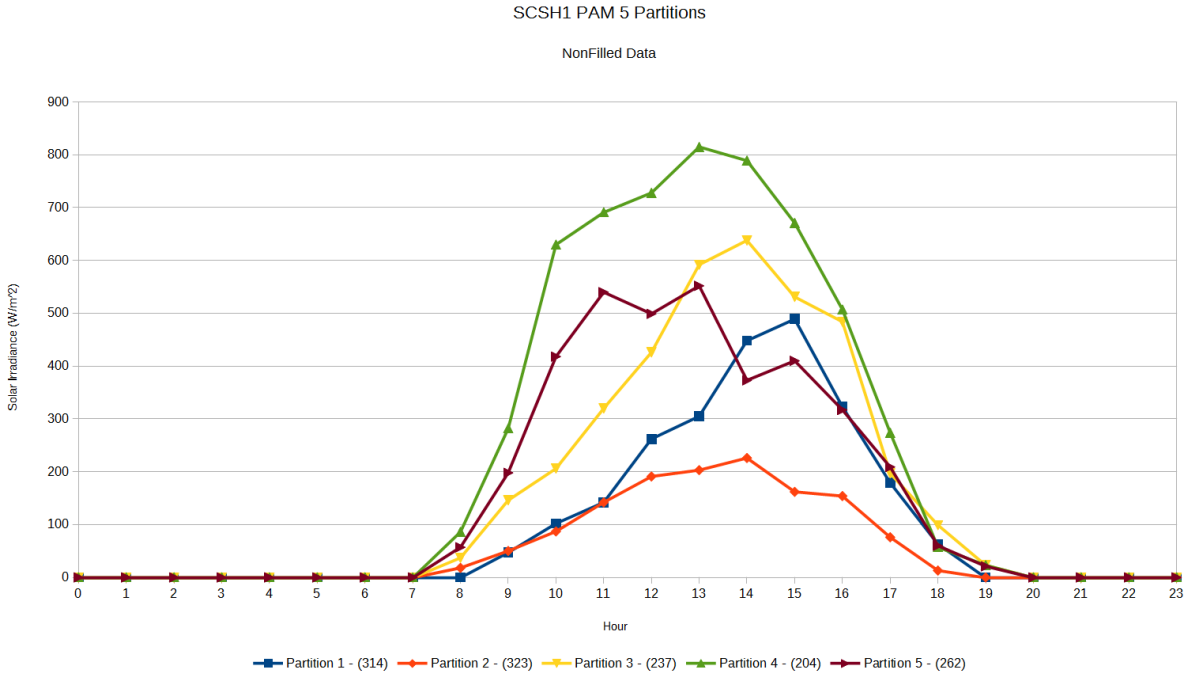
The second focus of our efforts was to detect and classify daily solar irradiance patterns (ex. sunny days, days that are cloudy all day, days that start sunny then get cloudy, etc.). for these experiments, we once again used the MesoWest dataset. To find these daily patterns we aimed to apply two clustering algorithms to a transformed set of data. In our transformation, we adjusted the data to form daily solar irradiance vectors where each vector contains the solar irradiance for a single day from hour 0 (midnight) to 23 (11 pm) so that the clustering can be performed on these vectors and cluster daily solar irradiance vectors rather than hourly values. Once the vectors were created, we then

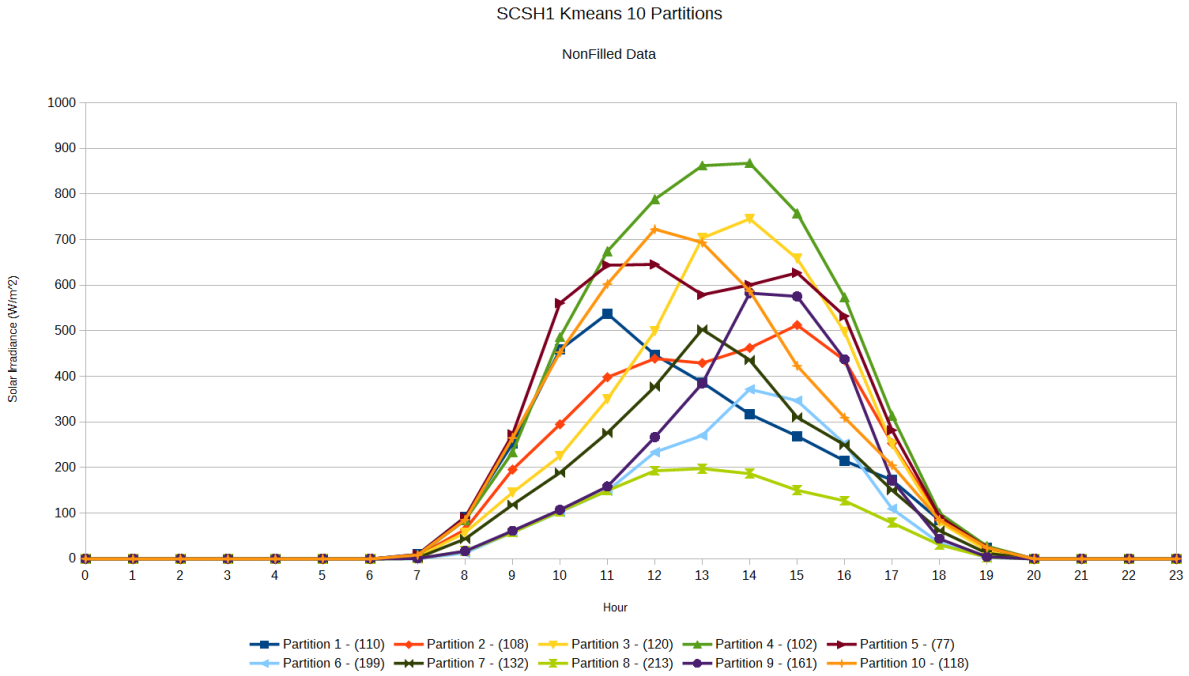
ran one of two clustering algorithms, either k-means or partitioning around medoids (PAM), on the vectors and examined the resulting partitions. These algorithms takes a number of partitions (k) as input and attempt to find a set of k partitions that minimizes the distance between the points in each partition. These partitions are then output by the algorithm. Once we have obtained the output, we can use the centers of the partitions, called centroids with the k-means algorithm and medoids with PAM, to visualize the solar irradiance curves and see what daily trends are prevalent. The number of partitions generated greatly influences which trends are present within the resulting curves as generating fewer partitions would cause more curves, with possibly different trends, to be associated with a given partition. As a result, we varied the number of partitions generated from 4 to 10 and examined the partitions generated by each to determine the number of partitions that appears to capture the major trends.

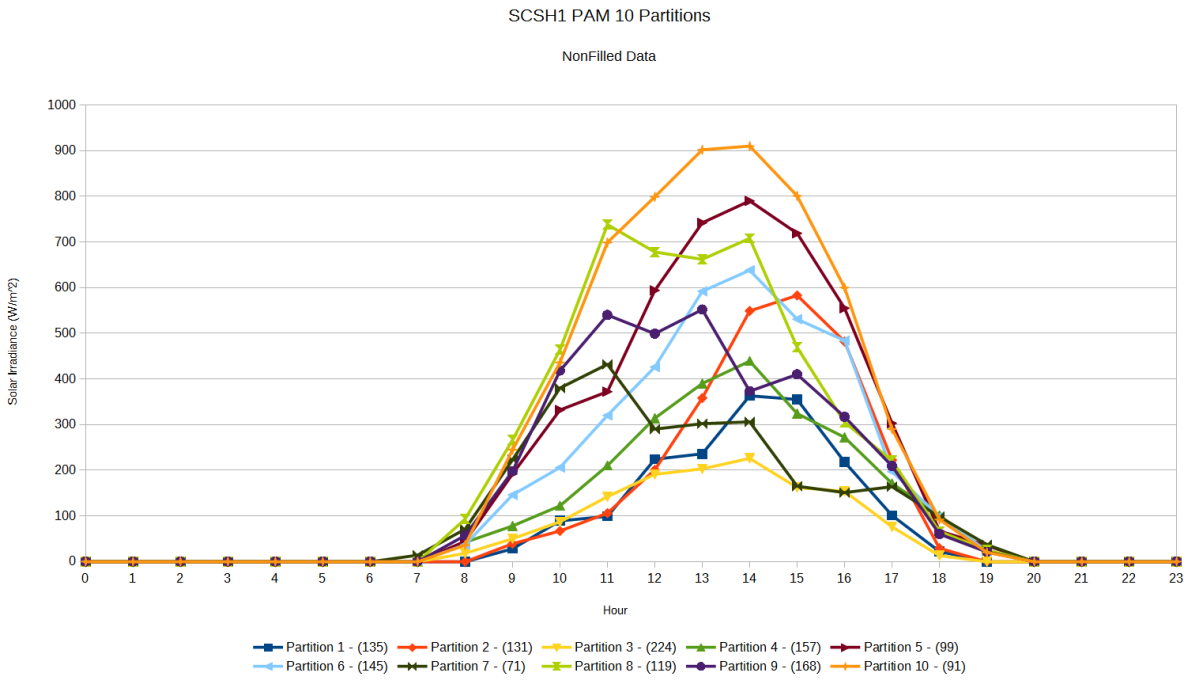
5.1 Resulting Partitions

The partitions that resulted from our clustering algorithms may be seen in the figures below. In general, there are five trends. First, there is always one mostly smooth, bell-shaped curve with a high peak that represents the very sunny days. Similarly, there is also a smooth curve with a small magnitude to represent the overall cloudy days. The rest of the curves generally fall into the other three trends and vary mostly only in magnitude. The third trend represents days with cloudy mornings, but sunny afternoons as the curve grows slowly in the morning and then peaks in the late afternoon. The fourth trend is the opposite as the curve is high in the morning and then tapers off quickly in the afternoon. Finally, the fifth trend is where the curve grows and falls at a similar rate in the mornings and afternoons, but has a dip in the middle of the day. Partitions that fall in this last trend only appear in higher numbers of partitions which suggests that days that follow this pattern are not as common as days that follow the other patterns.


SCSH1 Solar Irradiance Partitions
5 Partitions Using k-means
Solar Irradiance (W / m^2)
900
800
700
600
500
400
300
200
100
0
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
Hour
Partition 1 - (173)
Partition 2 - (335)
Partition 3 - (372)
Partition 4 - (244)
Partition 5 - (216)


SCSH1 PAM 5 Partitions
NonFilled Data
Solar Irradiance (W/m^2)
Hour
Partition 1 - (314)
Partition 2 - (323)
Partition 3 - (237)
Partition 4 - (204)
Partition 5 - (262)

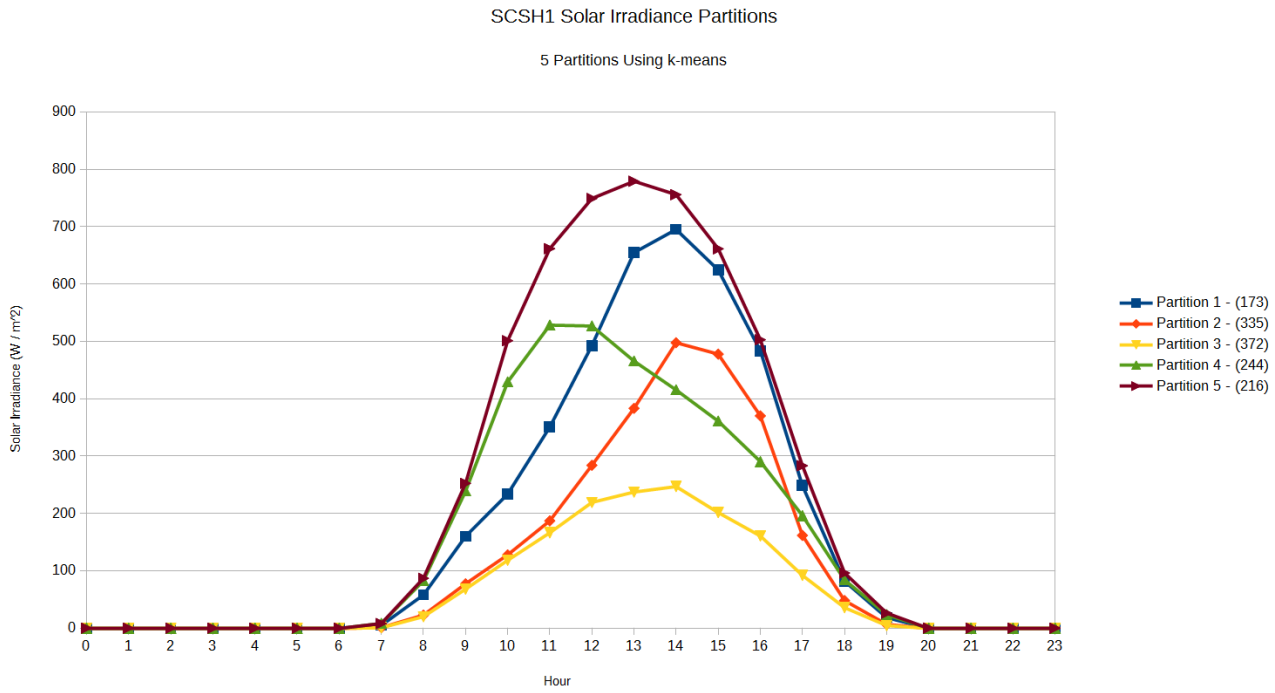

SCSH1 Kmeans 10 Partitions
NonFilled Data
Solar Irradiance (W/m^2)
Hour
Partition 1 - (110)
Partition 2 - (108)
Partition 3 - (120)
Partition 4 - (102)
Partition 5 - (77)
Partition 6 - (199)
Partition 7 - (132)
Partition 8 - (213)
Partition 9 - (161)
Partition 10 - (118)



5.2 Experiments

Now that we have partitioned the solar irradiance to classify each day to match one of several daily solar irradiance curves, we can examine their relationship with both the other partitions and with the other features. The first relationship we explored was the relationship between the assigned partitions of consecutive days as persistence in the assigned partition could prove useful when trying to predict the daily solar irradiance curve that we expect to see in the future. This was examined using both partition chains and transition probabilities. We then extend this notion to other features using feature entropy analysis which would help in exposing the relationships between any of the other features and the assigned partition.

For the following experiments, we chose to use the five partitions obtained using the k-means algorithm as that partitioning captured most of the daily solar irradiance patterns while removing many of the partitions that had similar shapes and only varied based on magnitude. The resulting partitions are shown in the chart below.



5.2.1 Partition Chains

Goal: Determine how persistent the detected solar irradiance patterns are. If they are persistent, the previously observed pattern might be a strong indicator of the pattern that would be observed next and hence provide a level of predictability.

Method: Partition chain length measures the number of consecutive days that were assigned the same partition as calculated using a simple sequential scan of the partition assignments ordered chronologically by date and counting the number of consecutive days with the same assigned partition. From this, we can calculate the average and maximum of these chain lengths and analyze those values. The average chain length would provide a rough idea of how long each of the solar irradiance patterns are likely to last while the maximum chain length provides the upper bound on the persistence of these patterns.

Results: The table below shows the results of this experiment. The average chain length for most partitions was about 2-3 which implies that there is about one or two days of persistence in terms the daily solar irradiance pattern. Partition 3 (the overall low magnitude curve) had the longest average chain length which shows that we are more likely to see a string of cloudy days (perhaps due to a storm or weather system moving through) than a string of very sunny days. However, the maximum chain lengths are much larger than the average which suggests that the partitions follow persistent weather phenomena that can last for a week or two.

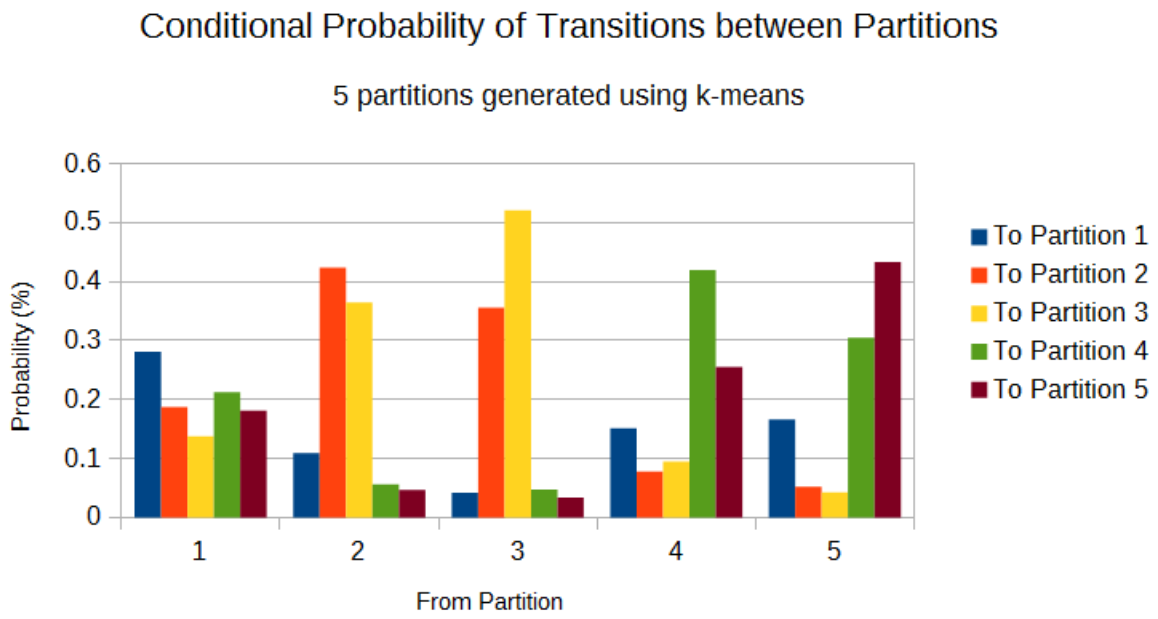
	Partition 1	Partition 2	Partition 3	Partition 4	Partition 5
Average Chain Length	2.286	2.863	3.583	2.732	2.717
Maximum Chain Length	5	11	13	6	11

5.2.2 Transition Probability

Goal: To determine the probability that we would observe certain partition next given the previously assigned partition. This would provide a more concrete measure of the predictability of daily solar irradiance patterns in the future.

Method: The transition probability was calculated by sorting the assigned partitions chronologically by date and scanning the resulting list. During the scan, we counted the number of times we observed each pairing of assigned partitions for the current day and the next day. We then divided these counts by the number of times the particular partition was observed to obtain a probability.

Results: The transition probabilities between the partitions are shown in the chart below. While the probabilities are comparable when transitioning from partition 1, there are notable spikes in the other partitions. Invariably, the probability that the next day would be in the same partition is the highest which reinforces the average partition chain lengths that observed. It also appears that partitions 2 and 3 are likely to transition between each other. Partitions 2 and 3 are the two lower magnitude partitions which suggests that the peak irradiance is still likely to be low during the next day if we observe a low irradiance pattern. There is a similar, albeit weaker, trend for partitions 4 and 5 which suggests that cloudy afternoons are more likely than cloudy mornings if clear weather was previously observed.



5.2.3 Feature Entropy Analysis

Goal: To identify and examine other features that have a strong relationship with the assigned partition and hence have a strong relationship with the daily solar irradiance pattern. The strength of the relationships is measured using entropy. Entropy is a measure of uncertainty of a variable based on the uniformity of a set of previously observed values. Larger entropy values indicate a higher uncertainty due to a large variance in the previously observed values (i.e. there are many different values) while a low entropy indicates that the values are more uniformed. This notion of uniformity can then be used to ascertain the strength of the relationship between a variable and the daily solar irradiance pattern (i.e. the assigned partition) as variables that are uniformed within each partition have a strong relationship because certain values of the variable are strongly associated with certain partitions. Furthermore, observing a low entropy could also signify that there is a non-uniformity between the partitions as ideally the partitions should experience this uniformity with different ranges of values with one partition having the majority of its observations in one range while another partition has most of its observations in a different range. If such a trend is observed, then it is a strong candidate as a predictor of the assigned partition since the data shows that we are more likely to see a certain partition given a value within a certain range.

Method: To accomplish this, we created daily vectors for each feature in a similar manner to what we did with the solar irradiance in the initial clustering step. We then take the average of these vectors to obtain a single daily average for each feature of each vector. We can then find the entropy of each of the daily averages for each feature. However, entropy must be calculated using discrete values and the averages are continuous. Consequently, we discretize these daily into 10 evenly distributed bins

and calculate the entropy for each partition using these bins. This will result in a set of entropy values for each feature (one per partition) and we can average the entropy values across the partitions to obtain a single average entropy value for the feature. Once the average entropy values have been calculated, we can examine the partition distributions of the features with the lowest entropy by graphing the number of observations in each interval of the distribution for each partition to visually extract patterns from the partitionings.

Feature	**Entropy averaged over 5 partitions**
Precipitation	0.0186615298
Relative Humidity	0.9082955671
Wind Direction	1.1311558666
Wind Speed	1.7573246131
Temperature	1.7967060444
Month	1.8585408212

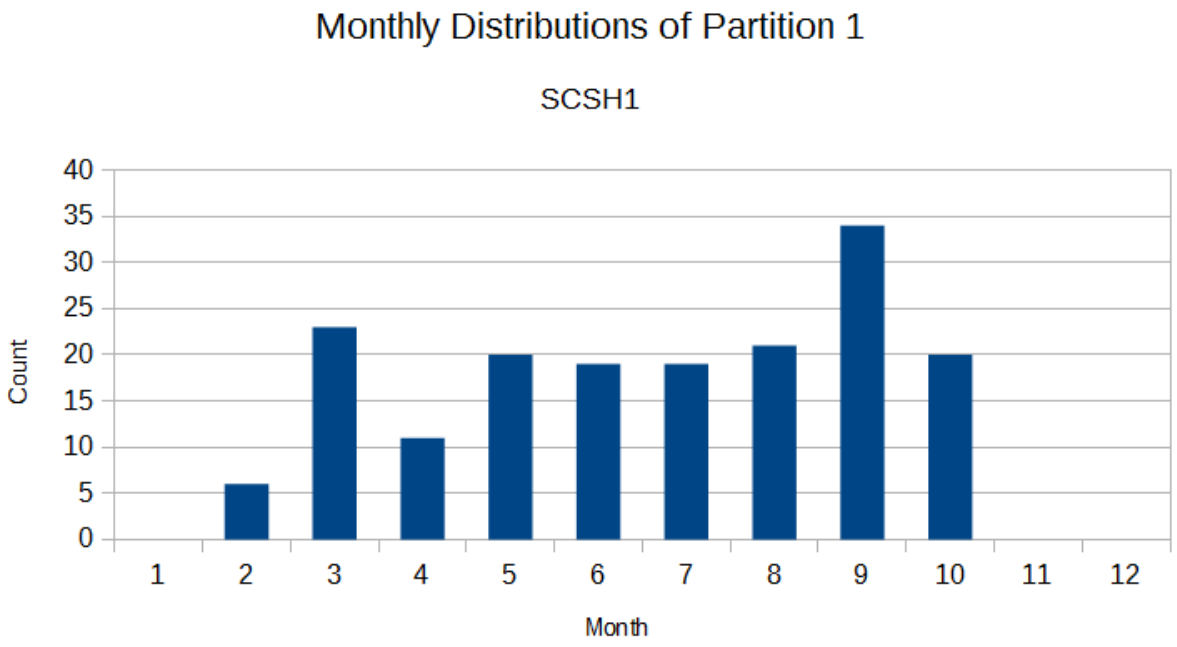
In general, features with an entropy less than 2 were considered for examination. While this technique did help us discover the patterns between a some of the features and the assigned partition, the discretization method used incorrectly caused a few features to have a lower entropy than expected. Since the bins were evenly assigned across the observed range of daily averages outliers would skew the ranges of the bins and if the outliers are especially large the majority of the values will be assigned to a single bin making the distributions of the values very uniformed within each partition, thus making the entropy very low even though that uniformity is artificial. For example, the distributions for the precipitation may be seen in the table below. Here, the majority of the observations fall within the first range of [0, 1.22] and there are only 4 observations that fall outside of that range. Yet, since the overall range must account for those outliers, the overall range of all of the bins is expanded to include values up to 12.2 and the outliers are assigned to the appropriate bins even though those other bins are almost

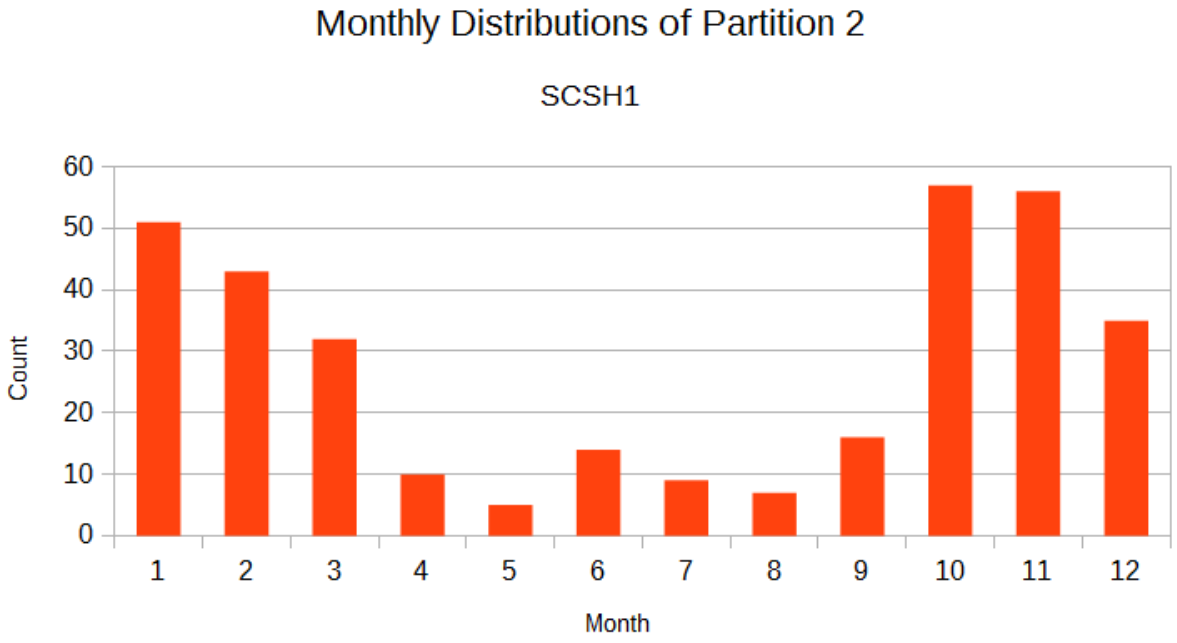
empty. As a result, the entropy analysis on the precipitation feature did not yield any useful results because binning only shows that the precipitation value are almost always within the [0, 1.22] range regardless of the partition. A similar issue occurred with the relative humidity as, once again, outliers caused the majority of the observations to be discretized into the first bin.

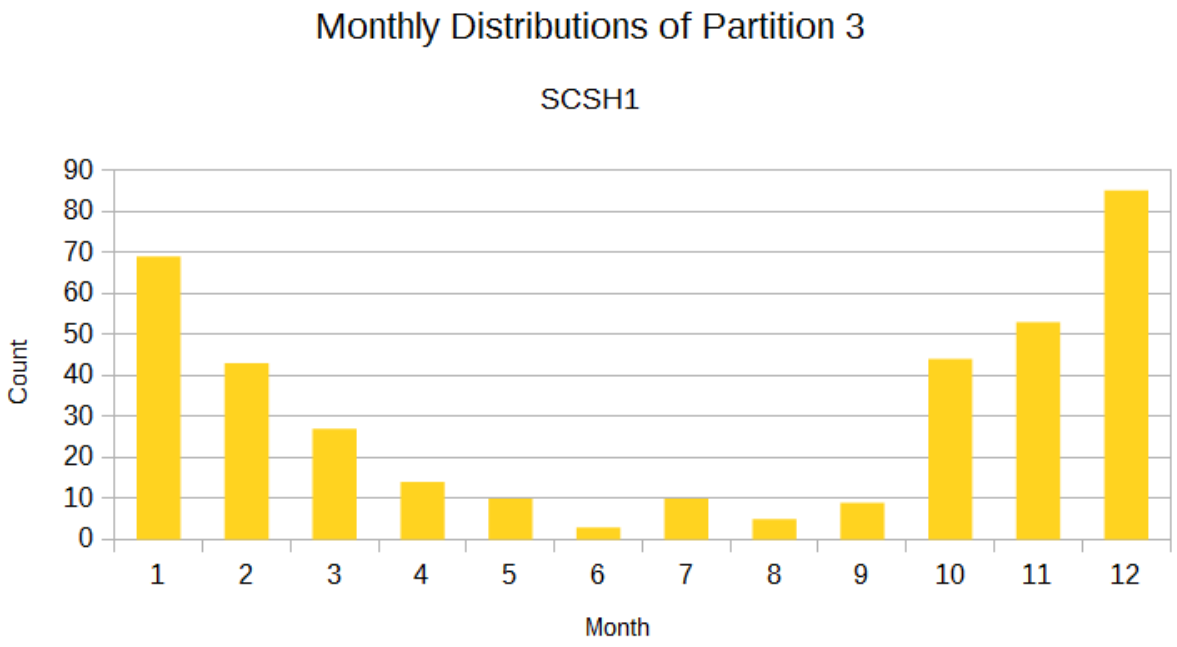
Range of Precipitation Values	**Partition 1**	**Partition 2**	**Partition 3**	**Partition 3**	**Partition 5**
[0,1.22]	173	335	370	243	215
(1.22,2.43]	0	0	0	1	0
(2.43,3.65]	0	0	1	0	0
(3.65,4.86]	0	0	0	0	1
(4.86,6.08]	0	0	0	0	0
(6.08,7.29]	0	0	0	0	0
(7.29,8.51]	0	0	0	0	0
(8.51,9.72]	0	0	0	0	0
(9.72,10.9]	0	0	0	0	0
(10.9,12.2]	0	0	1	0	0

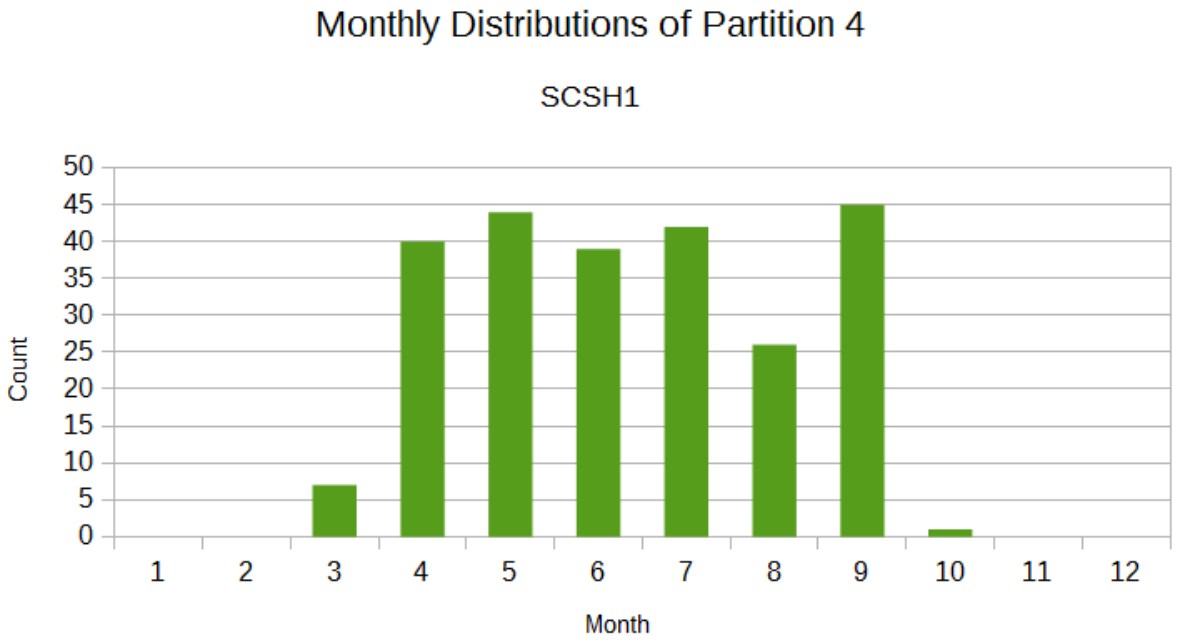
Some of the features that exhibited notable patterns resulting from our entropy analysis will be discussed in further detail below.

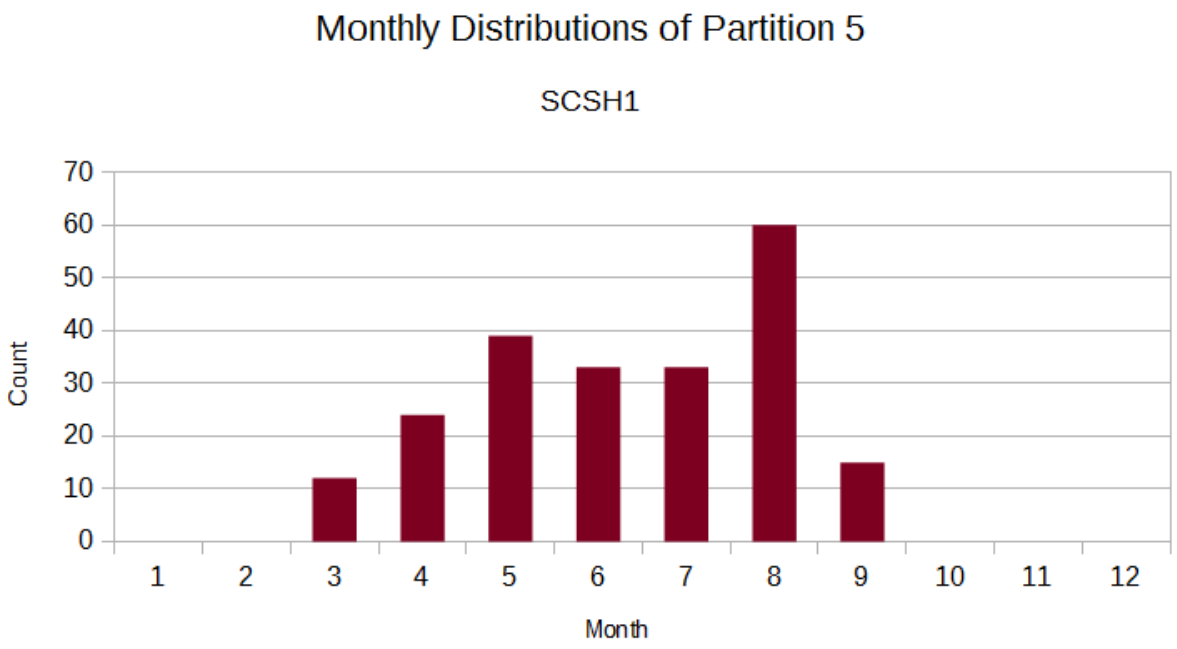
Month: The first features that displayed a noteworthy trend with the assigned partitions is the month of day. The chart below shows the monthly distributions of the partitions where the x-axis is the month and the y-axis is the number of observations from that month were binned in each cluster. The interesting trend here is that partitions 2 and 3 (which are the lower overall solar irradiance partitions) are predominantly found in the late Fall and Winter months. Similarly, the higher irradiance partitions (4 and 5) are found mostly in the Spring and Summer. These trends seem to follow the seasonal weather patterns experienced on Oʻahu. During the Winter, we tend to have more rain and hence see more cloudy (partition 3) days. Conversely, it tends to be very sunny and hot during the summer which is where most of the high irradiance days (partitions 4 and 5) are observed.


Monthly Distributions of Partition 1
SCSH1
Count
Month
0
5
10
15
20
25
30
35
40
1
2
3
4
5
6
7
8
9
10
11
12

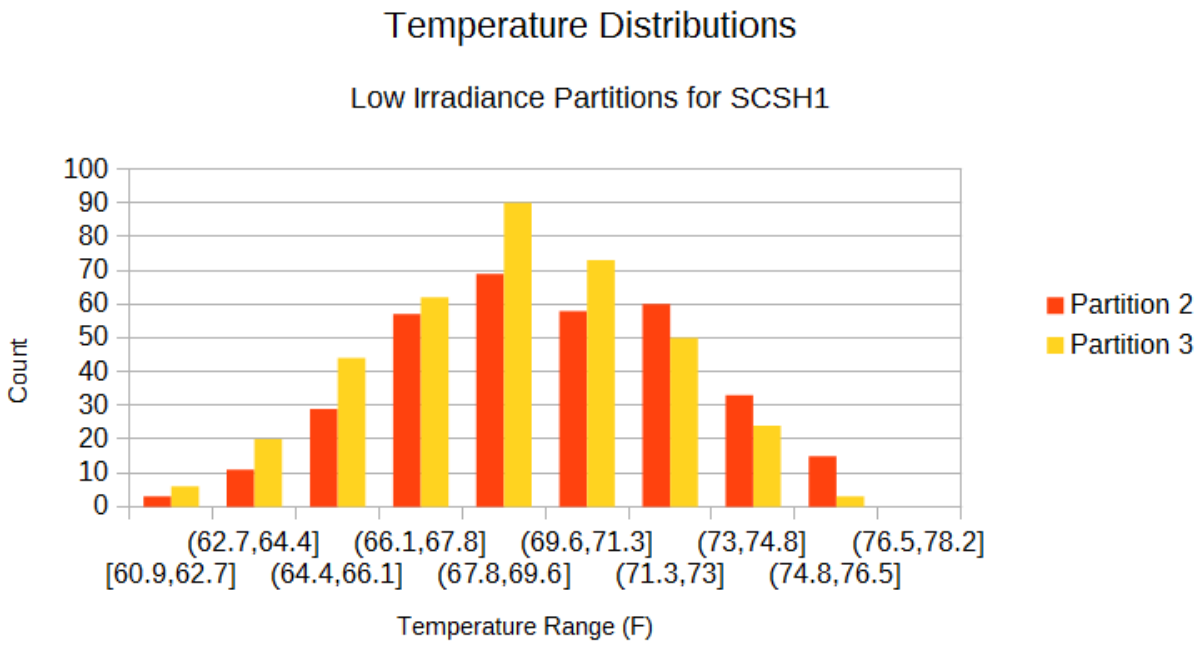

Monthly Distributions of Partition 2
SCSH1
Count
Month
0
10
20
30
40
50
60
1
2
3
4
5
6
7
8
9
10
11
12

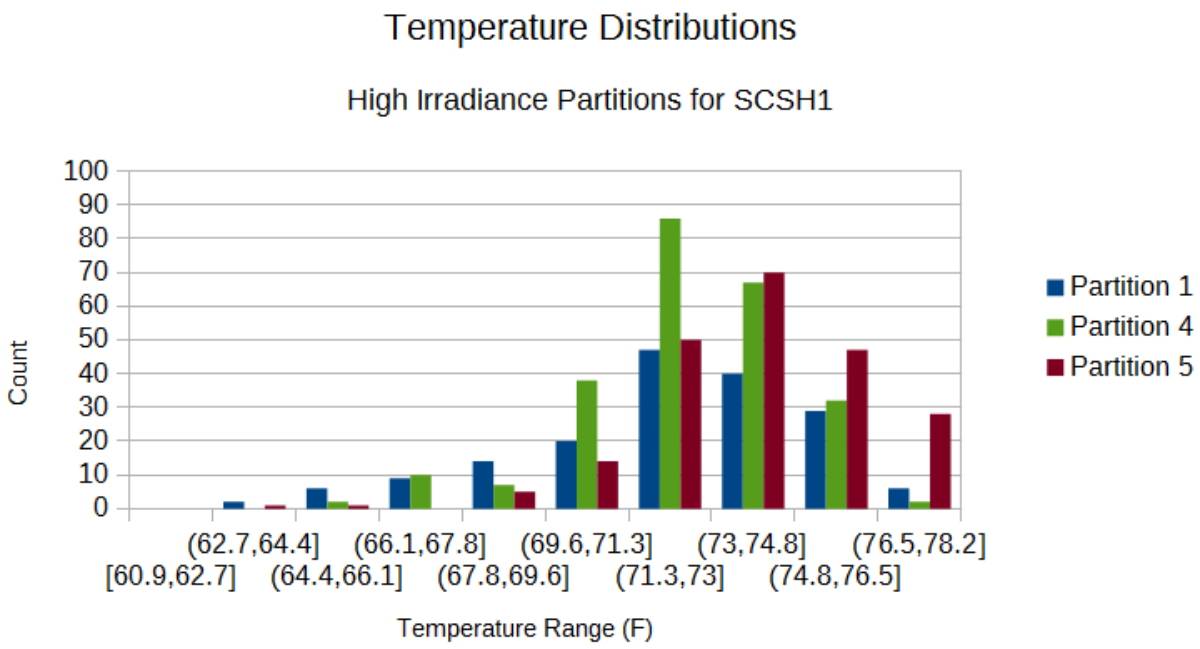

Monthly Distributions of Partition 3
SCSH1
Count
90
80
70
60
50
40
30
20
10
0
1
2
3
4
5
6
7
8
9
10
11
12
Month


Monthly Distributions of Partition 4
SCSH1
Count
50
45
40
35
30
25
20
15
10
5
0
1
2
3
4
5
6
7
8
9
10
11
12
Month

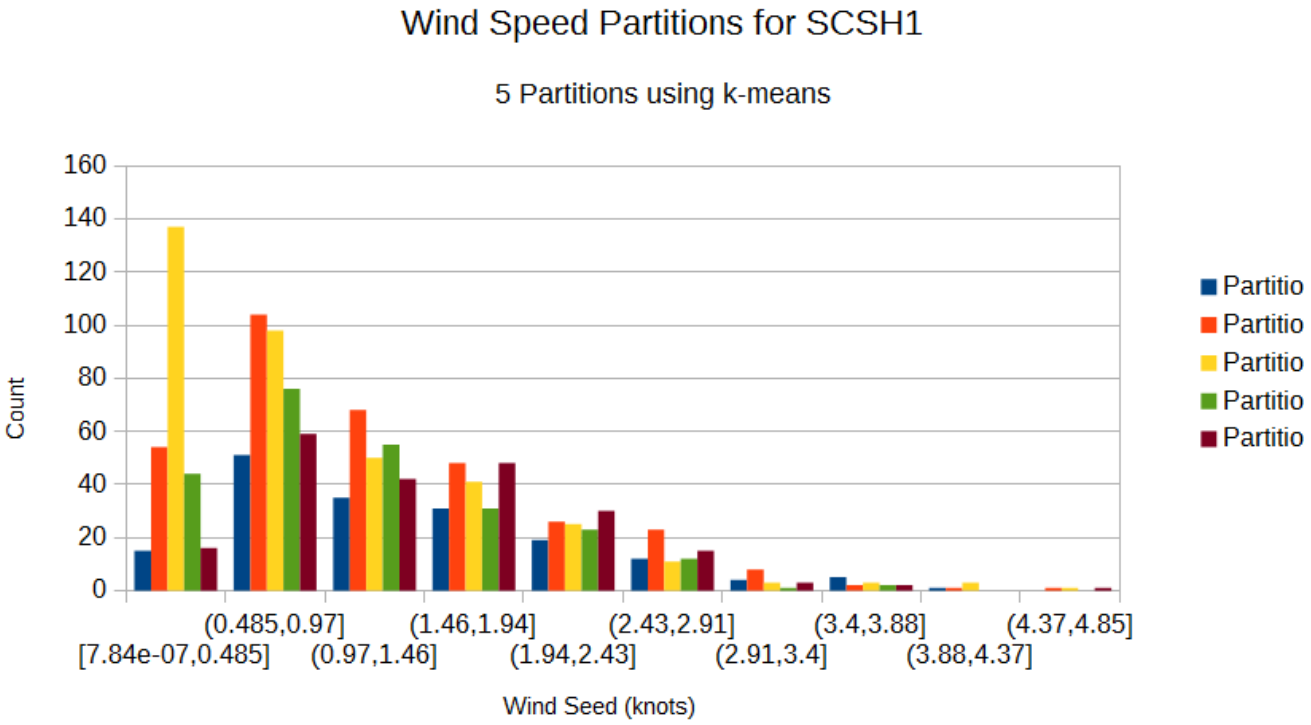


Temperature: The relationship between the temperature and the partition is not a surprising one. The lower solar irradiance partitions are more prevalent at the lower temperatures while the higher solar irradiance partitions tend to frequent the higher temperature ranges. This is no surprise as the higher the solar irradiance, the higher the temperature would be due to heating from the Sun.

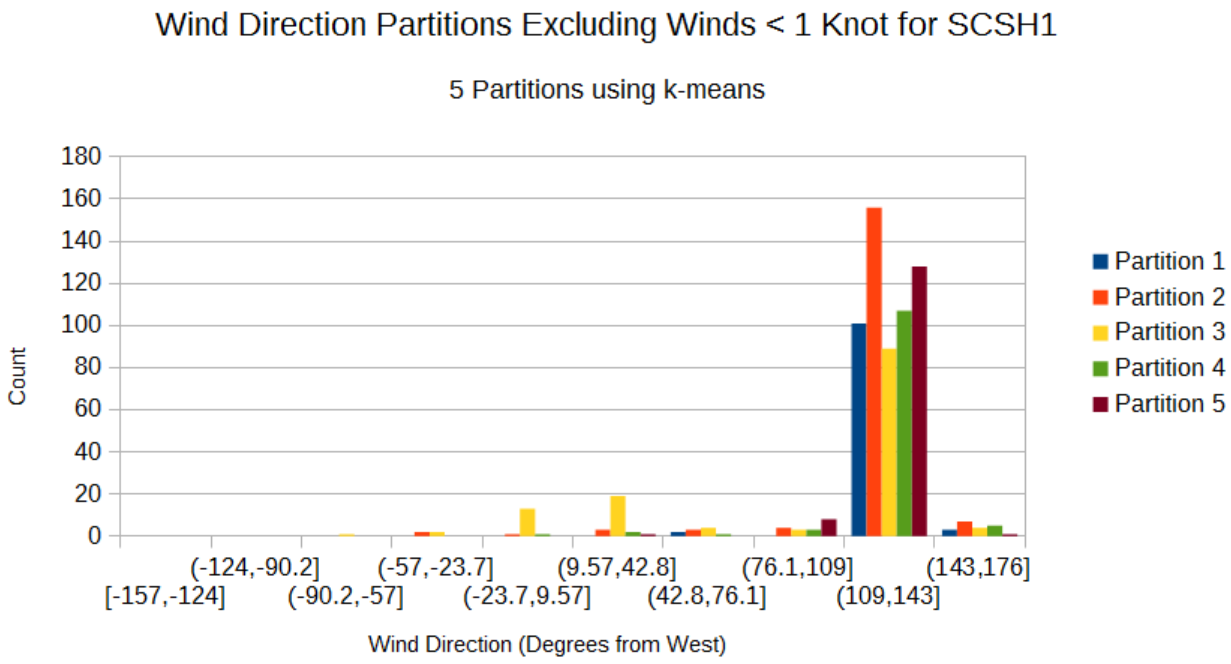




Wind Speed: The chart below shows the distributions of the wind speeds. The trend for wind speed is that the winds tend to be low with the majority of the winds being around 1 knot or less. It also seems that the low irradiance partitions are more prevalent at the lower wind speeds. This makes sense as low winds would cause the clouds to be more stationary which would cause the irradiance to be lower throughout the day.



Wind Direction: Finally, the chart below shows the distributions of the averaged wind directions for days that had a wind speed larger than 1 knot (0.514 m/s) to eliminate variable winds. The vast majority of the winds fall into the range 109 to 143 degrees from West. When converted to the more common degree system with 0 degrees being North, this range becomes 19 to 53 degrees. This means that most of the winds are coming from North North-East which would be tradewinds. This is somewhat inline with the observed wind patterns as tradewinds tend to be the most common wind pattern on O‘ahu, but these tradewinds tend to come from a more Easterly direction than what was observed. A possible explanation is that this experiment was only run on a single station though and thus it could be a result of the topography around this station.



6. Clustering on Other Features

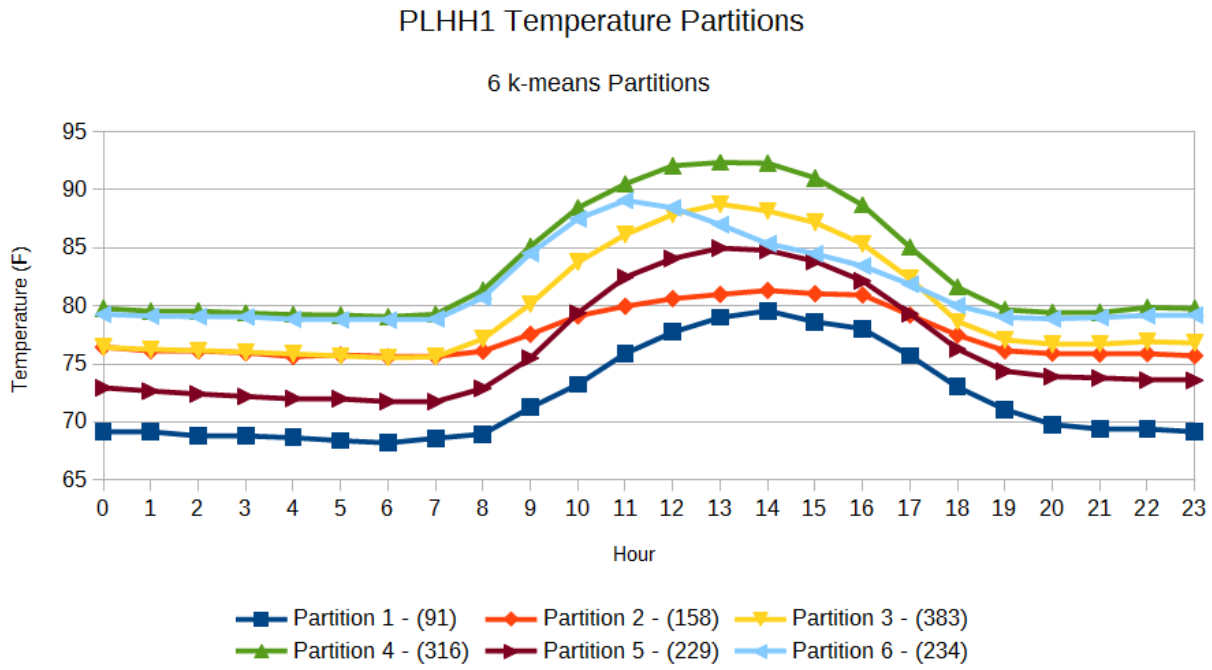
The method of performing daily vectors can also be applied to other features to extract the different daily trends in those features. As with the solar irradiance clustering, we transformed the data for the target feature into daily vectors, ran the k-means algorithm with 4 to 10 partitions, and then plotted the centers of the resulting partitions to visualize the trends.

The clustering for these features was performed on two new stations, PLHH1 and KTAH1, due to data quality issues with the majority relative humidity values taken at the SCSH1 station. Consequently, all experiments that involve the use of clusterings on features other than the solar irradiance will be run for the PLHH1 and KTAH1 stations instead of the SCSH1 station.

The following sections will discuss each of the additional features in detail. Note that the number in parenthesis in the legend of each chart is the number of days that were classified as being a part of that partition. For example, a legend key of “Partition 1 - (91)” indicates that 91 of the days examined were classified as being in partition 1.

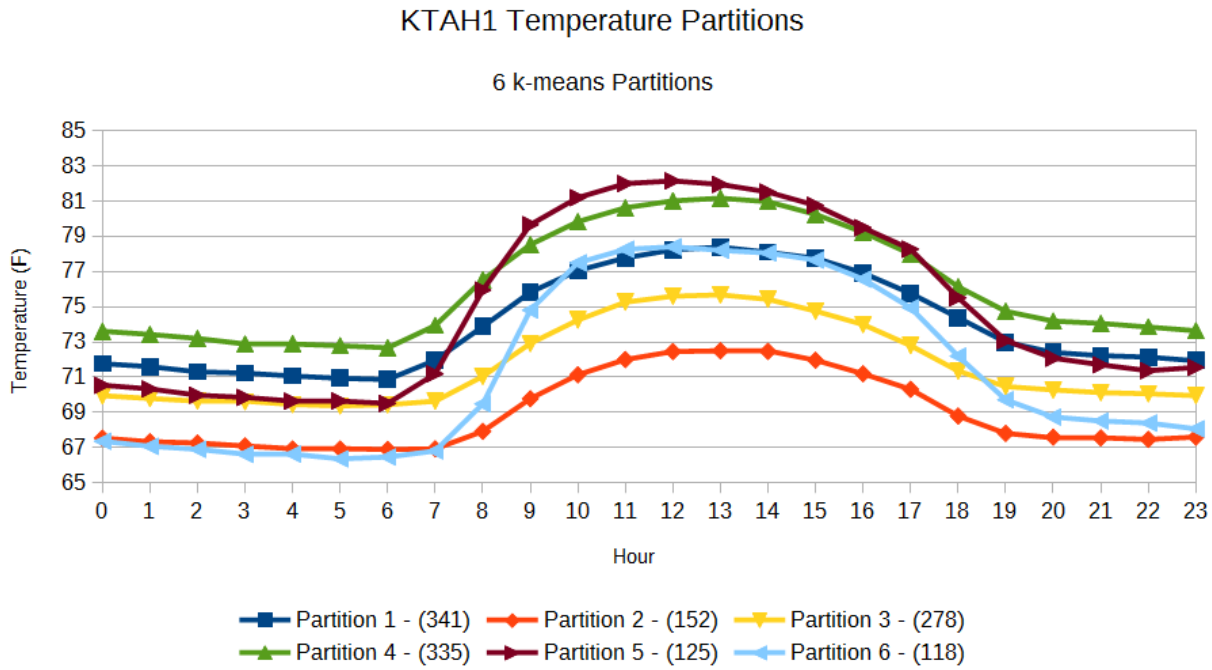
6.1 Temperature

6.1.1 PLHH1 Temperature Partitions



For PLHH1, the temperature exhibits three main trends. The first trend is exhibited by partitions 1, 3, 4, and 5 which is a smooth curve that peaks in the middle of the day around noon. This trend is expected as the temperature rises due to the heating from the sun and the differences in the magnitude of these curves can be attributed to seasonal tendencies. The second trend is found in partition 2 where the temperature increases during the day, but is relatively flat. This likely represents days that are cloudy or rainy where the entire day is relatively cool. The final trend is shown in partition 6 where the temperature rises in a manner similar to the first trend, but peaks earlier in the day and decreases quickly. This trend shows a warm morning and a cooler afternoon which likely indicate that clear morning with a cloudy / rainy afternoon.

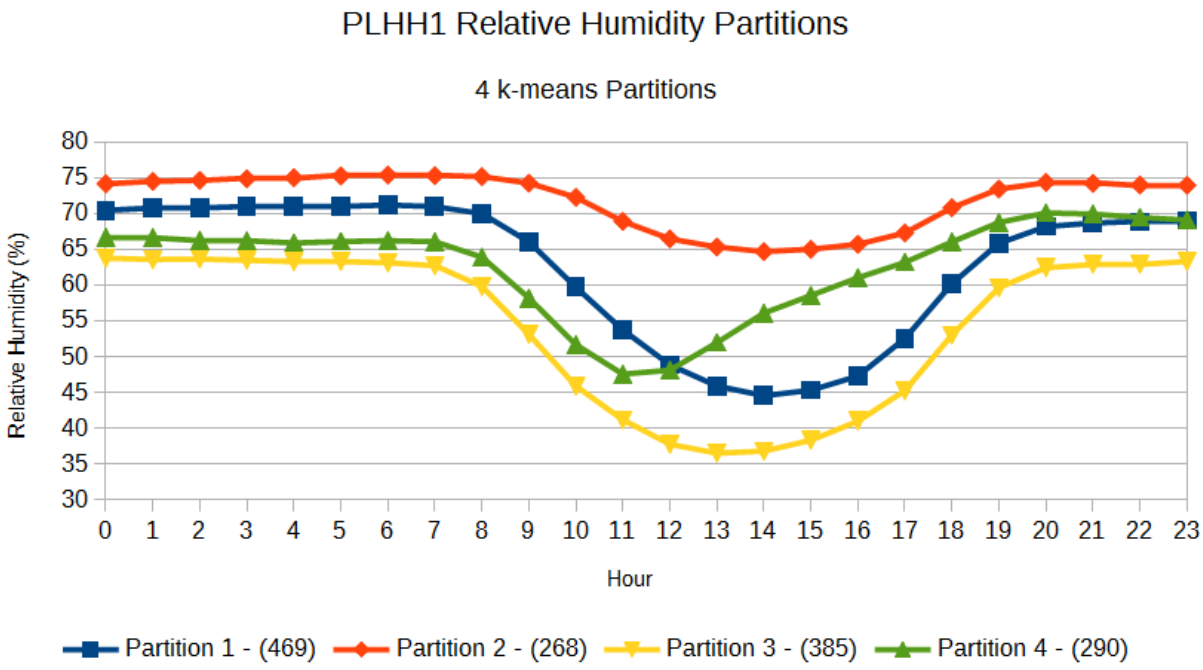
6.1.2 KTAH1 Temperature Partitions



There are two main trends in the temperature partitions for KTAH1. The first is the smooth curve with a peak in the middle of the day which is shown by partitions 1, 2, 3, and 4. This trend is similar to the one observed with the PLHH1 partitions. The other trend is one that was not observed in PLHH1 and is exhibited by partitions 5 and 6. In these partitions, the temperature increases very quickly in the early morning, slows down in the late morning, peaks in the middle of the day, and then slowly decreases until night time. Unlike the partitions for PLHH1, all of the partitions for KTAH1 are bell shaped which implies that it is almost always very sunny at KTAH1.

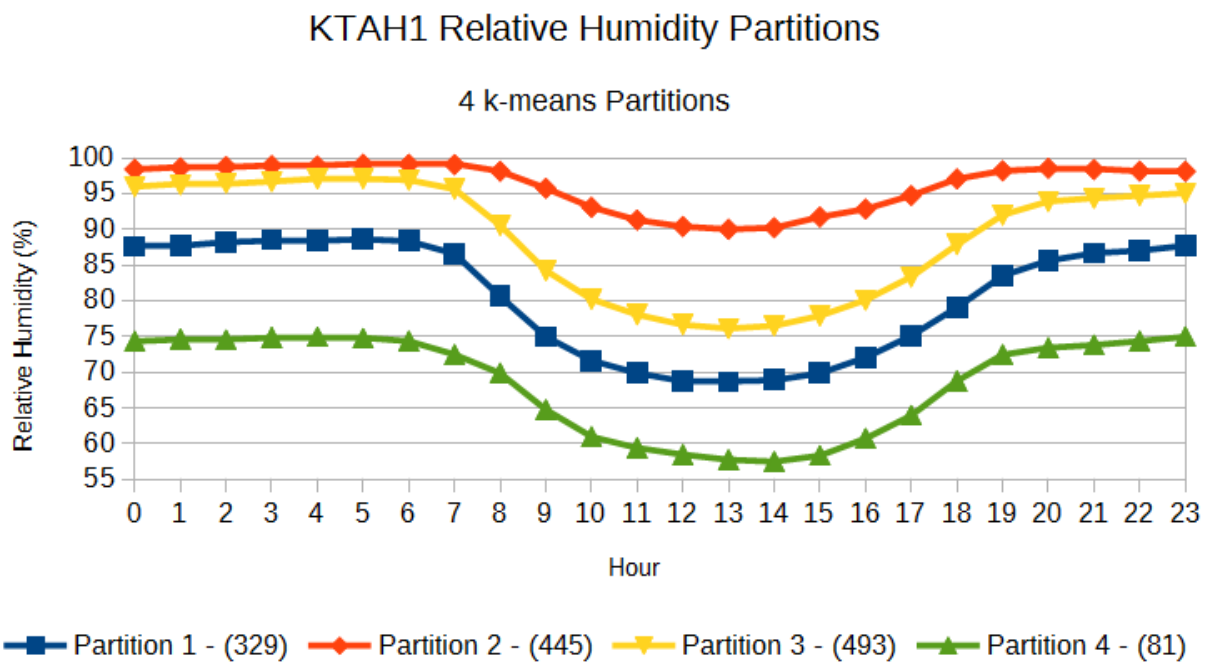
6.2 Relative Humidity

6.2.1 PLHH1 Relative Humidity Partitions



In general, the relative humidity curves for PLHH1 start with a high relative humidity value and decrease during the day as the temperature increases and the air becomes drier. Nonetheless, there are three major trends in this decrease in relative humidity. Partitions 1 and 3 show a large drop with the minimum being in the middle of the day. This large decrease implies that these trends likely correspond with the warmer sunnier days. On the other hand, partition 2 experiences a relatively small decrease which suggests that partition 2 be indicative of a cloudy or rainy day. Finally, partition 4 shows a large drop in the morning and a steady increase in the afternoon which likely indicates a sunny morning with a cloudy / rainy afternoon.

6.1.2 KTAH1 Relative Humidity Partitions

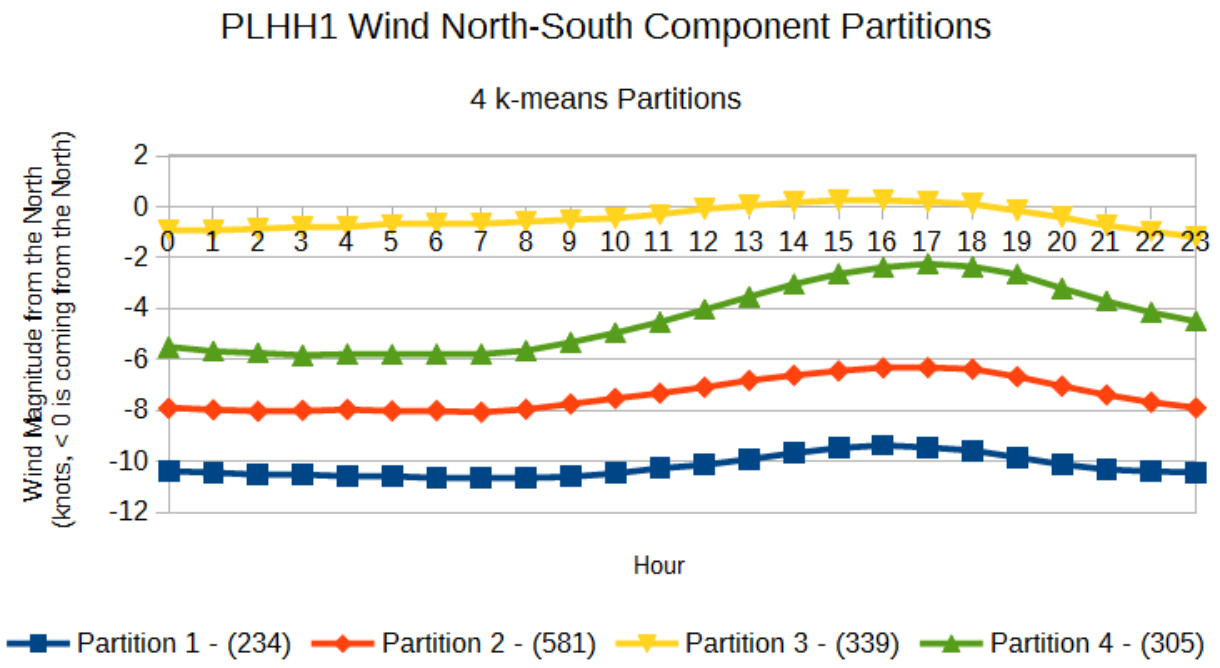


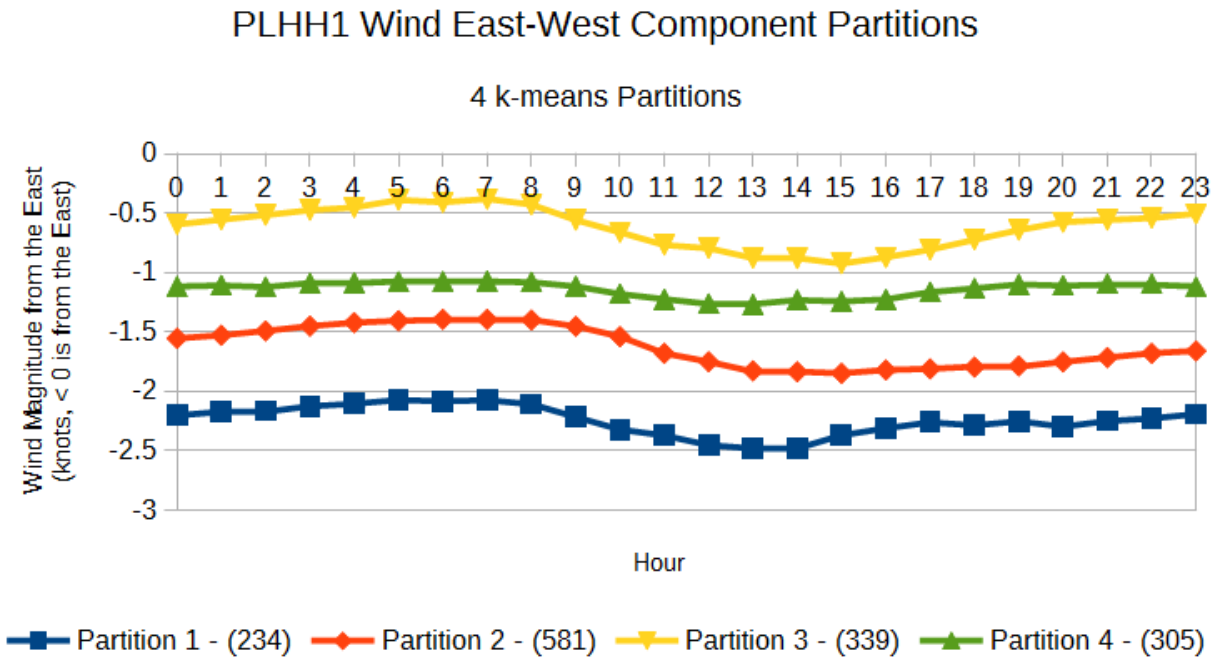
The relative humidity partitions for KTAH1 show two trends, both of which were also observed in the PLHH1 relative humidity partitions. The first trend is shown by partitions 1, 3, and 4 where there is a smooth decrease with a minimum at the middle of the day when it is the hottest. The other trend is from partition 2 where the relative humidity drops during the day, but the decrease is small. As with the temperature partitions, the smooth, bell-shaped nature of all of the relative humidity curves implies that KTAH1 is normally sunny all day long.

6.3 Wind

To cluster the wind variables we separated the wind vectors into their North-South and East-West components to create two 24 value vectors (one value for each hour of the day). We then combined these components into a single vector (with 48 values) to ensure that both components were taken into account and performed the clustering on that vector. For visualization purposes, the single 48 value vector has been separated back into the North-South and East-West wind components.

6.3.1 PLHH1 Wind Partitions

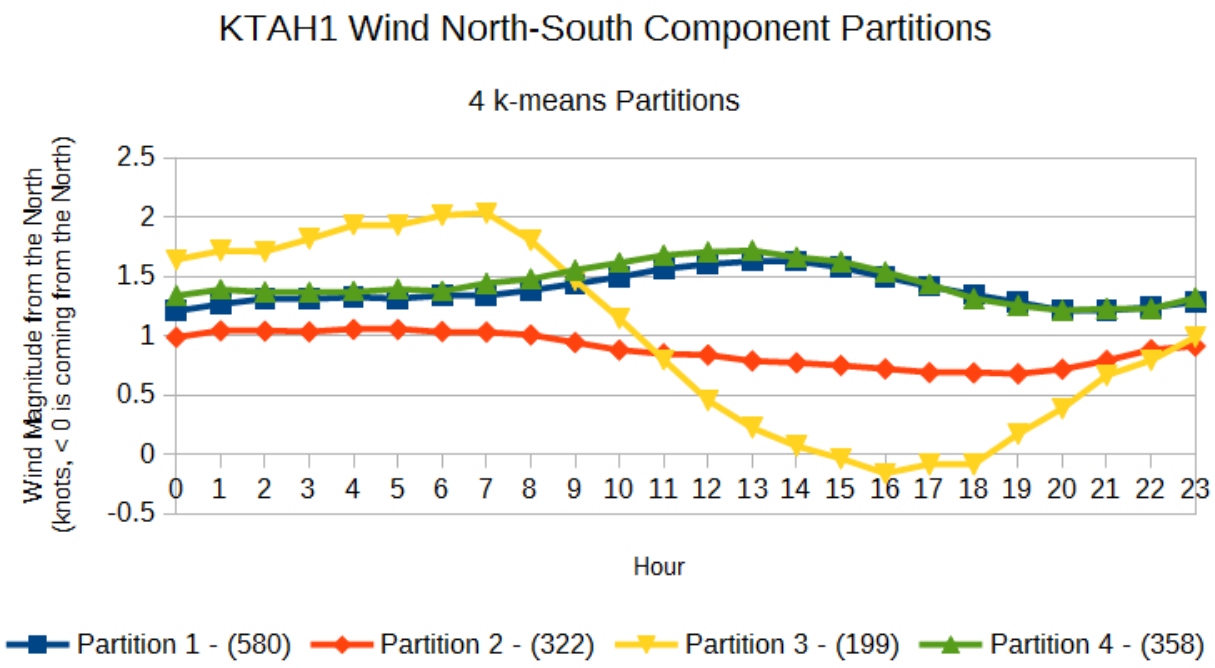


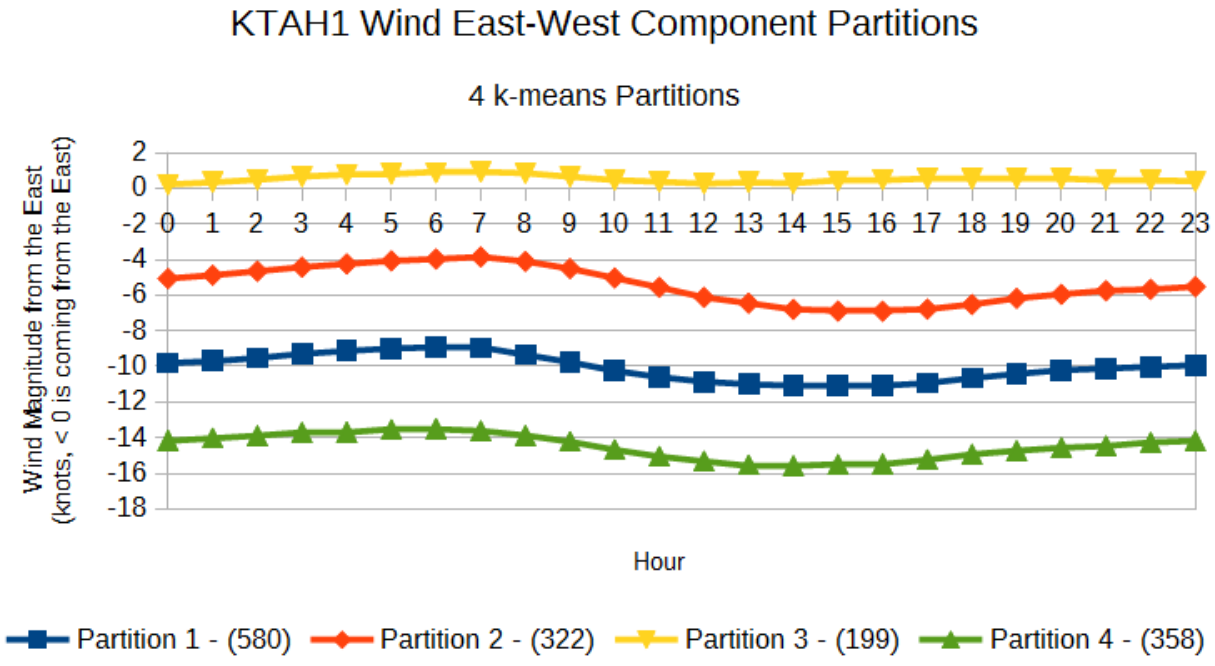


The wind partitions at PLHH1 appear to show two tendencies. Partitions 1, 2, and 4 show a strong wind coming from the North-East which would fall into the common tradewinds pattern. The magnitude of the wind from the North also seems to be much larger than from the East which indicates that the northern component would dominate (i.e. the wind would come mostly from the North). Partition 3 exhibits a different trend as

it has a much smaller overall wind magnitude and shows the days that experienced light and variable winds.

6.3.2 KTAH1 Wind Partitions





In general, the wind partitions for KTAH1 show consistent magnitudes for both components throughout the day. The exception is partition 3 which, while consistent in the East-West component shows a large drop in the North-South component in the middle of the day until the evening. Partition 3 is also the closest partition to the light and variable winds that were observed at PLHH1 even though its North-South component has a rather large magnitude during the morning. All other partitions show winds coming from the South-East which is peculiar since tradewinds should come from the North-East. The component from the South tends to be much smaller than the component from the East which may indicate that this might be a product of the geography around the station as it is located near the base of the northern tip of the Ko‘olau mountain range.

7. Cluster Predictions

An alternative to predicting the solar irradiance on an hour-by-hour basis would be to predict a solar irradiance curve for the entire day. While such predictions would lose resolution compared to the hour-by-hour predictions, the shape of the curve can be used to inform grid operators as to when energy generation from solar energy sources should be high or low and allow them to act accordingly. Consequently, we have explored the use of two techniques that can be used to predict the expected solar irradiance partition: a simple conditional probability classifier and the naive Bayes classifier.

7.1 Conditional Probability Classifier

As discovered in our analysis of the solar irradiance clustering for SCSH1 (section 4.2.2) there are some strong biases in probability of the next day’s solar irradiance partition given the solar irradiance partition that was observed the day before. The conditional probability classifier exploits this bias by choosing the most likely partition (i.e. the partition with the highest conditional probability) given the previous partition observed.

To train the classifier, the conditional probability of the first day before is calculated empirically by counting the occurrences of each pairing of the partitions at the date to be predicted and the day before. To add further days to the data window two approaches can be taken. The first approach is to empirically calculate the conditional probability of a partition given the previous two days in a similar manner to using one day of history with the only difference being that triples of partitionings are counted instead of pairs. This approach does not make any assumptions on the probabilities, but it might not be effective if the training data set is too small as the counts might be spread out too thinly amongst the triples. The second approach is to approximate the conditional probability by assuming stationarity and that the conditional probability remains the same regardless of the number of days before the prediction. This approximation can be

calculated with the following equations where P0 is the partition at the time that we want to predict, P1 is the partition one day before, and P2 is the partition two days before.

$P(P0 \mid P1, P2) = P(P0, P1, P2) / P(P1, P2)$

$P(P0, P1, P2) \sim= P(P0, P1) * P(P1, P2) / P(P1)$

Assume: $P(P0, P1) = P(P1, P2)$

Unlike the empirical calculation, the approximation only needs to find the partition pairings for one day before (i.e. P(P0, P1)) and hence is less sensitive to the size of the training data set. However, it does have the additional limitation that it can only use one set of partitions for its predictor features as the approximation will only be valid if the same partitioning as the one used to empirically calculate the pairing probabilities. Both of these approaches will be evaluated in the experiments. While the conditional probability classifier is only using the solar irradiance partitions as its features, the number of partitions used to perform the prediction can be modified and the effects of modifying the number of partitions in the features will be explored in our experiments.

7.2 Naive Bayes Classifier

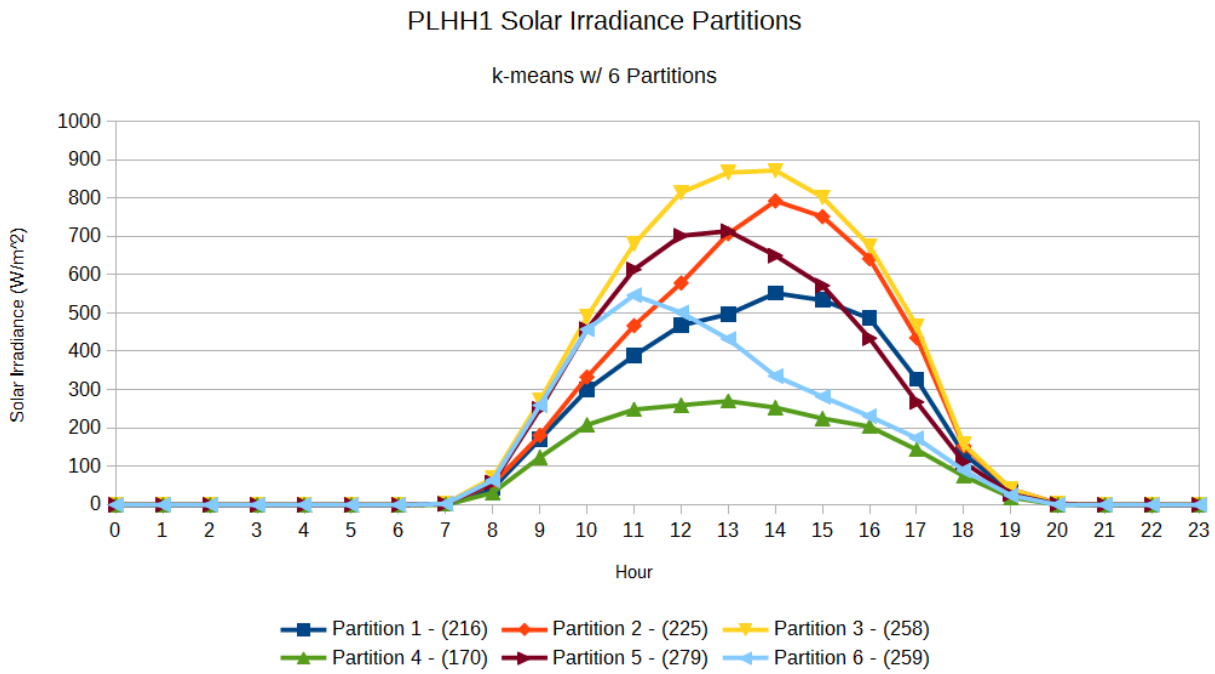
The other classifier used is the naive Bayes classifier which is a probabilistic classifier that applies Bayes' theorem while assuming an independence amongst all of the features. For the naive Bayes classifier, we used the partitionings of other features in addition to the solar irradiance. In our experiments, those additional features are the three features discussed in section 5: relative humidity, temperature, and wind. These features were hand selected as they were strong candidates for having a strong correlation with the solar irradiance due to physical reason.

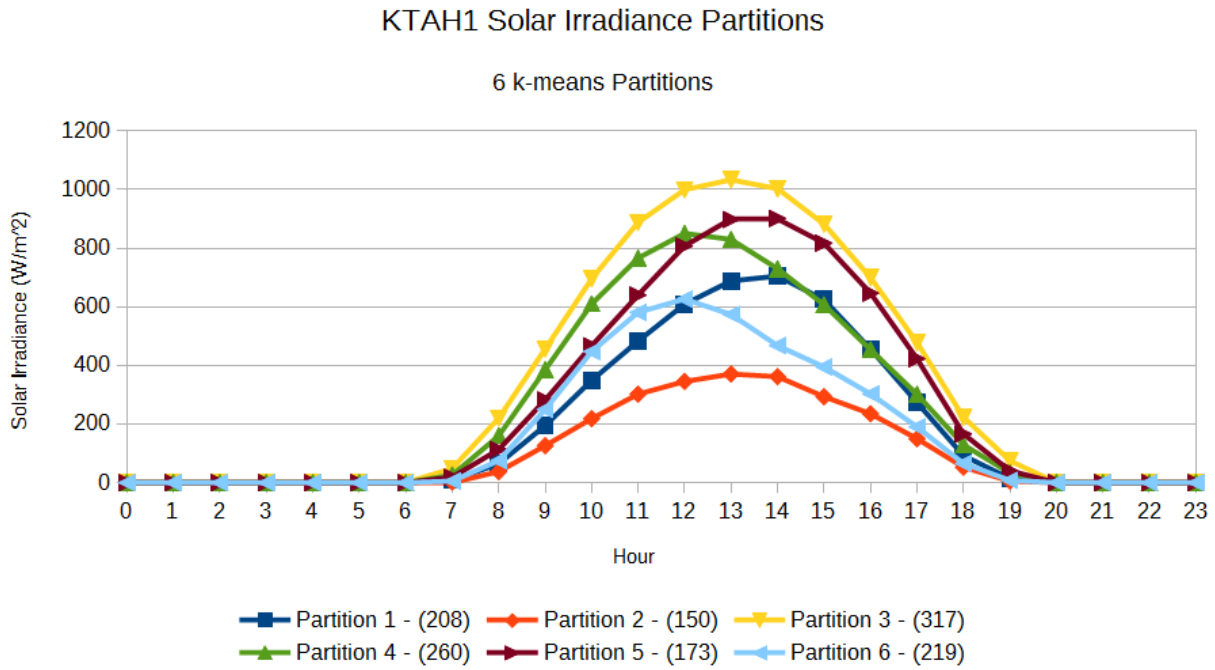
It is possible that using all features could confuse the classifier so all combinations of the features were tested and to find the best combination of features. As with the conditional probability classifier, the number of partitions for the features used could have an effect on the accuracy of the resulting model. While an exhaustive search could be used to find the absolute best combination of features with the best number of partitions for each feature, the time required to perform such an exhaustive scan would be impractical. Therefore, we use a greedy method of selecting the best number of partitions by running the test where the feature is the only feature for each possible number of partitions and selected the lowest mean absolute error for use in the aforementioned feature combination finding step. In the case where more than one day for the feature is present in the data window, each day would be considered to be a separate feature. For instance, if we wanted to use the relative humidity partitions from both one day and two days before, we would have two relative humidity features where one is for one day before while the other is for two days before.

7.3 Experiments

7.3.1 General Method / Metrics

All of the following experiments were run and evaluated using the following method. The models were trained using three years of data (2010 - 2012) and tested using one year of data (2013). To evaluate the accuracy of the models, we found the difference between the centers of the predicted partition and the centers of the actual partition and took the absolute value of this difference. We then found the mean of these errors for the daytime hours (hour 7 to hour 20 or 7 am to 8 pm) to get the mean absolute error. The models were used to predict a solar irradiance partition from a set of six partitions generated using the k-means algorithm unless stated otherwise. The six partitions for the PLHH1 and KTAH1 stations may be seen in the charts below. A full listing of the solar irradiance partitions used may be seen in the appendix.





Finally, all feature combinations will be represented in the following format and the abbreviations of the feature names may be found in the table below:

<Name of the Feature to Predict>_<Number of Partitions> <- <Feature 1 Name>_<Number of Partitions>_<Days Before the Prediction>, <Feature 2 Name>_<Number of Partitions>_<Days Before the Prediction>, …

Full Feature Name	**Abbreviation**
Solar Irradiance	SOLR
Temperature (Fahrenheit)	TMPF
Relative Humidity	RELH
Wind (North-South and East-West components combined into a single vector)	WIND

7.3.2 Conditional Probability Classifier

Size of Data Window - 1 day vs. 2 days

PLHH1 Empirical Conditional Probability	

Classifier - 1 Day Window	
Features	**Mean Absolute Error (W/m^2)**
SOLR_6 <- SOLR_7_1	88.8259518328
SOLR_6 <- SOLR_9_1	90.4467282347
SOLR_6 <- SOLR_6_1	91.8608492315
SOLR_6 <- SOLR_5_1	94.7942834542
SOLR_6 <- SOLR_8_1	95.1611343525

PLHH1 Empirical Conditional Probability Classifier - 2 Day Window Empirical	
Features	**Mean Absolute Error (W/m^2)**
SOLR_6 <- SOLR_8_1, SOLR_4_2	85.0123968512
SOLR_6 <- SOLR_7_1, SOLR_5_2	89.9404587885
SOLR_6 <- SOLR_7_1, SOLR_4_2	90.0458856739
SOLR_6 <- SOLR_7_1 + SOLR_7_2	90.3644608902
SOLR_6 <- SOLR_9_1 + SOLR_4_2	90.6454994783

KTAH1 Empirical Conditional Probability Classifier - 1 Day Window	
Features	**Mean Absolute Error (W/m^2)**
SOLR_6 <- SOLR_10_1	92.5328771105
SOLR_6 <- SOLR_6_1	97.3483350294
SOLR_6 <- SOLR_7_1	97.9746628501
SOLR_6 <- SOLR_8_1	98.1333758113
SOLR_5 <- SOLR_5_1	98.3876649423

KTAH1 Empirical Conditional Probability Classifier - 2 Day Window	
Features	**Mean Absolute Error (W/m^2)**
SOLR_6 <- SOLR_5_1, SOLR_4_2	89.8302549838
SOLR_6 <- SOLR_6_1, SOLR_4_2	90.978248234
SOLR_6 <- SOLR_5_1, SOLR_5_2	92.0265124932
SOLR_6 <- SOLR_6_1, SOLR_9_2	92.3161945008
SOLR_6 <- SOLR_6_1, SOLR_6_2	93.9986487512

The top 5 results for both data window sizes of one day and two days for both the PLHH1 and KTAH1 stations may be seen in tables above. The conditional probabilities for these classifiers were found empirically. Invariably, using a window size of two days is better than using a window size of one day as mean absolute errors for the best classifier is always lower when a window size of two is used. The improvement in both cases is not very large (approximately 3.81 W/m^2 or about 4.29% for PLHH1 and approximately 2.70 W/m^2 or about 2.92% for KTAH1), but it is non-negligible and it implies that using an extra day in the data window adds a noticeable amount of information to the classifier. Hence, a data window of at least two should be used hereafter.

Empirical Conditional Probability vs. Approximations

PLHH1 Approximated Conditional Probability Classifier - 2 Day Window	
Features	**Mean Absolute Error (W/m^2)**
SOLR_6 <- SOLR_7_1 + SOLR_7_2	89.356178031
SOLR_6 <- SOLR_9_1 + SOLR_9_2	90.9955879396
SOLR_6 <- SOLR_6_1 + SOLR_6_2	92.2313270558
SOLR_6 <- SOLR_5_1 + SOLR_5_2	94.9502902074
SOLR_6 <- SOLR_8_1 + SOLR_8_2	95.3213821565

KTAH1 Approximated Conditional	

Probability Classifier – 2 Day Window	
Features	**Mean Absolute Error (W/m^2)**
SOLR_6 <- SOLR_10_1 + SOLR_10_2	90.2385662633
SOLR_6 <- SOLR_6_1 + SOLR_6_2	96.6375899605
SOLR_6 <- SOLR_8_1 + SOLR_8_2	96.6622412119
SOLR_6 <- SOLR_7_1 + SOLR_7_2	96.701254514
SOLR_6 <- SOLR_5_1 + SOLR_5_2	97.8202105598

The two tables above show the top 5 results when the approximated conditional probabilities are used. In general, the classifiers that used the empirically calculated probabilities were more accurate, but the degree varies. For PLHH1, the empirical classifier had a mean absolute error of about 4.34 W/m^2 or about 4.86% less than the approximated classifier. On the other hand, the empirical classifier only had an improvement of approximately 0.41 W/m^2 or about 0.45% over the approximated classifier. Even so, the approximated classifier never outperformed the empirical classifier in our tests so the empirically calculated probabilities should still be used.

Number of Feature Partitions

We can use the empirical results using a two day window to look for patterns between the number of partitions used for the predictor features and the ranking of that classifier. For PLHH1, it favors more partitions for the first day before the prediction with selections of 7, 8 and 9 partitions while preferring fewer partitions for the second day before the prediction as it selected 4, 5, and 7 partitions. KTAH1 also favors a smaller number of partitions for the second day before the prediction with selections of 4, 5 and 6 partitions along with one instance of a 9 partition select, but it performs better with fewer rather than more partitions for the first day as it selected either 5 or 6 partitions. The conflicting results for the first day and the outliers for the second day make it difficult to assert a global pattern between the number of partitions and the accuracy of the resulting classifier. As a result, this experiment should be conducted on a site to empirically discover the best set of features if a conditional probability classifier is to be used.

7.3.3 Naive Bayes Classifier

Size of Data Window - 1 day vs. 2 days

PLHH1 Naive Bayes Classifier – 1 Day Window	
Features	**Mean Absolute Error (W/m^2)**
SOLR_6 <- SOLR_7_1	86.7094014254
SOLR_6 <- SKNT_N_SKNT_E_7_1 + SOLR_7_1	86.7420294358
SOLR_6 <- RELH_6_1 + SKNT_N_SKNT_E_7_1 + SOLR_7_1	87.8717645769
SOLR_6 <- TMPF_6_1 + SKNT_N_SKNT_E_7_1 + SOLR_7_1	89.4569883447
SOLR_6 <- TMPF_6_1 + SOLR_7_1	91.4942818641

PLHH1 Naive Bayes Classifier – 2 Day Window	
Features	**Mean Absolute Error (W/m^2)**
SOLR_6 <- SOLR_7_1	86.7094014254
SOLR_6 <- SKNT_N_SKNT_E_7_1 + SOLR_7_1	86.7420294358
SOLR_6 <- SKNT_N_SKNT_E_8_2 + SOLR_7_1 + SOLR_5_2	87.5622564841
SOLR_6 <- RELH_6_1 + SKNT_N_SKNT_E_7_1 + SOLR_7_1	87.8717645769
SOLR_6 <- TMPF_8_2 + SKNT_N_SKNT_E_8_2 + SOLR_7_1	88.0479917968

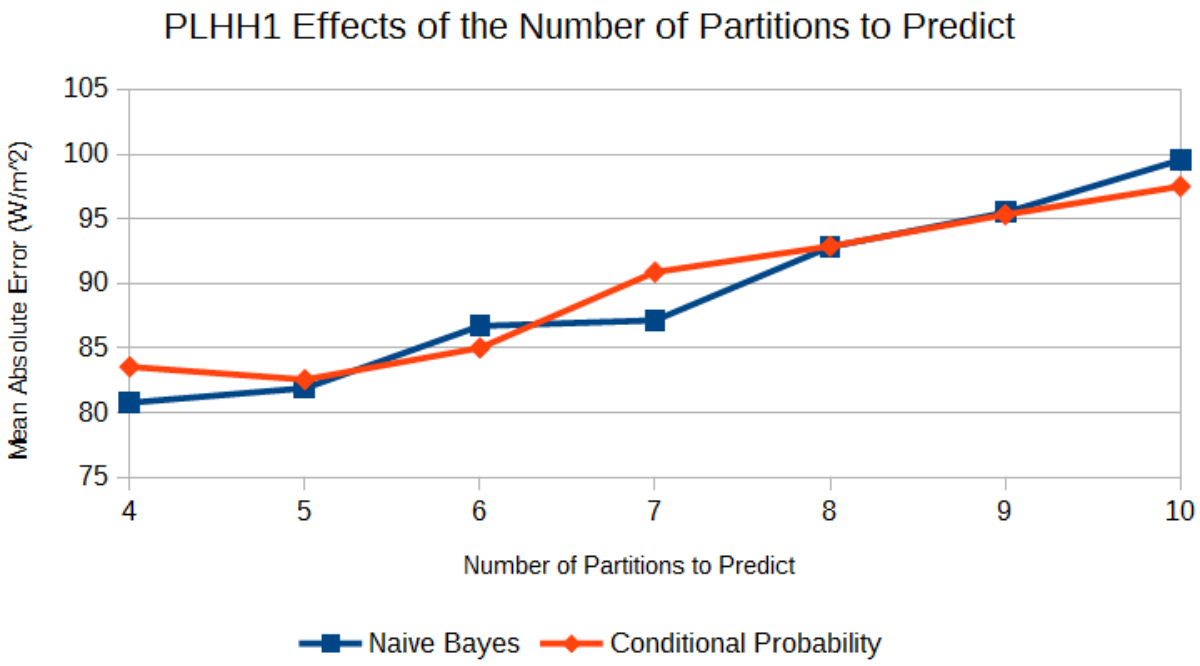
KTAH1 Naive Bayes Classifier – 1 Day Window	
Features	**Mean Absolute Error (W/m^2)**
SOLR_6 <- TMPF_9_1 + SOLR_6_1	87.6091644441
SOLR_6 <- SKNT_N_SKNT_E_8_1 + SOLR_6_1	88.075741259
SOLR_6 <- SOLR_6_1	88.1289916778
SOLR_6 <- TMPF_9_1 + SKNT_N_SKNT_E_8_1 +	88.8831803276

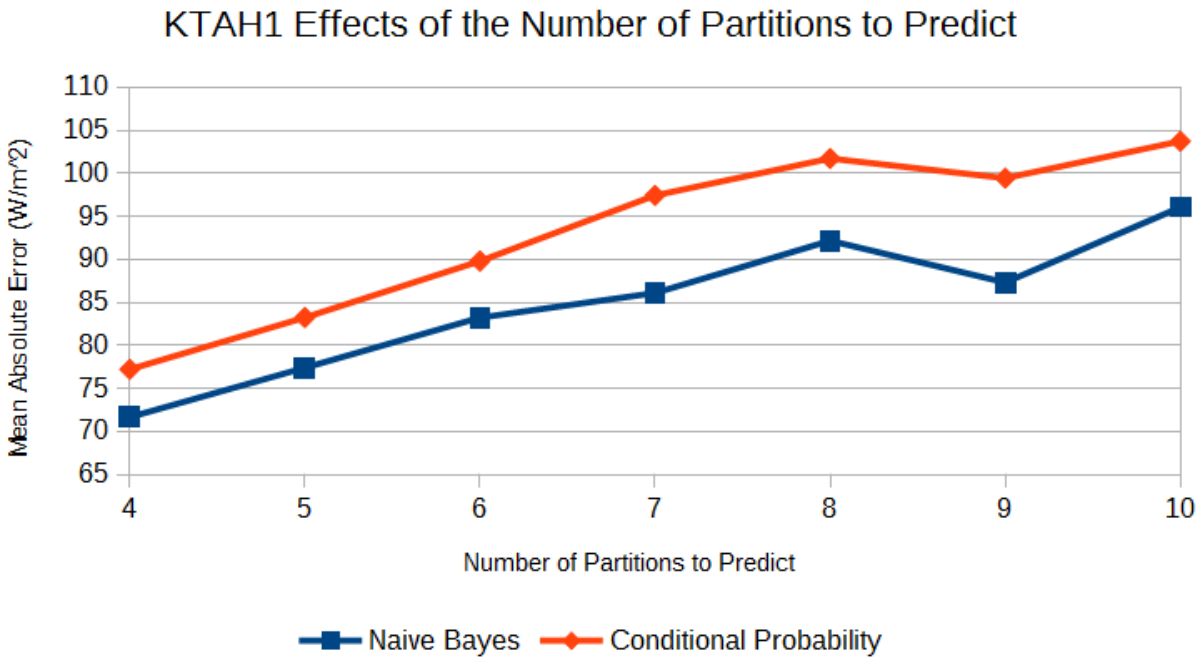
SOLR_6_1	
SOLR_6 <- RELH_9_1 + SOLR_6_1	89.880215286

KTAH1 Naive Bayes Classifier – 2 Day Window	
Features	**Mean Absolute Error (W/m^2)**
SOLR_6 <- TMPF_9_1 + SKNT_N_SKNT_E_9_2 + SOLR_6_1	83.2478246892
SOLR_6 <- RELH_9_1 + RELH_4_2 + TMPF_10_2 + SKNT_N_SKNT_E_8_1 + SKNT_N_SKNT_E_9_2 + SOLR_6_1	83.7630203293
SOLR_6 <- RELH_4_2 + TMPF_9_1 + SKNT_N_SKNT_E_9_2 + SOLR_6_1	84.3768276235
SOLR_6 <- RELH_4_2 + TMPF_9_1 + SKNT_N_SKNT_E_8_1 + SKNT_N_SKNT_E_9_2 + SOLR_6_1	84.7262616655
SOLR_6 <- RELH_4_2 + TMPF_10_2 + SKNT_N_SKNT_E_9_2 + SOLR_6_1	85.0390034776

The use of a one day window or a two day window has mixed results. PLHH1 experienced no improvement at all as the same feature combination (which only included a single feature from one day before the prediction) resulted in the lowest error for both data window sizes. On the other hand, KTAH1 experienced a small improvement as the error was decreased by about 4.36 W/m^2 or 4.98% when the window was expanded. As a result, the naive Bayes classifier and our greedy feature selection method should be used with a data window of two as it can potentially improve the accuracy by a small margin and in the worst case, the result would be the same as if a data window of one day was used. The only scenario where a data window of one day should be used is if the running time of the feature selection algorithm is an issue as increasing the size of the data window increases the number of features and consequently the number of combinations that the feature selection algorithm must search through.

7.3.4 Number of Partitions to Predict





To determine the effects of the number of partitions that the classifier is trying to predict, we ran the tests for the naive Bayes classifier and the conditional probability classifier while varying the number of partitions to be predicted from 4 to 10 using the best configurations as determined from our previous experiments. In general, increasing the number of partitions to predict decreases the accuracy of the classifier. This is expected

as fewer partitions causes more different trends to be merged into a single partition which means that even if the wrong trend is predicted, it could still fall within the correct partition and there would be no error. As the the number of partitions is increased, the trends become separated into their own partitions and incorrectly predicting the wrong trend is more likely to result in predicting the incorrect partition which would be detected as an error. Nevertheless, in practice the number of partitions chosen should depend on which trends need to be detected as while the smaller number of partitions results in fewer misclassifications, the partitions themselves might be too vague to be useful.

7.3.5 Conditional Probability Classifier vs. Naive Bayes Classifier

The charts showing the effects of the number of partitions to be predicted also show the differences in the prediction performance of the conditional probability classifier and naive Bayes classifier. The results vary based on the station in question as the two have a similar accuracy for PLHH1 while the naive Bayes classifier outperforms the conditional probability classifier for all numbers of partitions. Therefore, there it is not clear whether the naive Bayes classifier or the conditional probability classifier is more accurate for all stations and both should be considered for empirical evaluation.

8. Comparison with WRF / 4 Day Forecasts

To examine the accuracy of our methods in relation to existing techniques, we will compare the best configurations as discovered from our experiments with the forecasts of the established WRF model. For the hourly methods, the best configuration is using deseasonalized data and monthly-hourly models without splitting the data based on the wind. The best configurations for the cluster prediction methods was to use a two day window and to use empirically calculated probabilities for the conditional probability classifier.

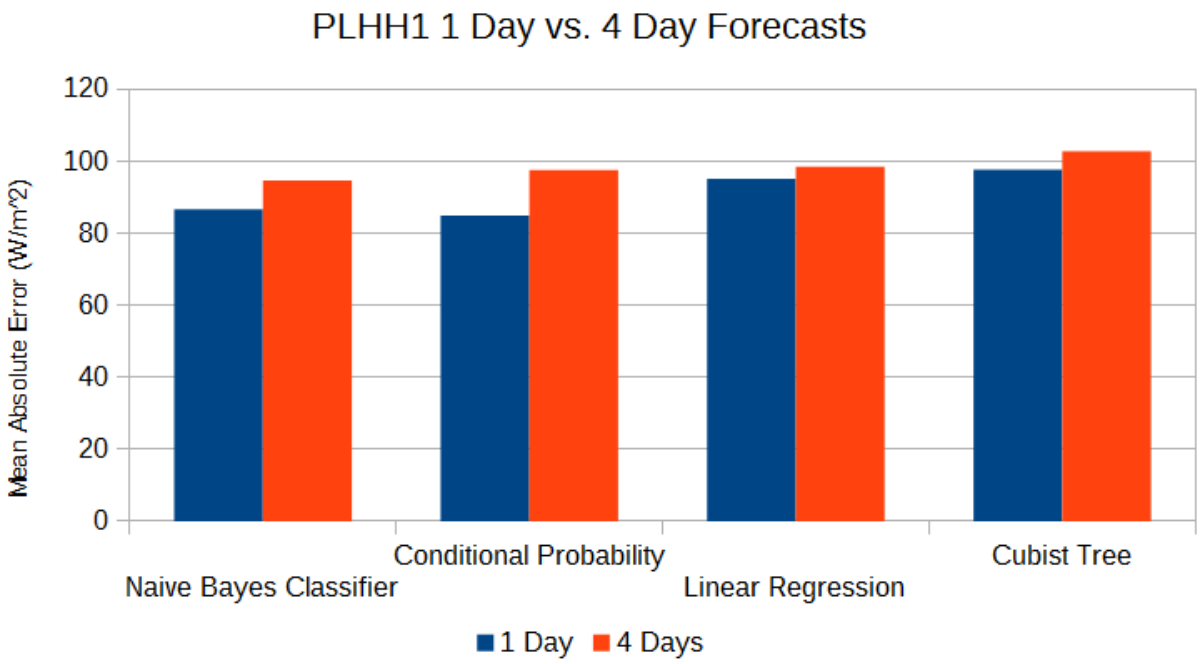
8.1 Experimental Setup

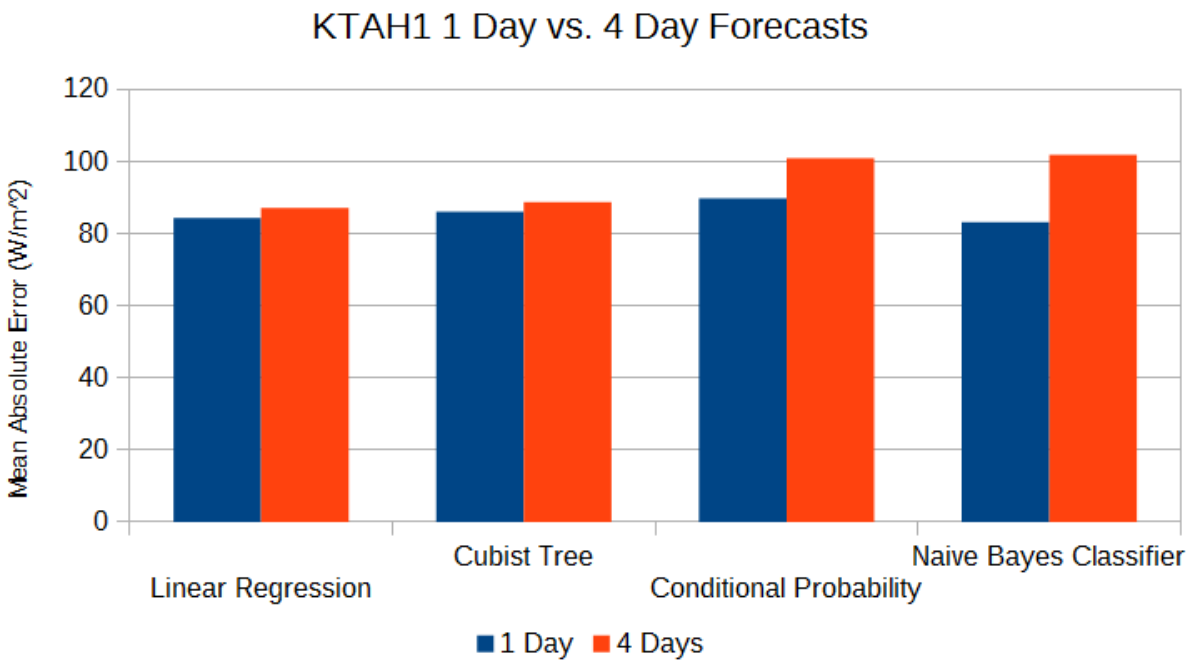
The WRF forecasts are using a lead time of 3.5 days, but our cluster prediction methods must work with forecast times and data windows that are full days. As a result, we used a lead time of 4 days for our models. For the hourly predictions (linear regression and cubist trees) we used a data window of 6 hours while we used a data window of 2 days and the set of 6 solar irradiance partitions for the cluster prediction methods (conditional probability classifier and naive Bayes classifier). The different numbers of partitions or features were tested as was done in the past experiments and the best set of partitions or features was selected for the comparison. All machine learning techniques were trained using three years of data (2010 to 2012) and tested using the last year of data (2013). The errors for the WRF forecasts are calculated using only the forecasts from 2013.

The metric used to compare the different models is the mean absolute error. For the hourly predictions and the WRF models, the error was calculated by finding the difference of the prediction at a given time and the actual observed solar irradiance value at that time. The errors for the cluster predictions are calculated by finding the differences between the centers of the predicted partition and the centers of the actual partition. In all cases, the differences are calculated for the daytime hours (hour 7 to hour 20 or 7 am to 8 pm) for consistency. Nonetheless, the mean absolute error values for the hourly predictions and WRF forecasts are not directly comparable with mean absolute error values for the cluster predictions, but they are still able to give a general idea as to how they performed in relation to each other.

8.2 Results / Analysis

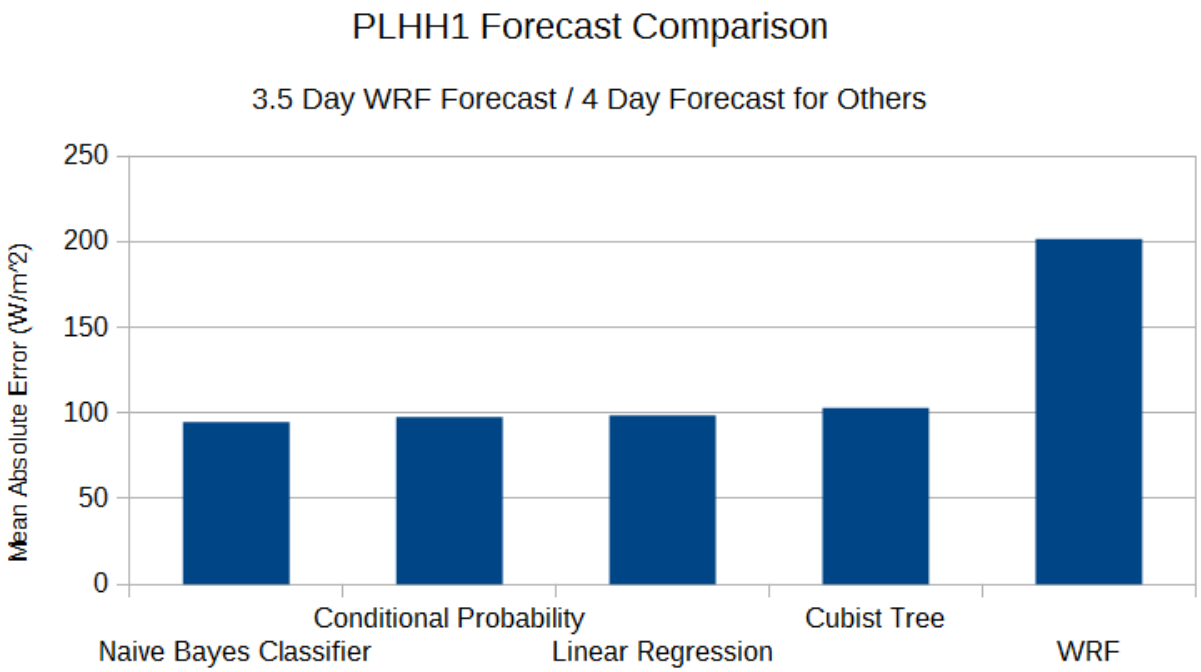
8.2.1 One Day Forecasts vs. Four Day Forecasts

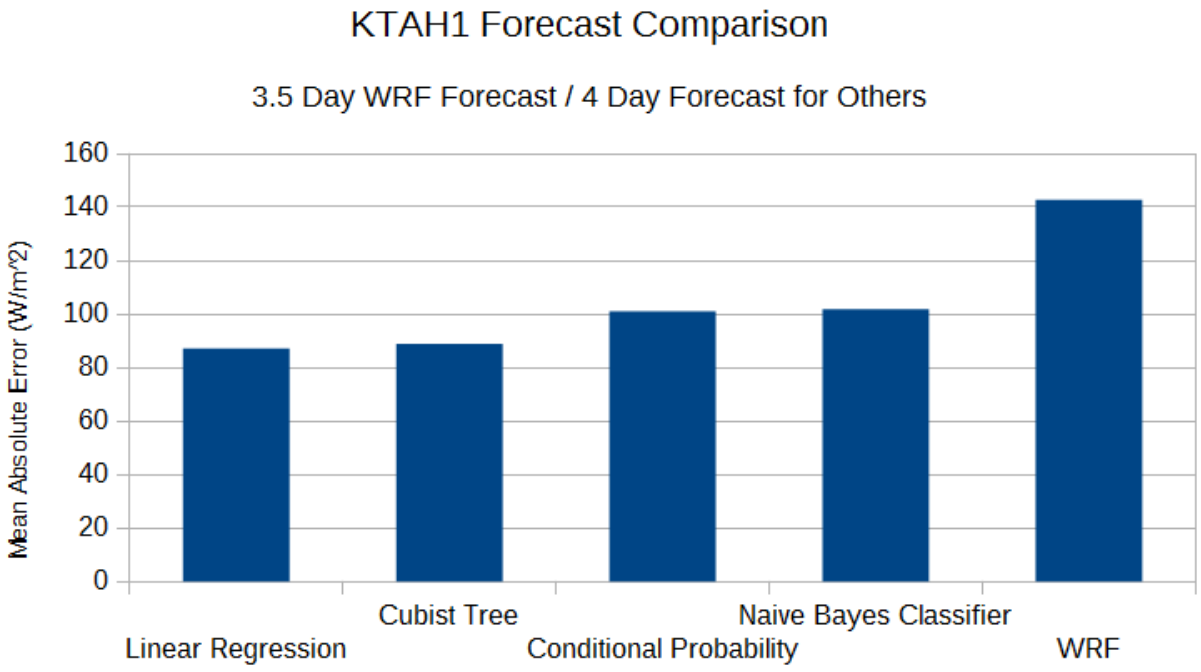




The first set of results show a comparison of the one day forecasts with the four day forecasts. The one day forecasts for the daily forecasting methods are identical to the best results from the experiments in section 6. The one day forecasts for the hourly prediction used the best setup as previously discussed, but with a 7 hour data window. In general, it the daily forecasts seem to experience a larger increase in error when the lead time is increased from one day to four days. The hourly forecasts appear to be much more stable in terms of their accuracy when the lead time is increased as the errors between the two lead times are within 5 W/m^2 of each other.

8.2.2 Comparison with WRF





The charts above show the errors resulting from the different forecasting methods. For both PLHH1 and KTAH1, the machine learning techniques performed significantly better than the WRF forecasts. At PLHH1, the naive Bayes classifier saw a reduction of approximately 106.87 W/m^2 in the mean absolute error which is about a 53% improvement. Due to how the errors are calculated, the results for the naive Bayes and

conditional probability classifiers are not directly comparable with the errors for the WRF forecasts and the observed improvement must be considered with caution. However, the best hourly prediction method, linear regression, which is directly comparable with the WRF forecasts performed almost as well with an improvement of about 103.03 W/m^2 or a 51% improvement. Our predictions at KTAH1 also performed rather well with linear regression showing a 55.53 W/m^2 reduction in the mean absolute error which is a 39% improvement. As a result, our simple machine learning techniques have proven to be quite effective with substantial improvements over the forecasts from the established WRF forecasting model.

9. Conclusion & Future Works

All in all, we were able to perform solar irradiance prediction using the sensor data that we obtained using four different simple machine learning methods and we were also able to perform various analyses on the data using standard data mining techniques to uncover interesting patterns. We have also shown the versatility of these techniques as they can be run at whichever lead time is required / most useful to the grid operators whether it be hours or days.

There are several options that can be explored to potentially further reduce the error. First, we only used the most basic machine learning techniques in our experiments and using more sophisticated techniques, such as artificial neural networks, could potentially capture the relationships better and hence lead to more accurate predictions. Furthermore, our data set was rather limited at only four years of data. Perhaps running the experiments on a larger data set would yield better results as there would be more training data for the models to learn from. Finally, the data set could be expanded to include more than just surface sensor data. One of the options that we had considered, but were unable to implement due to time constraints, was the use of GFS data to detect weather events that are approaching O‘ahu such as storms. Such weather events are difficult to detect with our current data set so knowledge of these events might improve our the predictions.

Nonetheless, our simple techniques have already proven to be rather accurate as they improved on the accuracy of the existing WRF forecasts by about 50% which makes a strong case for the use of machine learning techniques for solar irradiance forecasting tasks.

10. References

[1] Hammer, A., Heinemann, D., Lorenz, E., & Lückehe, B. (1999). Short-term forecasting of solar radiation: a statistical approach using satellite data. *Solar Energy*, *67*(1), 139-150.

[2] Lorenz, E., Remund, J., Müller, S. C., Traunmüller, W., Steinmaurer, G., Pozo, D., ... & Guerrero, C. G. (2009, September). Benchmarking of different approaches to forecast solar irradiance. In *Proceedings of the 24th European Photovoltaic Solar Energy Conference* (pp. 4199-4208).

[3] Reikard, G. (2009). Predicting solar radiation at high resolutions: A comparison of time series forecasts. *Solar Energy*, *83*(3), 342-349.

[4] Bacher, P., Madsen, H., & Nielsen, H. A. (2009). Online short-term solar power forecasting. *Solar Energy*, *83*(10), 1772-1783.

[5] Marquez, R., & Coimbra, C. F. (2011). Forecasting of global and direct solar irradiance using stochastic learning methods, ground experiments and the NWS database. *Solar Energy*, *85*(5), 746-756.

[6] Wang, F., Mi, Z., Su, S., & Zhao, H. (2012). Short-term solar irradiance forecasting model based on artificial neural network using statistical feature parameters. *Energies*, *5*(5), 1355-1370.

[7] Mellit, A., Benghanem, M., & Kalogirou, S. A. (2006). An adaptive wavelet-network model for forecasting daily total solar-radiation. *Applied Energy*, *83*(7), 705-722.

[8] Cao, J., & Lin, X. (2008). Study of hourly and daily solar irradiation forecast using diagonal recurrent wavelet neural networks. *Energy Conversion and Management*, *49*(6), 1396-1406.

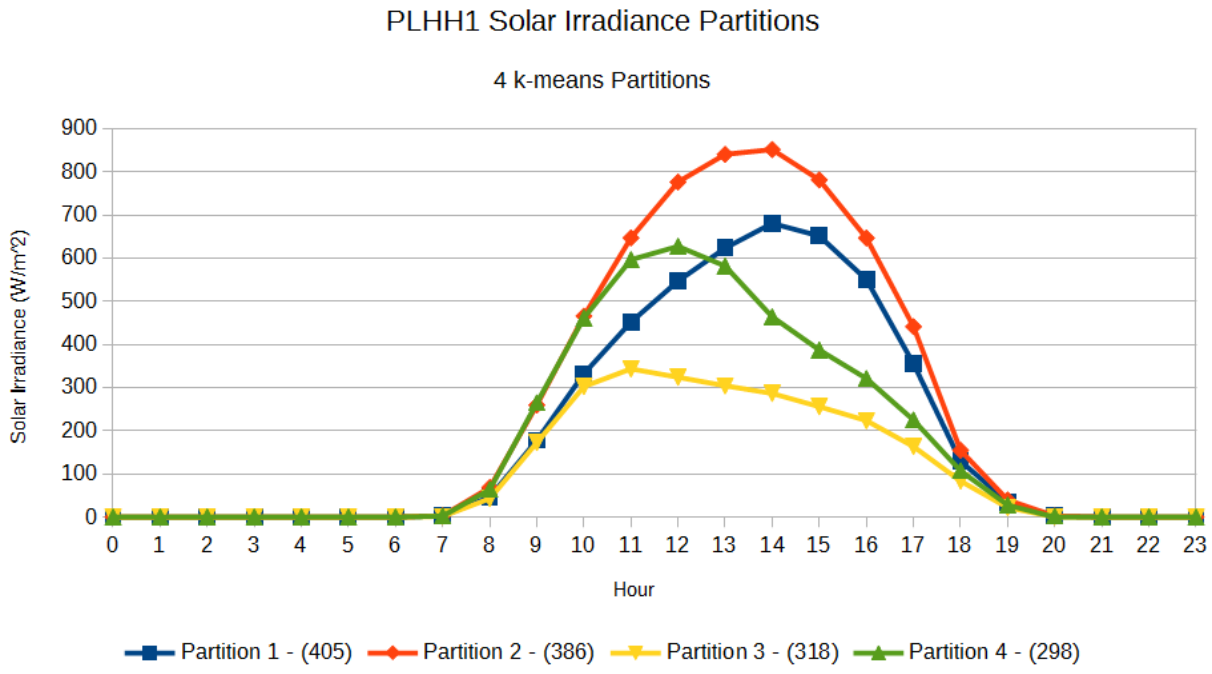
[9] Sfetsos, A., & Coonick, A. H. (2000). Univariate and multivariate forecasting of hourly solar radiation with artificial intelligence techniques. *Solar Energy*, *68*(2), 169-178.

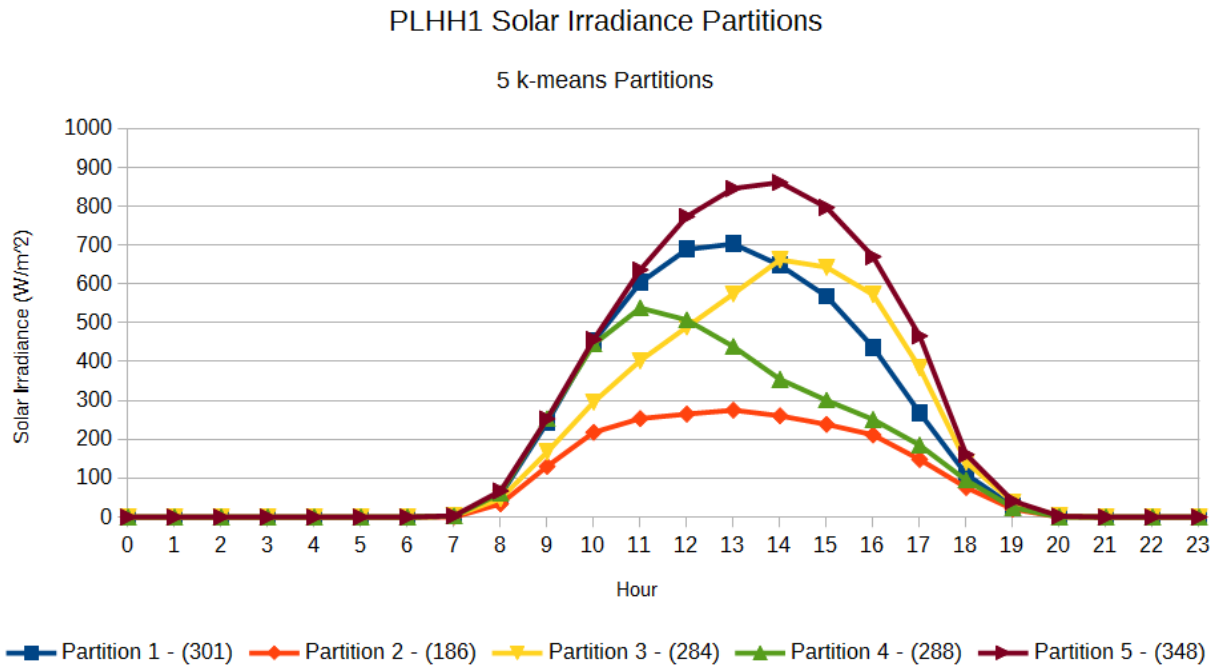
[10] Martín, L., Zarzalejo, L. F., Polo, J., Navarro, A., Marchante, R., & Cony, M. (2010). Prediction of global solar irradiance based on time series analysis: application to solar thermal power plants energy production planning. *Solar Energy*, *84*(10), 1772-1781.

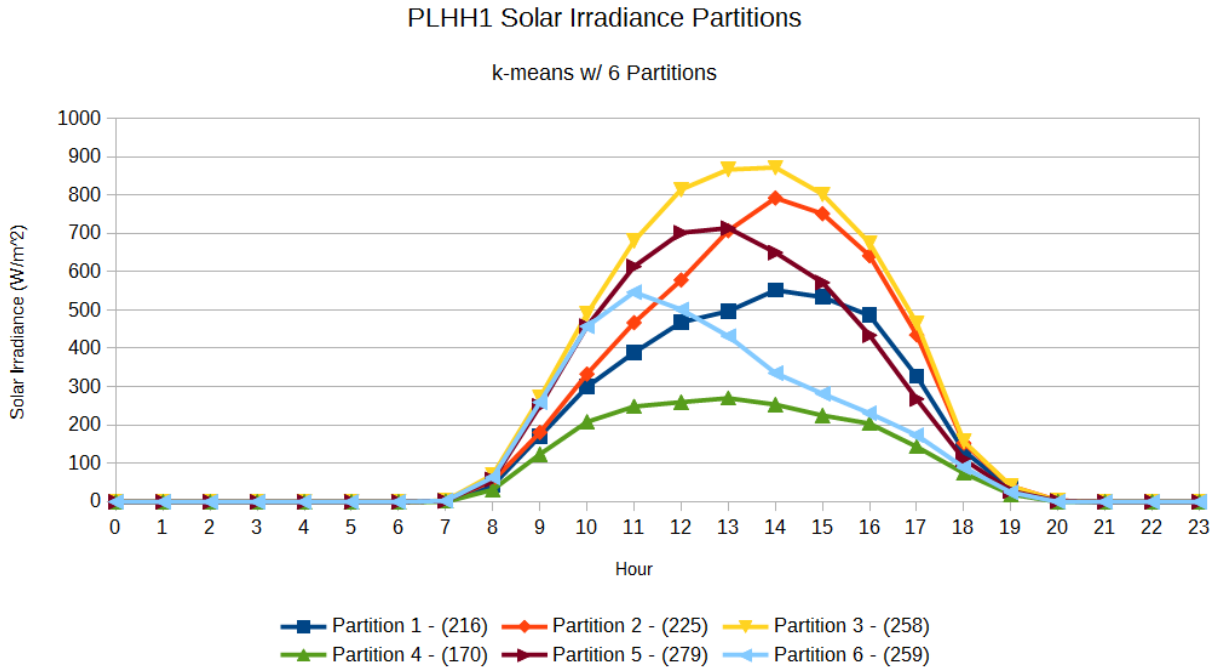
[11] Mellit, A., & Pavan, A. M. (2010). A 24-h forecast of solar irradiance using artificial neural network: Application for performance prediction of a grid-connected PV plant at Trieste, Italy. *Solar Energy*, *84*(5), 807-821.

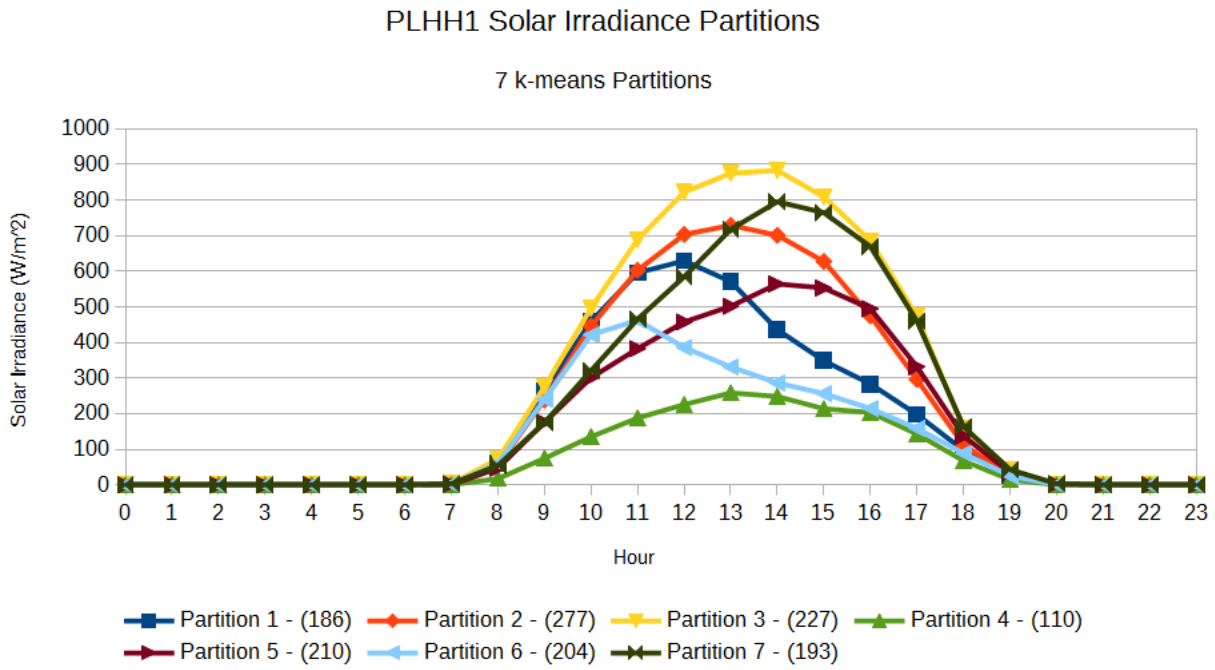
A. Appendix

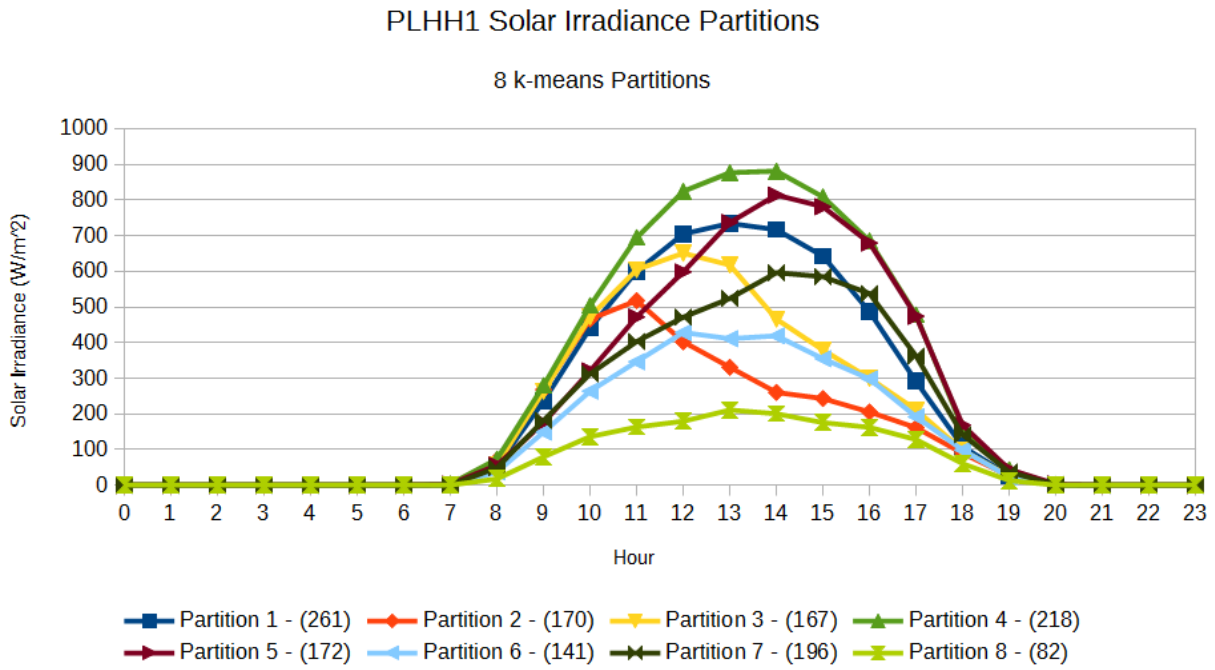
A.1 PLHH1 Solar Irradiance Partitions

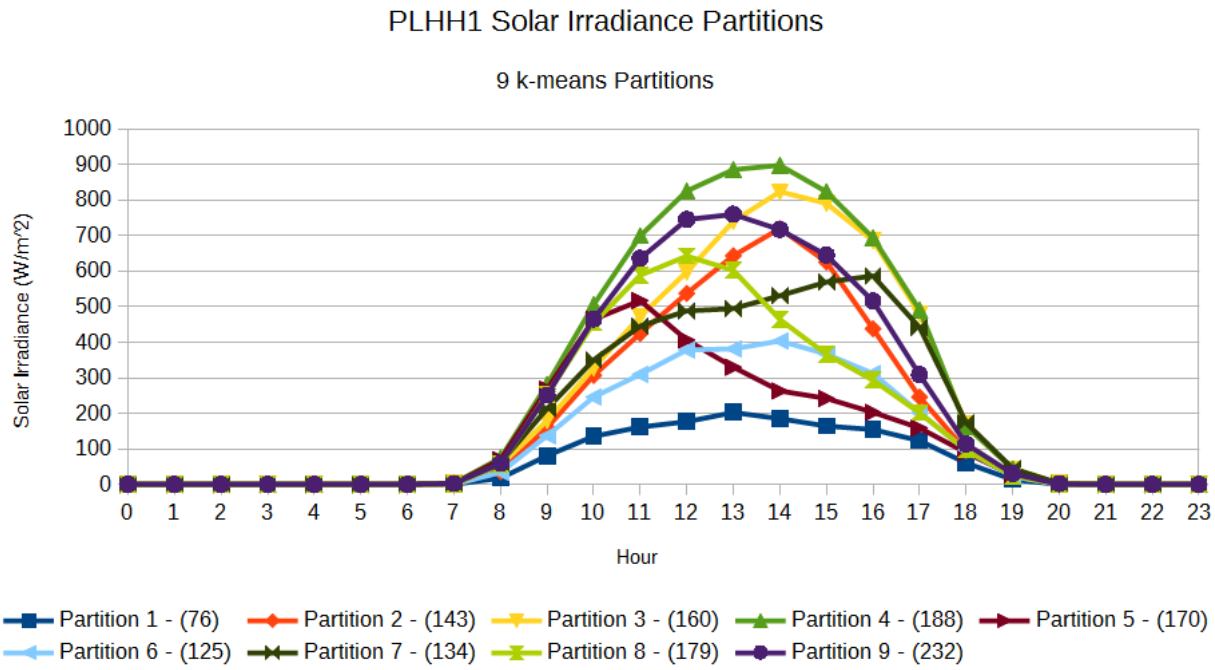


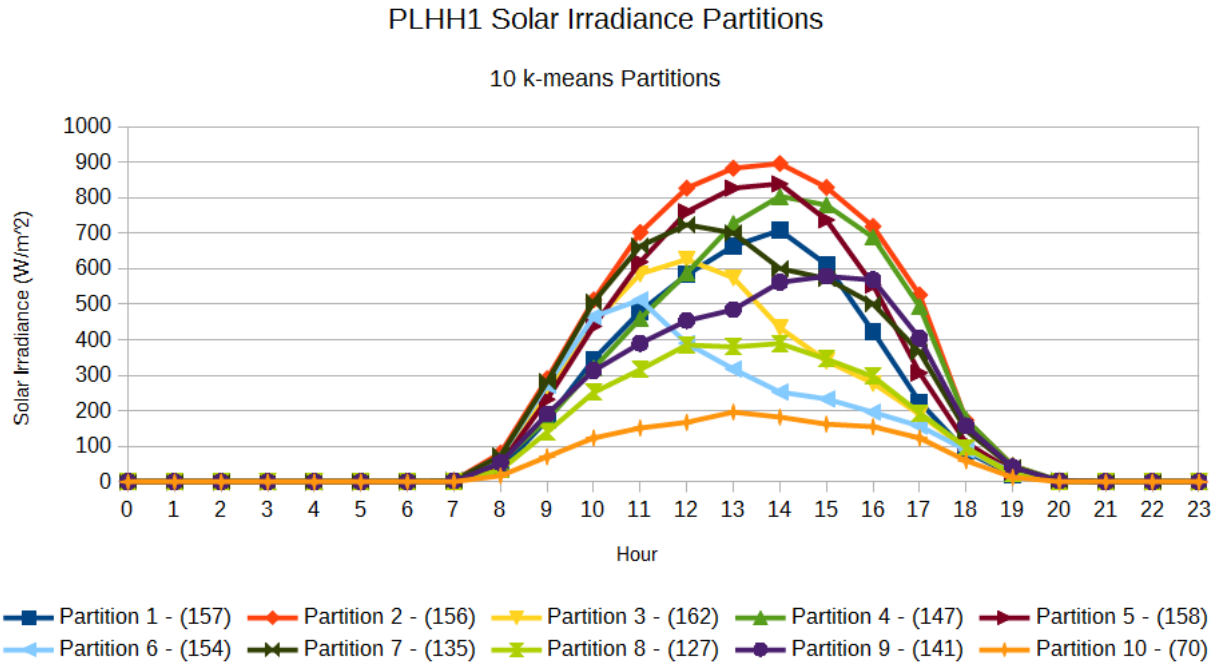

PLHH1 Solar Irradiance Partitions
5 k-means Partitions
Solar Irradiance (W/m^2)
1000
900
800
700
600
500
400
300
200
100
0
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
Hour
Partition 1 - (301)
Partition 2 - (186)
Partition 3 - (284)
Partition 4 - (288)
Partition 5 - (348)


PLHH1 Solar Irradiance Partitions
k-means w/ 6 Partitions
Solar Irradiance (W/m^2)
1000
900
800
700
600
500
400
300
200
100
0
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
Hour
Partition 1 - (216)
Partition 2 - (225)
Partition 3 - (258)
Partition 4 - (170)
Partition 5 - (279)
Partition 6 - (259)

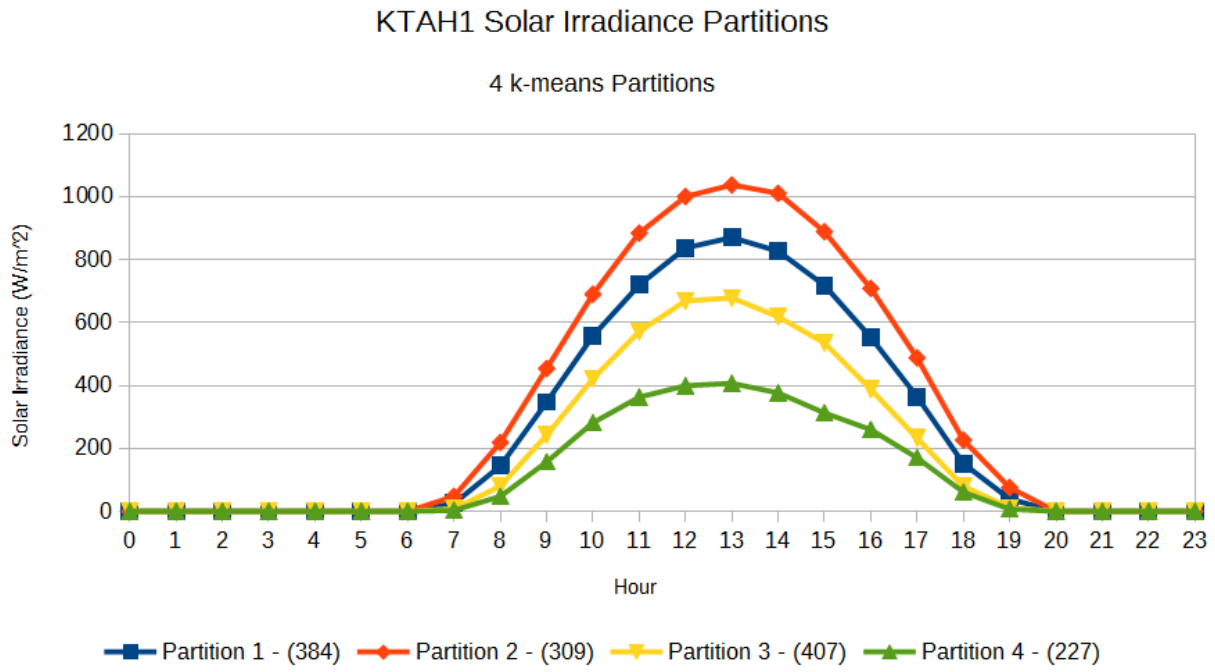

PLHH1 Solar Irradiance Partitions
7 k-means Partitions
Solar Irradiance (W/m^2)
Hour
Partition 1 - (186)
Partition 2 - (277)
Partition 3 - (227)
Partition 4 - (110)
Partition 5 - (210)
Partition 6 - (204)
Partition 7 - (193)

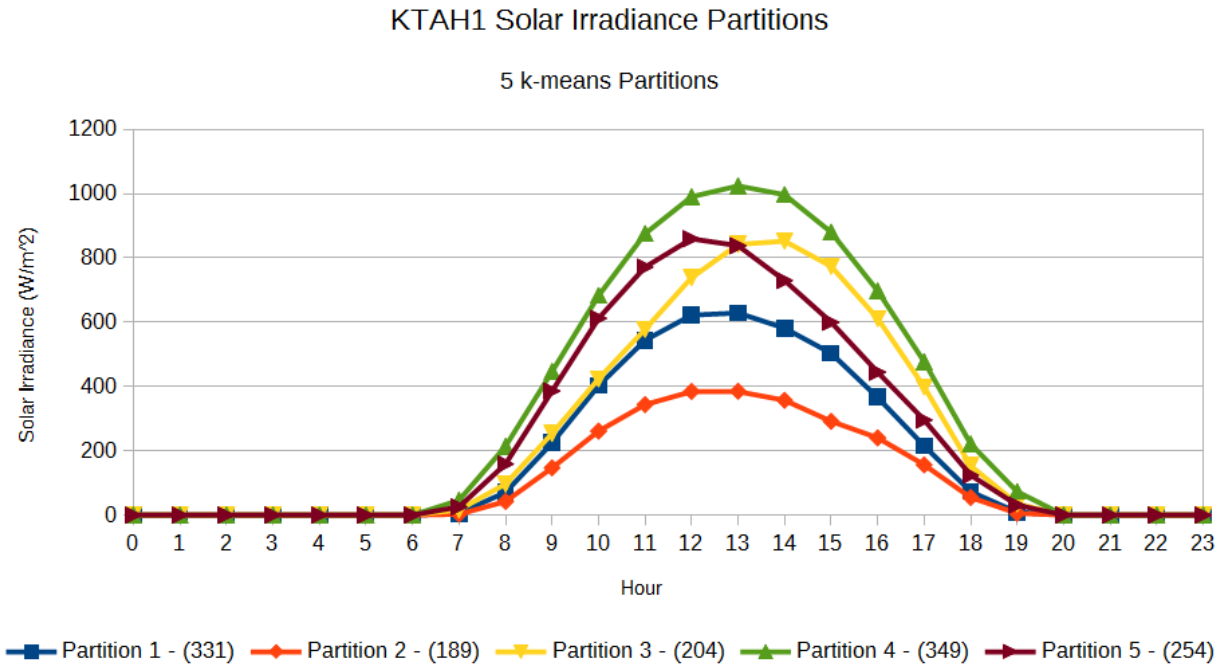

PLHH1 Solar Irradiance Partitions
8 k-means Partitions
Solar Irradiance (W/m^2)
Hour
Partition 1 - (261)
Partition 2 - (170)
Partition 3 - (167)
Partition 4 - (218)
Partition 5 - (172)
Partition 6 - (141)
Partition 7 - (196)
Partition 8 - (82)

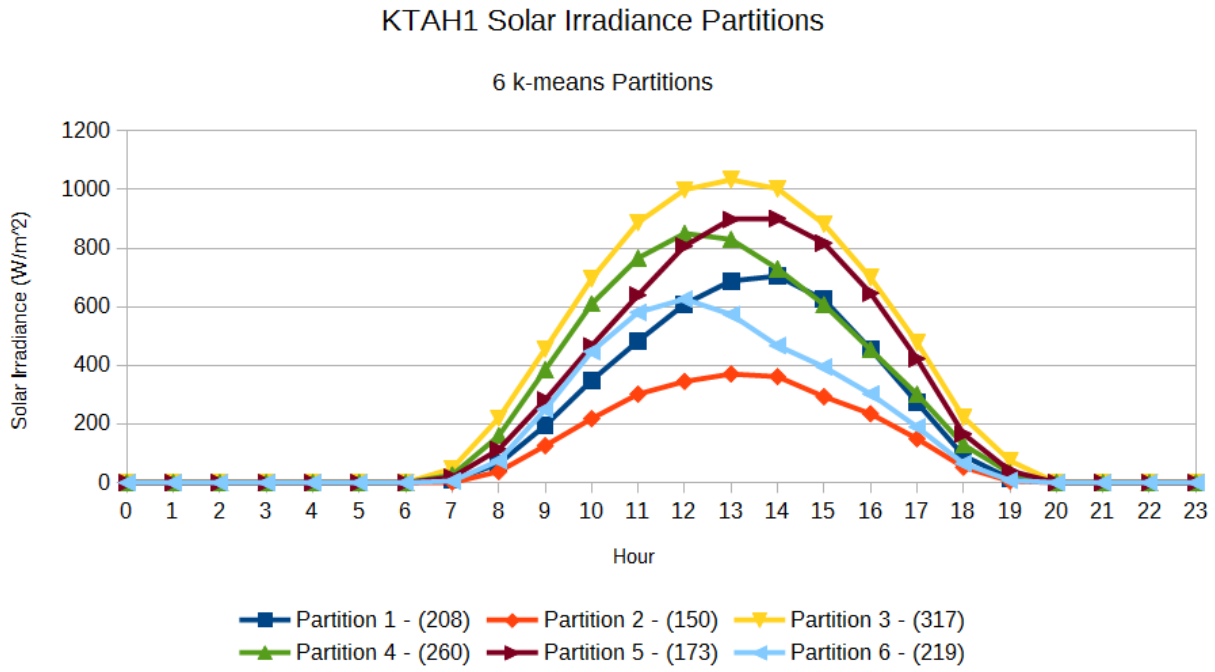


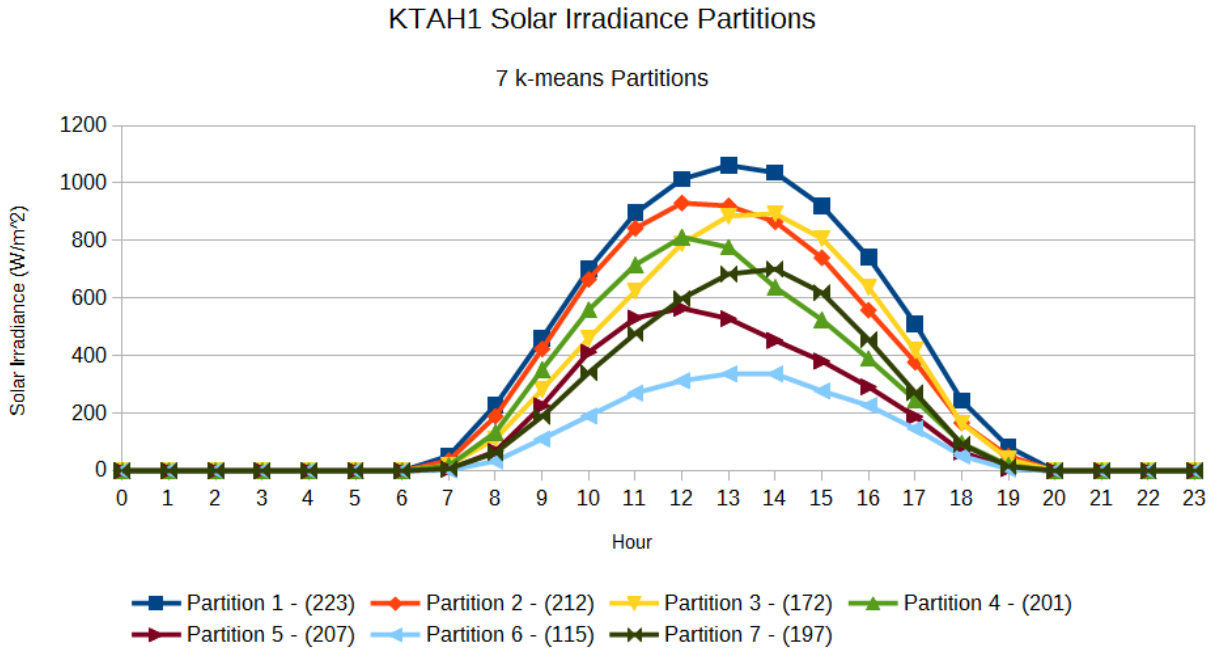


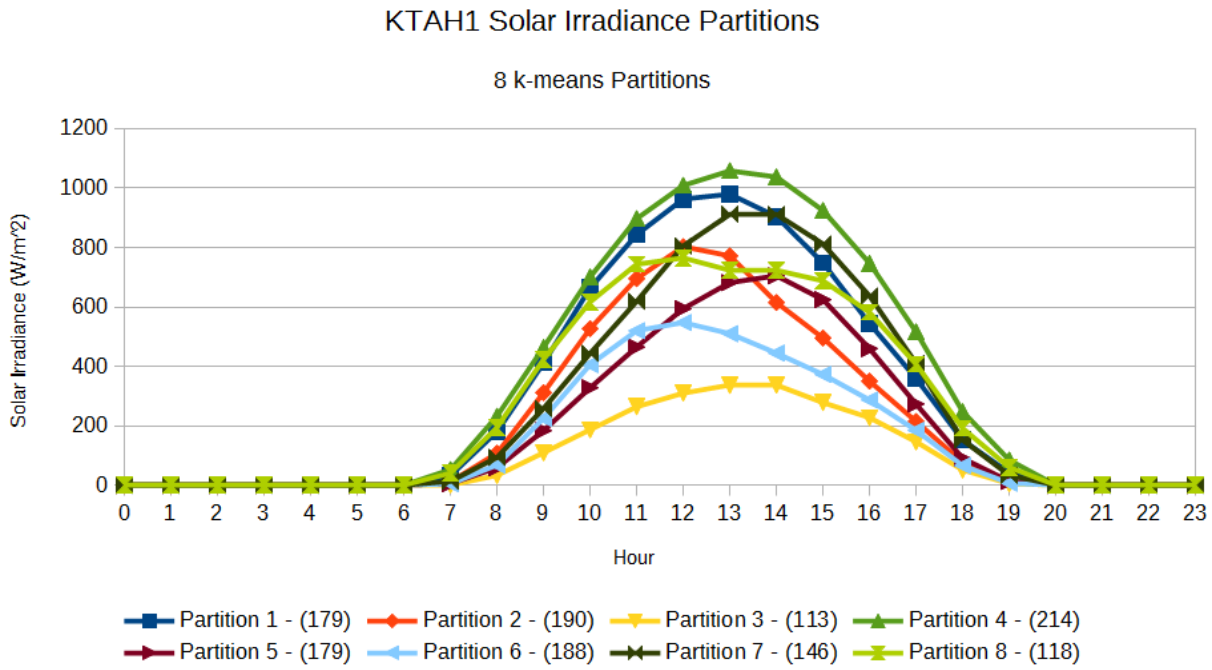
A.2 KTAH1 Solar Irradiance Partitions

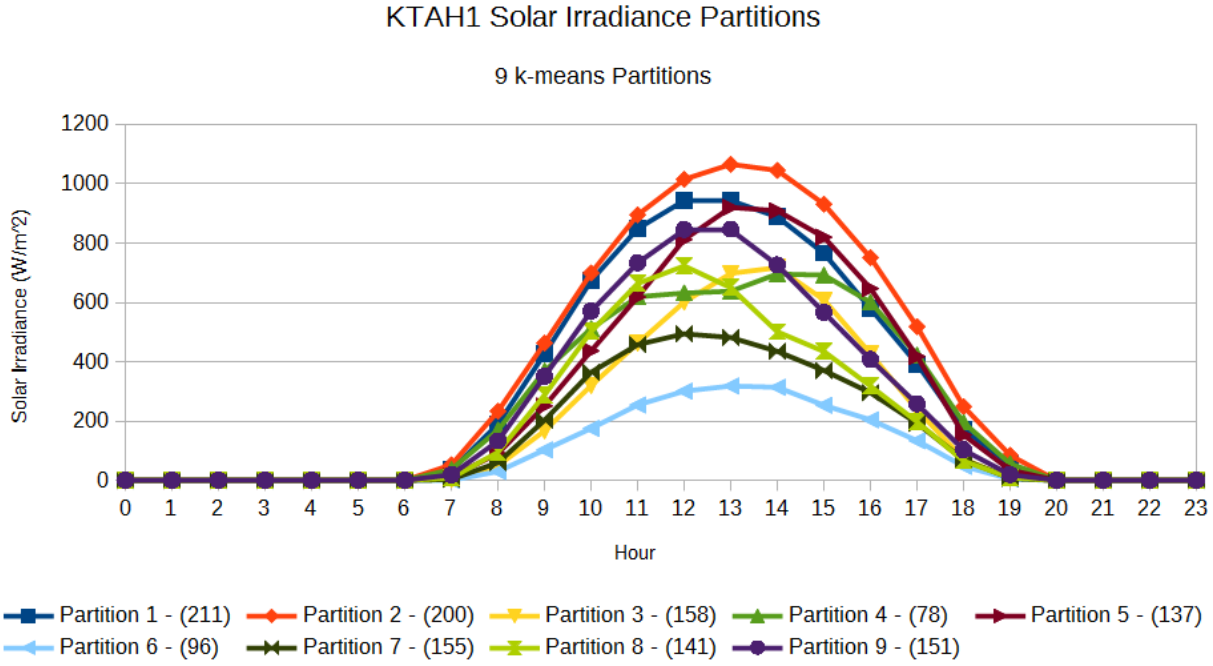

KTAH1 Solar Irradiance Partitions
4 k-means Partitions
Solar Irradiance (W/m^2)
1200
1000
800
600
400
200
0
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
Hour
Partition 1 - (384)
Partition 2 - (309)
Partition 3 - (407)
Partition 4 - (227)

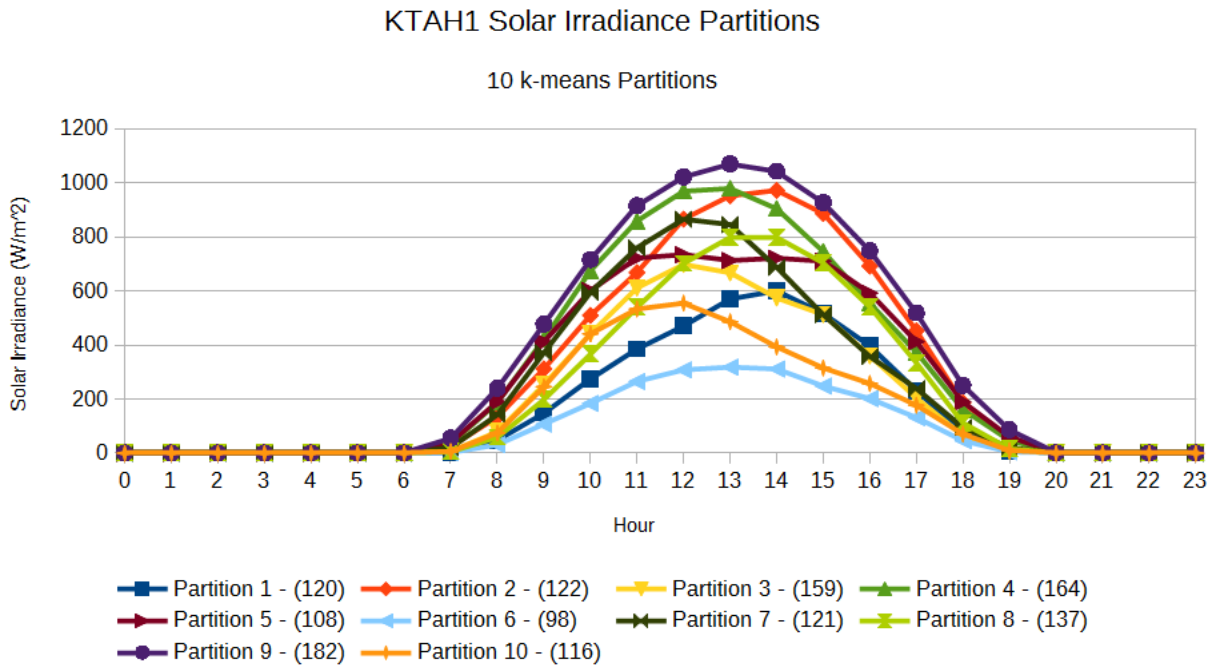

KTAH1 Solar Irradiance Partitions
5 k-means Partitions
Solar Irradiance (W/m^2)
1200
1000
800
600
400
200
0
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
Hour
Partition 1 - (331)
Partition 2 - (189)
Partition 3 - (204)
Partition 4 - (349)
Partition 5 - (254)


KTAH1 Solar Irradiance Partitions
6 k-means Partitions
Solar Irradiance (W/m^2)
Hour
Partition 1 - (208)
Partition 2 - (150)
Partition 3 - (317)
Partition 4 - (260)
Partition 5 - (173)
Partition 6 - (219)


KTAH1 Solar Irradiance Partitions
7 k-means Partitions
Solar Irradiance (W/m^2)
Hour
Partition 1 - (223)
Partition 2 - (212)
Partition 3 - (172)
Partition 4 - (201)
Partition 5 - (207)
Partition 6 - (115)
Partition 7 - (197)


KTAH1 Solar Irradiance Partitions
8 k-means Partitions
Solar Irradiance (W/m^2)
Hour
Partition 1 - (179)
Partition 2 - (190)
Partition 3 - (113)
Partition 4 - (214)
Partition 5 - (179)
Partition 6 - (188)
Partition 7 - (146)
Partition 8 - (118)


KTAH1 Solar Irradiance Partitions
9 k-means Partitions
Solar Irradiance (W/m^2)
Hour
Partition 1 - (211)
Partition 2 - (200)
Partition 3 - (158)
Partition 4 - (78)
Partition 5 - (137)
Partition 6 - (96)
Partition 7 - (155)
Partition 8 - (141)
Partition 9 - (151)


KTAH1 Solar Irradiance Partitions
10 k-means Partitions
Solar Irradiance (W/m^2)
1200
1000
800
600
400
200
0
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
Hour
Partition 1 - (120)
Partition 2 - (122)
Partition 3 - (159)
Partition 4 - (164)
Partition 5 - (108)
Partition 6 - (98)
Partition 7 - (121)
Partition 8 - (137)
Partition 9 - (182)
Partition 10 - (116)