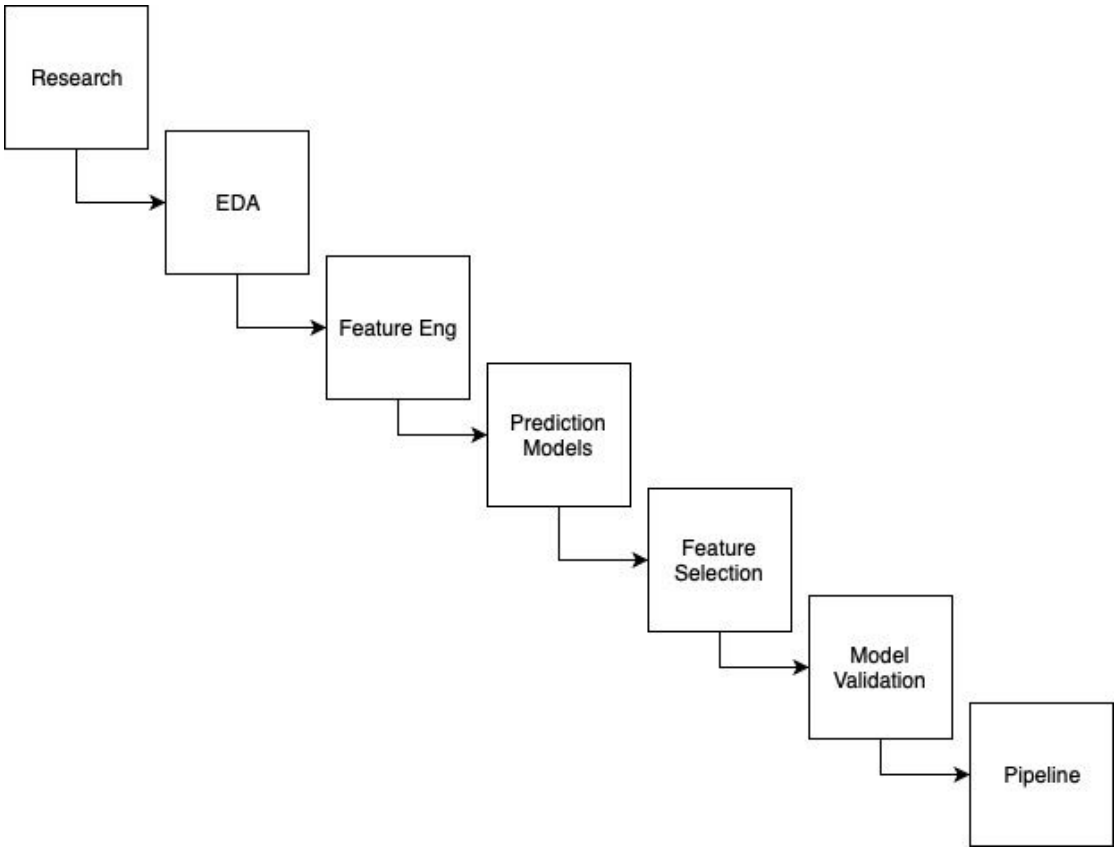
Assignment 2: ML with Energy Dataset

Rupesh Acharya, Niyati Maheshwari, Peter Vayda

November 4, 2018

Summary:

The goal of this assignment is to utilize data from IoT sensors and weather data to predict residential appliance electrical load. To do this, we utilized a dataset from a collection of local sensors and weather environment data for a house in Stambruges, Belgium.



Part 1: Research

To start our journey into model development, we reviewed the assigned research papers. Paper A “Data driven prediction models of energy use of appliances in a low-energy house,” looked into developing a appliance electrical load prediction model based on interior and exterior environment data similar to our assignment. The author tested selecting features with Boruta and recursive feature elimination (RFE) and tested models in multiple linear regression, support vector machine with radial basis function kernel, random forest, and gradient boosting machines (GBM).

Paper B “A review of artificial intelligence based building energy use prediction: Contrasting the capabilities of single and ensemble prediction models,” looked into AI-based approaches for building energy use prediction. Specifically, how combining single approaches (linear regression, neural networks, etc) into ensemble (multi-AI) approaches can improve prediction accuracy.

Paper C “Prediction of appliances energy use in smart homes,” looked into predicting appliance to enable prediction of total energy grid load. The paper detailed the use of a stochastic model and the application of predictors to solve the problem.

Part 2: Exploratory Data Analysis

An exploratory data analysis (EDA) was conducted to understand the housing electrical load dataset. Our data was from a variety of local sensors observing temperature and relative humidity, electrical load meters, and outdoor weather data from the nearby airport. By characterizing our dataset, we were able to understand the various data sources available and potential correlation to each other and to our target (appliance electrical load). Overall we have a date timestamp and a variety of numerical data-types. We need to break the timestamp into usable components (features) that can be used by our model. From life experience, it is apparent that appliance electrical load is a direct result of human interaction when they are home. This makes the time of day a very important factor to determining appliance electrical load. For example when people are at work, there will be less appliances used and less electrical load required.

Through our EDA, we assessed the quality and availability of the data. Our dataset was quite complete with no missing values throughout the timespan. The quality of the dataset is acceptable with specifically designed meters to capture data for our purposes. Through evaluation of the data we were able to identify redundant data sources through their high correlation to each other. These included our seconds and

hour features , outside temperatures (airport and house exterior sensor), and temperatures in the parent room and ironing room.

Part 3: Feature Engineering

Feature engineering was necessary to identify important features, remove duplicate or unreliable sources of data, and transform data into model digestible states. Time is an important predictor to the model, but was captured in a field that wasn't usable to our model. To utilize this feature, we had to transform the data into a usable format. The timestamp was transformed into month, time, hour, day, seconds, day of week, numerical week, and weekType (weekday/weekend). This created a problem for how to handle some of our new features (i.e. strings). The string features were encoded as integers (0- Mon, 1- Tues, and so on) to be used by the predication algorithms. We also combined our load electrical load variables (appliance load and light load) into one electrical load variable. Finally, we eliminated redundant features like time in seconds, clock time, temperature from the exterior house sensor, and temperature from the sensor in the parent's room.

Part 4: Prediction Algorithm

Models were created using linear regression, random forest, and neural network frameworks. We recommend using the random forest model because it had the lowest RMS, MAPE, and MAE and the highest R^2.

Part 5: Feature Selection

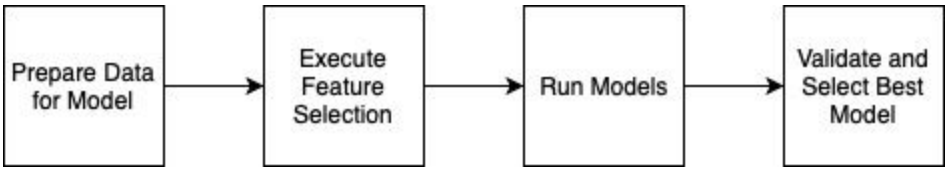
During feature selection, we evaluated multiple algorithms to select features which features would go into our automated pipeline. In general feature importance was quantified with a p-value and ranked in comparison to other features. Features that were eliminated were done so due to their low correlation to the target. The algorithms we tested were: boruta, tpot, and tsfresh. Boruta is wrapper method (forward/ backward selection) built on the random forest classifier. It works in an all relevant features manner which means the algorithm will use all the features it can to positively improve the prediction. Tpot is an automated machine learning approach that tests multiple machine learning methodologies to find the optimal result. The downside of this approach is that it takes a lot of time. Tsfresh is another wrapper method (like Boruta) that works to utilize all relevant features.

We chose Boruta for our pipeline due to the accuracy and speed of the predictions.

Part 6: Model Validation and Selection

To tune our model's hyperparameters, we executed cross validation techniques and regularization. Cross validation involves maintaining two sets of data, a training set and a testing set, and training the model against the training set and validating the model's accuracy against the testing set. This is done to eliminate the bias associated with using the same data to test the dataset. We can score the model by how it performs with the testing dataset.

Part 7: Final Pipeline



Our final model model involved our typical feature engineering (dropping our redundant variables mentioned earlier), using Boruta for feature selection, random forest prediction algorithm, random forest regressor to tune hyper parameters, and cross validation to validate the model. We used this pipeline because it is a good balance of accuracy and speed

References:

1. Data driven prediction models of energy use of appliances in a low-energy house. Luis M. Candanedo, Véronique Feldheim, Dominique Deramaix. Energy and Buildings, Volume 140, 1 April 2017, Pages 81-97, ISSN 0378-7788, http://dx.doi.org/10.1016/j.enbuild.2017.01.083.
2. Exploratory Data Analysis in Python; PyCon 2017; Silicon Valley Data Science
3. https://jakevdp.github.io/PythonDataScienceHandbook/05.04-feature-engineering.html
4. http://scikit-learn.org/stable/modules/feature_selection.html#feature-selection
5. https://machinelearningmastery.com/feature-selection-in-python-with-scikit-learn/
6. https://jakevdp.github.io/PythonDataScienceHandbook/05.03-hyperparameters-and-model-validation.html