Title : Appliance Energy Prediction

Author’s Name: Mohd abdul samad

Keywords: Data Science, machine learning, Linear regression

26 March 2023

Abstract

Currently there is an uncontrollable damage to the environment because of rapid consumption of natural resources of Earth. The increase of C02 is rampant and the damage to the ozone layer is critical. The usage of appliances in daily households also are contributing to the damage of Earth and its environment. Tracking the usage and the amount of energy can be very useful in curbing the problems by keeping the usage in control. We are tasked with tracking the usage using supervised ML algorithms.

Problem Statement

We are tasked with predicting the amount of energy consumed in watt per hour(wh) by tracking the usage of appliances in the household from the data collected throughout 4.5 months at every 10 minute interval to understand the trend and growth of energy consumption of residential buildings in Belgium.

The dataset contains weather information like Temperature, Humidity, Windspeed, Visibility, TDewpoint, Lights, Pressure), the Appliances used in terms of energy (Wh) and date information.

Attribute Information:

- Date: Date and time of Appliances usage recorded
- Appliances : Values of appliance usage in Watt per hour(Wh)
- Lights : Energy use of light fixtures in the house in Wh
- T1 : Temperature in kitchen area, in Celsius

- Rh1 : Humidity in kitchen area, in %
- T3 : Temperature in laundry room area
- Rh3 : Humidity in laundry room area in %
- T4: Temperature in office room, in Celsius
- Rh4: Humidity in office room, in %
- T5: Temperature in bathroom, in Celsius
- Rh5: Humidity in bathroom, in %
- T6 : Temperature outside the building(north side), in %
- Rh6: Humidity outside the building(north side), in %
- T7: Temperature in ironing room, in Celsius
- Rh7: Humidity in ironing room, in %
- T8: Temperature in teenager room 2, in Celsius
- Rh8: Humidity in teenager room 2, in %
- T9: Temperature in parents room 2 in Celsius
- Rh9: Humidity in parents room, in %
- To: Temperature outside(from Chievres weather station), in Celsius
- Pressure (from Chievres weather station), in mmHg
- RHout, Humidity outside in %
- Wind Speed in m/s
- Visibility in Km
- Tdewpoint in Â°C
- rv1, Random variable 1, nondimensional
- rv2, Random variable 2, nondimensional

Steps Involved

A) Exploratory Data Analysis

This analysis is important in order to gain useful insights within data and also with respect to dependent variables. In EDA variables are compared using plots and meaningful insights are drawn. It gives us a better idea of which feature behaves in which manner compared to the target variable.

B) Data Cleaning

Checked for null values in the columns and there weren’t any.

C) Encoding of Categorical Columns

All the datatypes for the columns of the dataset were of integers and float so we found that there isn’t any use of encoding the columns.

D) Feature Scaling

Feature scaling is essential for machine learning algorithms that calculate distances between data. If not scale, the feature with a higher value range starts dominating when calculating distances.

E) Fitting different Models

At first we tried with basic linear regression, but soon realized we will need a much more complex model and so we then used Decision tree Regressor, Random Forest Regressor, Extra Trees Regressor and XGBoost Regressor. Model is then boosted and results are compared.

Algorithms

A) Regression

Regression searches for relationships among variables.

For example, you can observe several employees of some company and try to understand how their salaries depend on the features, such as experience, level of education, role, city they work in, and so on.

This is a regression problem where data related to each employee represent one observation. The presumption is that the experience, education, role, and city are the independent features, while the salary depends on them.

Similarly, you can try to establish a mathematical dependence of the prices of houses on their areas, numbers of bedrooms, distances to the city center, and so on.

Generally, in regression analysis, you usually consider some phenomenon of interest and have a number of observations. Each observation has two or more features. Following the assumption that (at least) one of the features depends on the others, you try to establish a relation among them.

In other words, you need to find a function that maps some features or variables to others sufficiently well.

The dependent features are called the dependent variables, outputs, or responses.

The independent features are called the independent variables, inputs, or predictors.

Regression problems usually have one continuous and unbounded dependent variable. The inputs, however, can be continuous, discrete, or even categorical data such as gender, nationality, brand, and so on.

It is a common practice to denote the outputs with y and inputs with x. If there are two or more independent variables, they can be represented as the vector x= (x1, …, xr), where r is the number of inputs.

B) Decision Tree Regressor

Decision Tree is a decision-making tool that uses a flowchart-like tree structure or is a model of decisions and all of their possible results, including outcomes, input costs and utility.

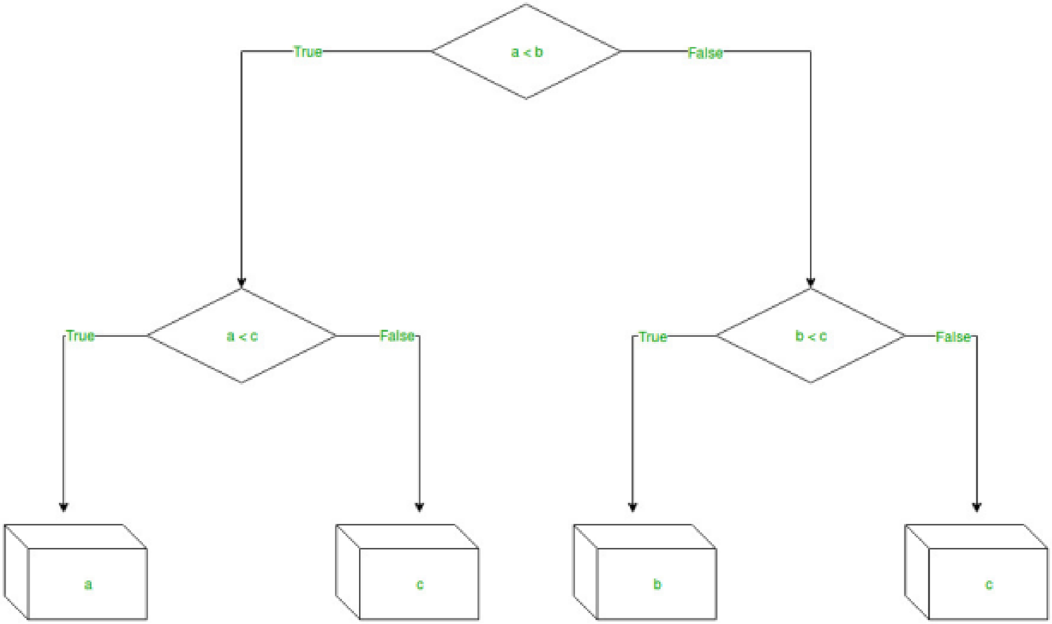
Decision-tree algorithms fall under the category of supervised learning algorithms. It works for both continuous as well as categorical output variables.

The branches/edges represent the result of the node and the nodes have either:

1. Conditions [Decision Nodes]

2. Result [End Nodes]

The branches/edges represent the truth/falsity of the statement and takes makes a decision based on that in the example below which shows a decision tree that evaluates the smallest of three numbers:



Decision tree regression observes features of an object and trains a model in the structure of a tree to predict data in the future to produce meaningful continuous output. Continuous output means that the output/result is not discrete, i.e., it is not represented just by a discrete, known set of numbers or values.

C) Random forest regression model:

Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision trees at training time.

For classification tasks, the output of the random forest is the class selected by most trees. For regression tasks, the mean or average prediction of the individual trees are returned.

Random decision forests correct for decision trees' habit of overfitting to their training set. Random forests generally outperform decision trees, but their accuracy is lower than gradient boosted trees. However, data characteristics can affect their performance.

D) Extra-trees regression model:

Extra Trees is an ensemble machine learning algorithm that combines the predictions from many decision trees. It is related to the widely used random forest algorithm. It can often achieve as good or better performance than the random forest algorithm, although it uses a simpler algorithm to construct the decision trees used as members of the ensemble. It is also easy to use given that it has few key hyperparameters and sensible heuristics for configuring these hyperparameters.

E) XGBoost Regression Model:

XGBoost is an ensemble learning and a gradient boosting algorithm for decision trees that uses a second-order approximation of the scoring function. This approximation allows XGBoost to calculate the optimal “if” condition and its impact on performance.

XGBoost can then store these in its memory in the next decision tree to save recomputing it.

While training, the XGBoost algorithm constructs a graph that examines the input under various “if” statements (vertices in the graph). Whether the “if” condition is satisfied influences the next “if” condition and eventual prediction.

XGBoostprogressively adds more and more “if” conditions to the decision tree to build a stronger model. By doing so, the algorithm increases the number of tree levels, therefore, implementing a level-wise tree growth approach.

XGBoost learns a model faster than many other machine learning models(especially among the other ensemble methods) and works well on categorical data and limited datasets.

Model Performance

1. **R-squared (R2)**, which is the proportion of variation in the outcome that is explained by the predictor variables. In multiple regression models, R2 corresponds to the squared correlation between the observed outcome values and the predicted values by the model. The Higher the R-squared, the better the model.

2. **Root Mean Squared Error (RMSE),** which measures the average error performed by the model in predicting the outcome for an observation. Mathematically, the RMSE is the square root of the mean squared error (MSE), which is the average squared difference between the observed actual outcome values and the values predicted by the model. So, MSE = mean((observed - predicted)^2) and RMSE = sqrt(MSE). The lower the RMSE, the better the model.

3. **Residual Standard Error (RSE),** also known as the model sigma, is a variant of the RMSE adjusted for the number of predictors in the model. The lower the RSE, the better the model. In practice, the difference between RMSE and RSE is very small, particularly for large multivariate data.

4. **Mean Absolute Error (MAE),** like the RMSE, the MAE measures the prediction error. Mathematically, it is the average absolute difference between observed and predicted outcomes, MAE = mean(abs(observed - predicted)). MAE is less sensitive to outliers compared to RMSE.

5. **HyperParameter Tuning** ,Hyper-parameters are those sets of information that are used to control our parameters in order to get good results. We used Grid Search CV for hyper parameter tuning.

6. **Grid Search CV** : It is the process of performing hyperparameter tuning in order to determine the optimal values for a given model. As mentioned above, the performance of a model significantly depends on the value of hyperparameters. Note that there is no way to know in advance the best values for hyperparameters so ideally, we need to try all possible values to know the optimal values. Doing this manually could take a considerable amount of time and resources and thus we use GridSearchCV to automate the tuning of hyperparameters.

GridSearchCV is a function that comes in Scikit-learn's(or SK-learn) model_selection package.So an important point here to note is that we need to have Scikit-learn library installed on the computer. This function helps to loop through predefined hyperparameters and fit your estimator (model) on your training set. So, in the end, we can select the best parameters from the listed hyperparameters.

Conclusion

We are finally at the conclusion of our project! Coming from the beginning we did EDA on the dataset and also cleaned the data according to our needs.After that we were able to draw relevant conclusions from the given data and then we trained our model on linear regression and other models.

Out of all models used , with the extra-trees regression model we were able to get the r2-score of 0.80.The model which performed poorly was Lasso Regression model with r2-score of 0.31. Given the size of data and the amount of irrelevance in the data , the above score is good.