Readme

July 4, 2023

1 Machine Learning for Renewable Energy Systems

University of Tübingen, Summer Term 2023, Timothy Leske

1.1 Individual Assignment - Coding Track - ReadMe

1.1.1 Building Electricity Demand

The code provided takes historic energy data for buildings and uses a linear regression model to forecast consumption for time horizons of one hour, one day and one week.

Data Pre-processing Raw utility meter data is imported and combine into a single .csv. The code used was provided along with the original datasets (available here https://github.com/buds-lab/building-data-genome-project-2). This section of the code only needs to be used if the processed dataset has already been generated.

If the processed dataset is already available, it can also be read in at this point (see code) and the previous step can be skipped.

For the purpose of this assignment, the overall dataset is filtered to extract data just for the buildings where benchmarks were provided for comparison. Non-electricity data was discarded (for now), as its used was not found to improve performance of the currently used model.

The building_benchmark_data.csv available in this repo contains the data used for the evaluation in this code, so the full data pre-processing does not need to be carried out.

Feature Engineering This function in the code below used to generate features for each timestep, based on the forecast horizon provided. In each case, a week of previous consumption is added as 168 features for each timestep. The function chooses the offset for the previous consumption based on whether the horizon is daily, hourly or weekly. For example, if a weekly forecast horizon is chosen, hourly consumption for every hour between 2 weeks prior and 1 week prior to the current time will be added as features.

In addition, radial basis functions are added for the hour, month and day of week, to allow the model to identify cyclic trends in the dataset.

The feature engineering process results in NaN values at the start of the dataset, which are discarded. Interpolation is used to fill in actual missing electricity data values.

Model Specification Data is split into test and train sets, with 2016 used for training and 2017 used for testing.

A test train model function allows for a choice of model. Later, a simple ridge linear regression algorithm is used.

Forecasting In the section, energy consumption predictions are generated using the functions specified previously, along with a basic ridge linear regression model. This is performed for hourly, daily and weekly time horizons, and in each case RMSE and MAE are calculated. The predicted consumption values are saved into a dataframe for plotting purposes. The calculated errors are saved into a dataframe for evaluation against the provided benchmarks.

It is noted that several models were evaluated here, including gradient boosting and random forest regressors. These models were much slower than the linear regression, and were not found to perform better with the current feature set. There is obviously potential to use more sophisticated models to achieve better results, which will be investigated.

Evaluation against Benchmarks The calculated errors for the forecasts are compared below to the provided benchmarks. In the majority of cases, the trained model achieves RMSE and MAE values similar to or better than the benchmark, however it appears to have performed more poorly for buildings where the benchmark RMSE or MAE was already high.

Visualisation Predicted and actual consumption values were plotted for the first two weeks of the test set period to identify obvious model shortcomings: - There is what possibly looks like a one day lag between the actual and predicted values in the daily model, suggesting that the model is basing predictions too heavily on the previous day of data, and not accounting sufficiently for weekly cycling variation. - There appear to be some meter failures during the test period, already visible in the first two weeks (notably for Hog_office_Bill and Lamb_assembly_Bertie). Regardless of our model, we will not be able to predict these failures in advance, so the RMSE will inevitably be high for buildings where frequent failures occur. Meter failures during the training period may be affecting model training.

Extension Ideas

- Account for daylight savings in datasets
- Use meter data for other utility meters as additional features
- Use data for other buildings in the same category
- Use more advanced machine learning algorithms
- Use past weather data and weather forecasts
- Identify meter failures in training data and make sure this data is not used for training?