Classifying Building Energy Consumption Behavior Using an Ensemble of Machine Learning Methods

Kunal Sharma, Nov 26th 2018

Technical adviser: Dr. Jung-Ho Lewe
Academic adviser: Dr. Dimitri N. Mavris

Areospace Design Lab

Georgia Institute of Technology

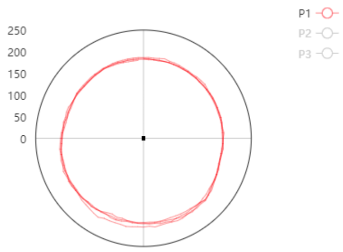
Objective

The goal of this project is to detect outlier cases of energy usage of the 137 buildings on the Georgia Tech campus using energy consumption data. These outlier cases often represent energy mismanagement, overconsumption, or electric issues. Current literature on energy consumption data focuses on the usage of regression analysis for predicting energy usage. Decision trees and Artificial neural networks have been used to predict the energy consumption of Hong Kong's energy usage in the domestic sector from 1971 to 2001 [1]. Other attempts have also used Support Vector Machines to forecast regional annual electric load in Taiwan using data from 1981 to 2000 [2]. While this project will also be working with energy consumption data, the focus will be on defining and classifying certain common behavior types.

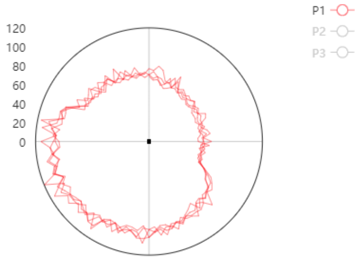
Methodology

Our main approach to detecting if a building's behavior has changed is to first be able to classify a building's behavior based on its energy usage or “fingerprint”. The fingerprint is a polar plot of the energy usage of a building over 24 hours. The data is collected and plotted in 15 min intervals. Notable differences in the energy signatures allow us to classify a building's consumption behavior into 5 types: Concentric, People, Scheduler, Reverse, and Random.

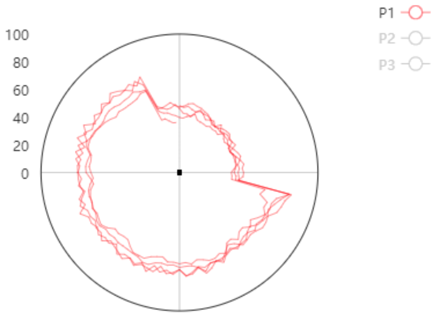
1. Concentric: The building's energy consumption is relatively constant throughout the day.



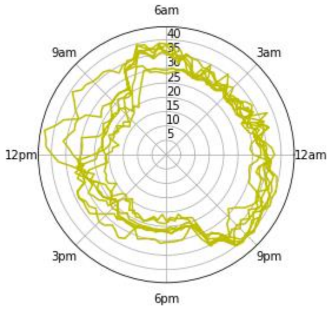
2. People: The building uses more energy during work hours likely due to energy consumption by people (lights, air-conditioning, heating, etc.).



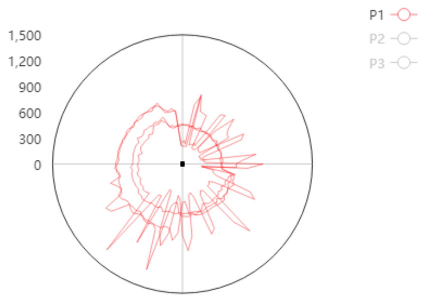
3. Scheduler: Building that operates with a scheduler machine. The plot typically has sudden jumps and drops in energy usage at set time periods.



4. Reverse is when the energy usage during non-work hours is much greater than the energy usage for the rest of the day.



5. Random is the final catch all term, of building energy behavior that is abnormal and cannot be classified into the categories previously mentioned.



The data that we are using to build this classifier comes from a server that has been recording 15-min intervals of energy usage for more than 3 years. Our analysis will focus on energy usage only during the summer time.

This project was developed using Python (version 3.5) and Jupyter notebooks. Several machine learning libraries were used including Pandas, Numpy, and Scikit-Learn.

Machine learning algorithms perform well on pattern recognition (classification) problems given sufficient examples. In order to classify the energy usage, a number of different machine learning models were selected and tested. 2,926 energy signatures were manually labelled for training purposes. Ultimately, a handful of the best performing algorithms were chosen and fed into an ensemble classifier which saw the best results in the classification.

Machine Learning Algorithms Attempted:

- Decision Trees
- Random Forests
- K Nearest Neighbors
- Support Vector Machine
- Naïve Bayes
- Logistic Regression
- Artificial Neural Network

The algorithms that consistently had the highest accuracy were Random Forests, K Nearest Neighbors and Artificial Neural Networks. These models we're used together to create an ensemble classifier with a higher accuracy than any one algorithm.

In order to increase the size of the dataset, artificial data generation and data manipulation was attempted. Defining the classification prototypes was an iterative process of creating the classifiers and analyzing misclassified cases.

Since buildings with very little energy consumption have very volatile energy usage, such buildings were filtered out of the dataset at different threshold values.

Lessons Learned and Takeaways

Labelling data is a time consuming process. In an attempt to reduce the time needed to complete the project, we wondered whether random generators could be created to generate artificial data that would still be able to effectively train the classifiers.

The random forest classifier achieved a 98.7% accuracy in classifying artificial data but only achieved a mere 7% accuracy when classifying real instances of classes. This is likely because while the model can classify the programmed variation in the randomly generated plots, said plots do not accurately represent the true variation in the data.

Since randomly generating data did not work, we turned to applying data transformations on our existing labelled data set to multiply the amount of labelled data that we had available to train our models.

We multiplied the dataset size by “rotating” or changing the time of each consumption record by some multiple of 15 mins to between 2 hours before to 2 hours after the time of the original data. This method improved the accuracy by about 2%.

To create even more data we tried multiplying every 15 minute data point by a small number to create a slightly different energy signature. However, this decreased the accuracy by 3% likely because it only added noise to the data and distorted the nuances in the data that were key to classification.

It was difficult to evaluate the performance of the models because the data was heavily unbalanced. The overwhelming majority of samples were “Concentric” or “People”.

The classifiers had the highest error rate on cases that had characteristics of two or more classification types. The majority of these cases turned out to be buildings with extremely small energy usage. We decided to classify these cases separately.

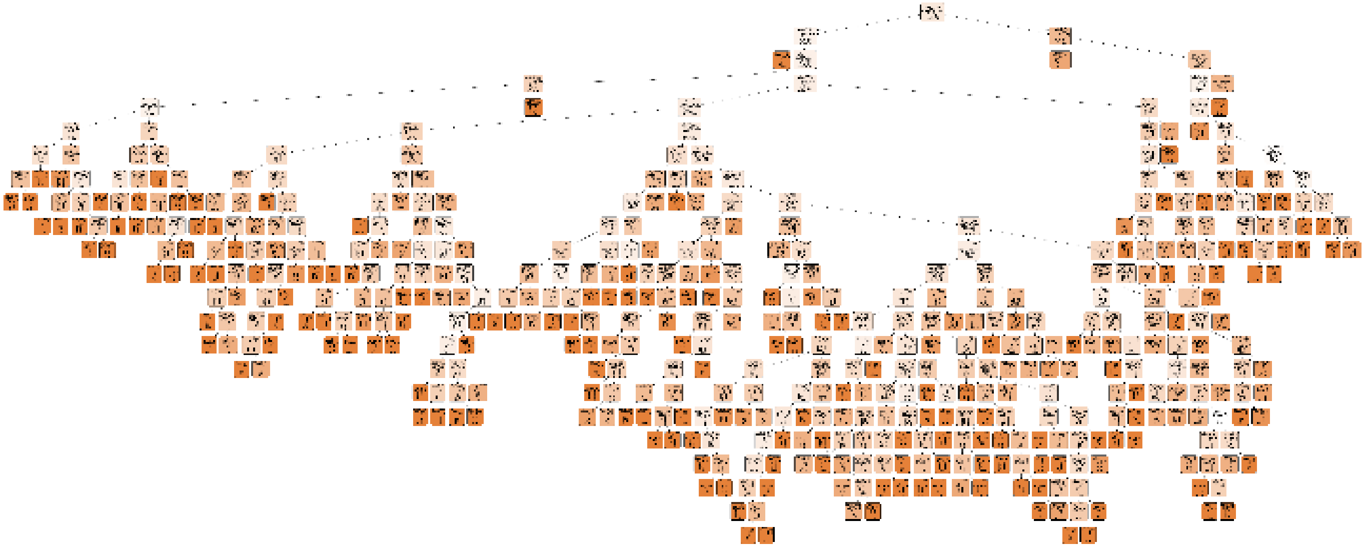
We also faced a tradeoff between labelling the data quickly and labelling the data accurately. Often times the graph may look like one energy class but when the magnitude of the data is looked at more closely and not just the shape then a different classification is given. To handle mislabeled cases we created functions that could search and update records as well as print out the misclassified instances.

Labelling two types of behavior as one class decreased the accuracy of our model significantly. The class “Reverse” was originally defined as an energy signature with a sudden strange drop in consumption or significantly greater consumption during off-hours. After, we changed our definition to the latter and relabeled accordingly, the accuracy increased 1-2%. We found that the process of redefining classification prototypes to handle grey misclassified cases as avoidable with better initial data exploration.

Classification Results

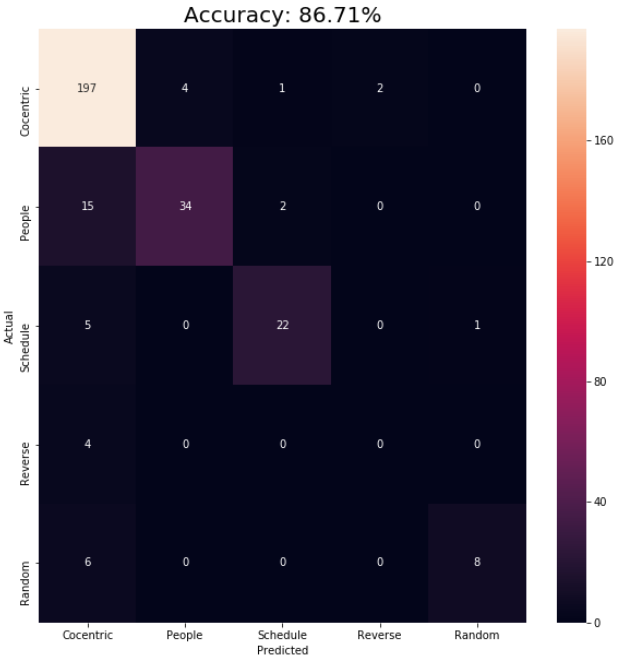
Of the seven classification algorithms, the three that consistently performed the best were: Random Forests, K-Nearest Neighbors, and Artificial Neural Network. The models were all used to build a voting ensemble classifier that achieved a classification accuracy higher than any one of the algorithms.

The random forest algorithm uses a collection of decision trees that “vote” on the classification of the data instance. A decision tree is a model that comes to a classification decision after splitting the data on multiple attributes and thresholds. A visualization of a decision tree classifying a building’s consumption behavior can be seen below.

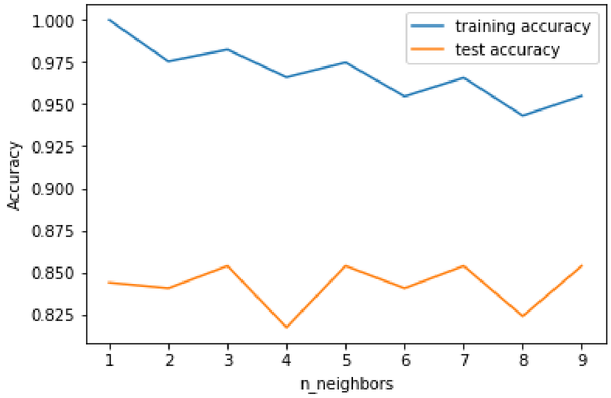


Decision Tree Classifier Visualization

The Random Forest’s accuracy is 86.71%. As can be seen from the confusion matrix below, the most difficult area for the model to classify is between “concentric” and “people” this is because there are several cases in the data that are a close mix between the two. To mitigate this issue, we went back through the data and only classified cases that were obviously “People” as so and otherwise left the classification as concentric. This consistency helped improved the accuracy.

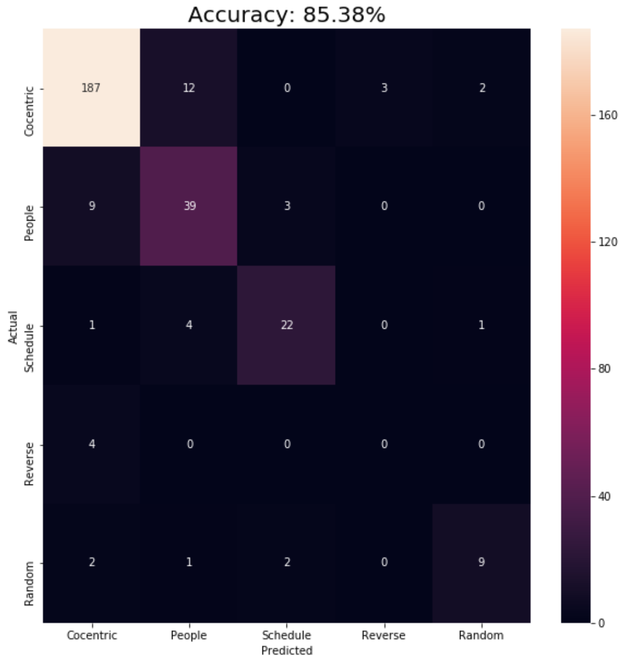


The K-Nearest Neighbor algorithm looks at K number of data points closest to the data point to be classified. The final classification is the majority class of the K nearest data points. To find the optimal performance we can test different values for K. For example:

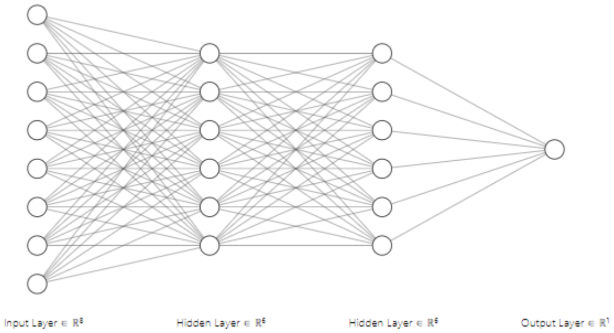


Here we can see that the optimal K value is 5 with an accuracy of about 86%. The rotation of data to increase the size of the dataset has a direct impact on the KNN algorithm as it gives it more points of the same classification in a very close proximity (only 15 min apart).

The K-Nearest Neighbor algorithm achieved an accuracy of 85.38%, slightly less than that of Random Forest.

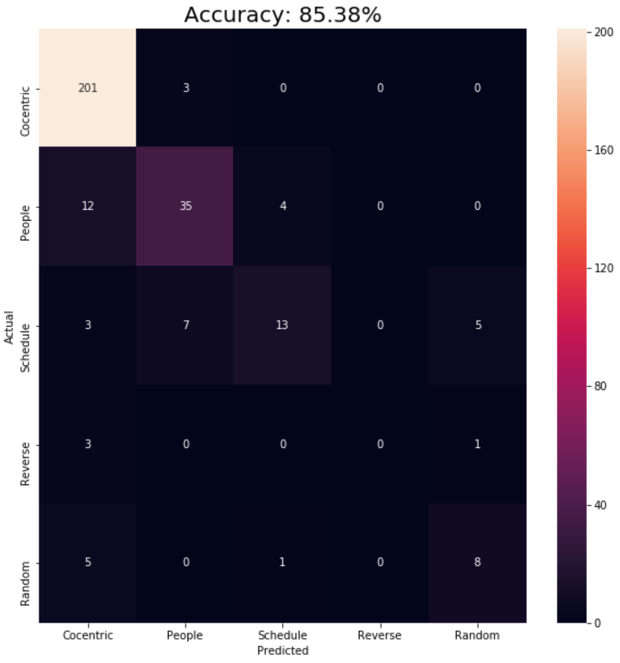


The Artificial Neural Network is network of nodes and weights that forward propagate an input and output a classification. Error is backpropagated through the network and the weights are adjusted to minimize the error of the network.

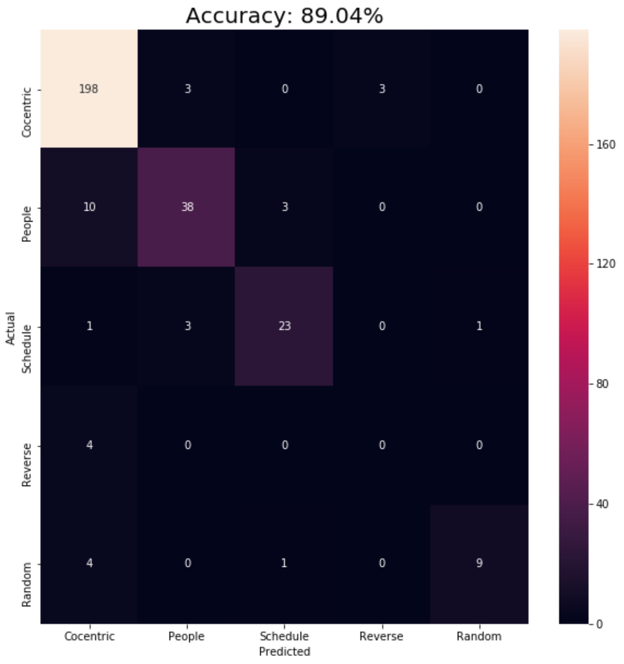


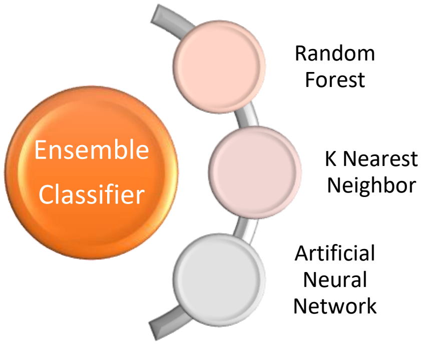
After several trails, the best performing neural network architecture for this problem seems to be [96, 30, 20, 10, 8, 5]. Where 96 is the number of nodes in the input layer (one for every 15 min), 5 is the number of nodes in the output layer (number of classifications), and the numbers in between represent the number of nodes in the four hidden layers.

The artificial neural network achieved an accuracy of 85.38%, the same performance as the KNN algorithm.



The final ensemble classifier is a voting machine that includes the random forest model, the k nearest neighbor model and the artificial neural network. By having all of the algorithms bias in each of the algorithms is reduced and as a result the overall accuracy increases. The ensemble classifier achieved an accuracy of 89.04%.

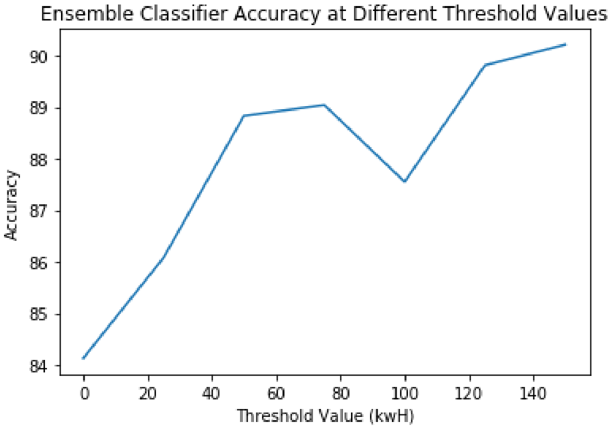




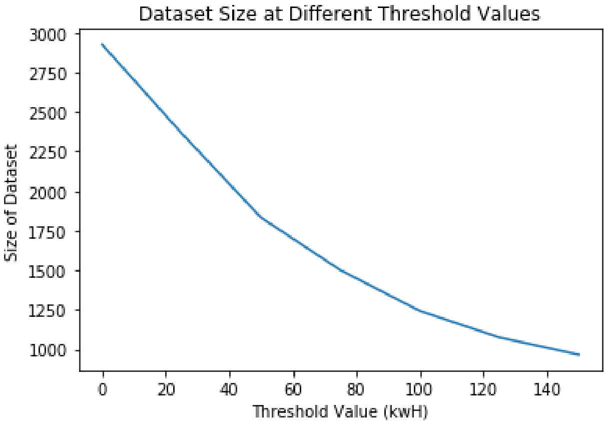
Below is a summary of the classifier results on data that had a higher average energy consumption of 75kwH.

Algorithm	Accuracy
Ensemble Classifier	**89.04%**
Random Forest	86.71%
KNN	85.38%
ANN	85.38%

As expected, the ensemble classifier has the best performance at 89.04% accuracy. Below is a chart with the performance of the ensemble classifier when the data is filtered at different thresholds.



We can immediately observe that as the threshold value increases, the accuracy increases as well. At a threshold of 150kwH the accuracy achieved is 90.21%. As expected the size of the dataset decreases as the threshold increases.



Conclusion

In conclusion, machine learning is an effective approach for classifying time series data. The highest accuracy was achieved by using an ensemble classifier which included a random forest model, K-Nearest Neighbor model, and an Artificial Neural Network Model. If this project was repeated then we would spend more time on data exploration in order to obtain a clear definition of classification prototypes.

There are a number of strategies that can be implemented to improve on this project. Grid search can be used for parameter optimization. More specific classifiers can be developed to classify low-consumption cases that we previously filtered out. Statistical techniques can be applied to identify consumption anomalies. Creating regression models to predict building energy consumption.

References

1. Tso, Geoffry KF. “Predicting Electricity Energy Consumption: A Comparison of Regression Analysis, Decision Tree and Neural Networks.” ELS-CDN, 2005, c.els-cdn.com/S0360544206003288/1-s2.0-S0360544206003288-main.pdf?_tid=55f12b7c-1f7f-40f4-9554-7d3ee7d69e5a&acdnat=1543608070_f1e9d5f6860e5cfbb6edb99e48ab8804.

2. Deb, Chirag. “A Review on Time Series Forecasting Techniques for Building Energy Consumption.” Elsevier, Pergamon, 11 Mar. 2017, reader.elsevier.com/reader/sd/pii/S1364032117303155?token=54885EB8DFF9689CCE1517FD47DCC5E8A5731380017AE656F2F3669695188044FD73803D29336244F8E4E789C0073914