ASHRAE—GREAT ENERGY PREDICTOR III: HOW MUCH ENERGY WILL A BUILDING CONSUME?

Pierre-Antoine Ksinant

April 6, 2020

ABSTRACT

In this report, we will study a challenge proposed through Kaggle by ASHRAE in 2019, with the aim to develop accurate models of metered building energy usage in the following areas: chilled water, electricity, hot water, and steam meters. Indeed, thankfully, significant investments are being made to improve building efficiencies to reduce costs and emissions, and, under pay-for-performance financing, building owners make payments based on the difference between their real energy consumption and what they would have used without any retrofits. Thus, with better estimates of these energy-saving actions, large scale investors and financial institutions will be more inclined to invest to enable progress in building efficiencies.

***K*eywords** Kaggle ⋄ ASHRAE ⋄ Machine Learning ⋄ Extreme Gradient Boosting ⋄ Light Gradient Boosting Machine ⋄ Shapley Additive Explanations

1 INTRODUCTION

1.1 Project Overview

In 2019, through Kaggle[1], the American Society of Heating, Refrigerating and Air-Conditioning Engineers (ASHRAE)[2] organized a challenge[3] with the aim to develop accurate models of metered building energy usage in the following areas: chilled water, electricity, hot water, and steam meters.

Indeed, thankfully, significant investments are being made to improve building efficiencies to reduce costs and emissions, and, under pay-for-performance financing, building owners make payments based on the difference between their real energy consumption and what they would have used without any retrofits. Thus, with better estimates of these energy-saving actions, large scale investors and financial institutions will be more inclined to invest to enable progress in building efficiencies.

Current methods of estimation are fragmented and do not scale well, some assume a specific meter type or don't work with different building types.

As challenge's title indicates, it's not the first time ASHRAE is involved in this type of activities, it has previously hosted 2 data competitions, called "Great Building Energy Predictor Shootout I" (1993) [8] and "Great Building Energy Predictor Shootout II" (1994) [6]. For both competitions, participants were asked to develop empirical models for predicting building energy consumption from data sets, and to compare how these models could be used to forecast energy usage (Shootout I) and calculate energy conservation retrofit savings (Shootout II).

For history, the 1994 competition asked participants to retrieve the competition training data set via an FTP server and teams were required to submit their empirical models along with predictions via floppy disks. Each submitted package included predictions of energy savings and a sufficient explanation of how the specific method (calculation method, data removal...) was applied: Submissions using black-box, or proprietary methods, were disqualified. It was a combination accuracy metric of CV-RMSE (Coefficient of Root Mean Square Error) and MBE (Mean Bias Error) that was used to evaluate prediction accuracy. While this 1994 competition awarded no monetary prizes, over 150 teams competed, and 6 winning teams were formally recognized and asked to write an ASHRAE paper for presenting at ASHRAE conferences to document their efforts: As a part of

[1] Kaggle is an online community of data scientists and machine learners, owned by Google LLC: https://www.kaggle.com.

[2] Founded in 1894, ASHRAE is an American professional association, which counts more than $54,000$ members—mainly building services engineers, architects, mechanical contractors, building owners and equipment manufacturers employees— serving in more than 132 countries worldwide, and seeks to advance healthing, ventilation, air conditioning and refrigeration (HVAC&R) systems design and construction; it supports research, standards writing, publishing and continuing education, shaping tomorrow's built environment today: https://www.ashrae.org.

[3] The challenge was entitled "ASHRAE—Great Energy Predictor III: How much energy will a building consume?": https://www.kaggle.com/c/ashrae-energy-prediction.

these efforts, over a dozen peer-reviewed papers were published (e.g., see [11], winning contribution for "Great Building Energy Predictor Shootout I" (1993), or [3], winning contribution for "Great Building Energy Predictor Shootout II" (1994)) and several software vendors incorporated the algorithms described in these papers.

Finally, it can be noticed that the present challenge inscribes itself in current initiatives trying to limit or optimize energy consumption thanks to machine learning. An interested reader can consult, for example, the 2 following articles: "Machine Learning Can Boost The Value Of Wind Energy"[4], by Carl Elkin and Sims Witherspoon from DeepMind[5], and "How AI Could Smarten Up Our Water System"[6], by Mary Catherine O'Connor.

1.2 Problem Statement

To allow participants to build their solution to tackle the proposed challenge, ASHRAE provided 6 tabular data files[7] to support the construction of the prediction model, and selected 1 quality metric to evaluate and rank the different solutions it expected to receive.

For the challenge, the data comes from over $1,000$ buildings over a 3-year timeframe: To summarize—data will be discussed more in detail latter—, over the 3-year timeframe, it has been provided information about energy usage (electricity, chilled water, steam and hot water) registered on various buildings, with different characteristics, according to the various weather conditions that have happened.

The design of our study plan is largely inspired by [5] and focuses on "benchmarking" 2 current popular and very performant methods: Extreme Gradient Boosting (also known as XGBoost, see [2]) and Light Gradient Boosting Machine (also known as LightGBM, see [7]).

Finally, it can be noticed that our study has been technically supported and run on a machine (see Table 1) with following hardware main characteristics:

Table 1: Hardware main characteristics

Model Name	MacBook Pro
Model Identifier	MacBookPro9,2
Processor Name	Dual-Core Intel Core i5
Processor Speed	2.5 GHz
Number of Processors	1
Total Number of Cores	2
L2 Cache (per Core)	256 KB
L3 Cache	3 MB
Hyper-Threading Technology	Enabled
Memory	4 GB
Boot ROM Version	229.0.0.0.0
SMC Version (system)	2.2f44
Serial Number (system)	C02JC2HXDTY3
Hardware UUID	F0F7E293-C121-5E28-87C0-D169FDA41B45
Sudden Motion Sensor State	Enabled

1.3 Data

The core data on which the prediction models will be constructed is reported on 6 tabular data files:

- *building_metadata.csv*: Key characteristics of the buildings taken into account for the study (building identification, primary category of activities[8], gross floor area, opening year and number of floors);
- *sample_submission.csv*: Correct format for submitting predictions for the challenge;
- *test.csv*: Testing set (building identification, primary category of activities and timestamp);
- *train.csv*: Training set (building identification, primary category of activities, timestamp and energy consumption);
- *weather_test.csv*: Weather data from a meteorological station as close as possible to the site (building identification, timestamp, air temperature, dew temperature, cloud coverage, precipitation depth in the hour, sea level pressure, wind direction and wind speed);
- *weather_train.csv*: Weather data from a meteorological station as close as possible to the site (building identification, timestamp, air temperature, dew temperature, cloud coverage, precipitation depth in the hour, sea level pressure, wind direction and wind speed).

Nonetheless, if tabular data files *sample_submission.csv*, *test.csv* and *weather_test.csv* had their interest in the context of the challenge, here, we can discard them for our analysis: They don't contain usable information (energy consumption is not provided), and, so, are not relevant anymore.

[4]See blog post: https://deepmind.com/blog/article/machine-learning-can-boost-value-wind-energy.

[5]DeepMind is a UK artificial intelligence company founded in September 2010 and acquired by Google LLC in 2014: https://deepmind.com.

[6]See blog post: https://medium.com/s/ai-for-good/how-ai-could-smarten-up-our-water-system-f965b87f355a.

[7]The association has been generously helped for data collection by the following organizations: SinBerBEST2 (Singapore Berkeley Building Efficiency and Sustainability in the Tropics 2—http://sinberbest.berkeley.edu), Buds Lab (Building and Urban Data Science—https://www.budslab.org) and TEES (Texas A&M Engineering Experiment Station—https://tees.tamu.edu).

[8]Indicator based on EnergyStar property type definitions, see more detail here: https://www.energystar.gov/buildings/facility-owners-and-managers/existing-buildings/use-portfolio-manager/identify-your-property-type.

As tabular data files *building_metadata.csv*, *train.csv* and *weather_train.csv* will be the main base to build our prediction models, we are going to detail them more precisely (see, respectively, Table 2, Table 3 and Table 4):

Table 2: Detail of *building_metadata.csv*

Label	Type
site_id	integer
building_id	integer
primary_use	string
square_feet	integer
year_built	float
floor_count	float

Table 3: Detail of *train.csv*

Label	Type
building_id	integer
meter	integer
timestamp	date
meter_reading	float

Table 4: Detail of *weather_train.csv*

Label	Type
site_id	integer
timestamp	date
air_temperature	float
cloud_coverage	float
dew_temperature	float
precip_depth_1_hr	float
sea_level_pressure	float
wind_direction	float
wind_speed	float

Tabular data file *building_metadata.csv* contains 1449 data points with 6 variables each, for $1,868$ missing values, tabular data file *train.csv* contains $20,216,100$ data points with 4 variables each (no missing values), and *weather_train.csv* contains $139,773$ data points with 9 variables each, for $136,820$ missing values.

1.4 Quality Metric

ASHRAE provided 1 quality metric to evaluate prediction models: Root Mean Squared Logarithmic Error (RMSLE).

It can be calculated like this:

$$\epsilon = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\ln(\hat{y}_i+1)-\ln(y_i+1))^2}$$

Where:

- ϵ is the RMSLE score;
- n is the total number of observations in the data set;
- $(\hat{y}_i)_{i\in[\![1,n]\!]}$ is the prediction value of the target variable in the data set;
- $(y_i)_{i\in[\![1,n]\!]}$ is the right value of the target variable in the data set;
- $\ln$ is the natural logarithm function.

Although Root Mean Square Error (RMSE) is generally the quality metric used for regression problems, RMSLE is more and more put to contribution for these problems.

2 DATA COMPREHENSION

2.1 Exploration and Analysis

2.1.1 building_metadata.csv

The first tabular data file lists some aspects of the buildings taken into account for this study (see Table 2). Several elements are interesting to observe:

- Buildings' primary use repartition on the data set exploited for the study (see Figure 1);
- Buildings' repartition by site ID on the data set exploited for the study (see Figure 2);
- Buildings' gross floor area variability on the data set exploited for the study (see Figure 3).

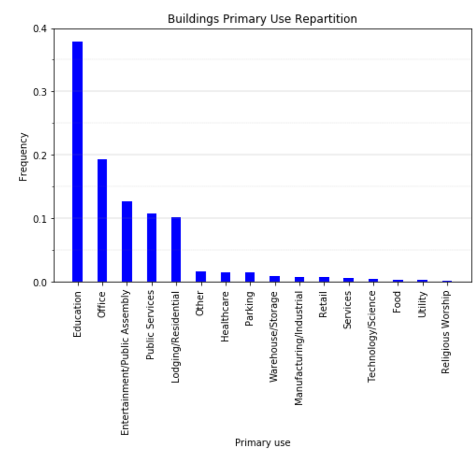

Figure 1: Buildings' primary use repartition

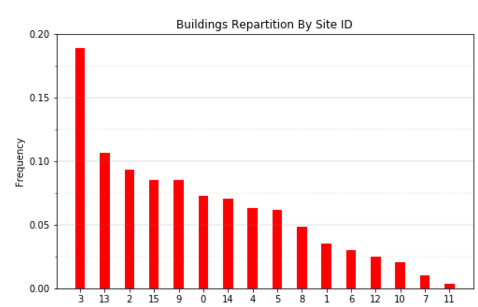

Figure 2: Buildings' repartition by site ID

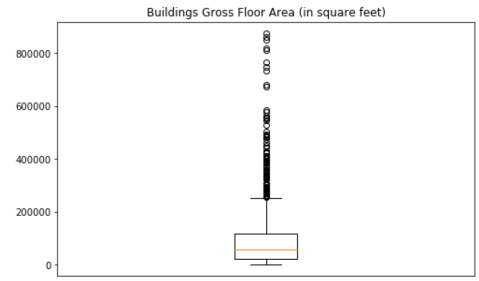

Figure 3: Buildings' gross floor area variability

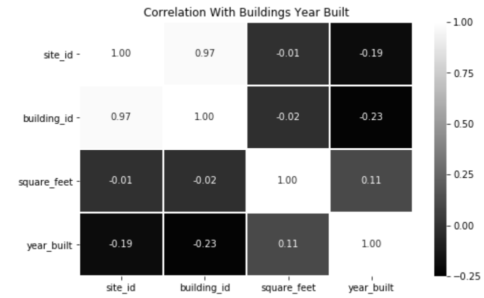

Figure 4: Buildings' year built correlation matrix

Some statements can be made:

- It appears that for this study, the primary use of the buildings that have been taken into account is reparted around 16 categories, not very well balanced, with a large majority of them (more than a third) dedicated to education. We can suppose that this feature has a great incidence over a building energy consumption, independently of its other characteristics: Indeed, a building dedicated to education works a very different way from another dedicated to religious worship, for example. Thus, it is quite probable that the prediction models we are going to build will present some kind of bias, due to the fact that they will be, in a great part, built thanks education buildings, and, so, will be consistent with this type of buildings, but probably not so with other types of buildings;

- The buildings taken into account for this study are reparted around 16 different geographical sites, with their own specific weather conditions. As for previous conclusion, the observed repartition is not well balanced, thus, too, it is quite probable that the prediction models we are going to built will present some kind of bias, due to the fact that during the training they will be more exposed to some specific weather conditions;

- Respectively to buildings gross floor area, we note approximately 7.32% of outliers, element important to tackle, as it appears obvious that a building's gross floor area impacts its energy consumption.

During the exploration we performed of this tabular data file, we observed that the missing values are concentrated into 2 columns, *year_built* and *floor_count*, and that these missing values represented a significant proportion of all registered values for these 2 features (respectively 53.42% and 75.50%). Furthermore, we noted no strong correlation between each one of these 2 features and the other ones (see Figures 4 and 5), which significates that these 2 features contain proper information: This is a serious issue.

Figure 5: Buildings' floor count correlation matrix

2.1.2 *weather_train.csv*

The second tabular data file (see Table 4) contains, over the year 2016, some weather's conditions registered on the various site IDs exploited during this study (see Figures 6, 7, 8 and 9).

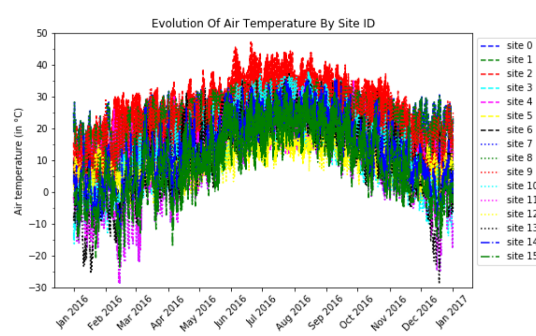

Figure 6: Evolution of air temperature by site ID (2016)

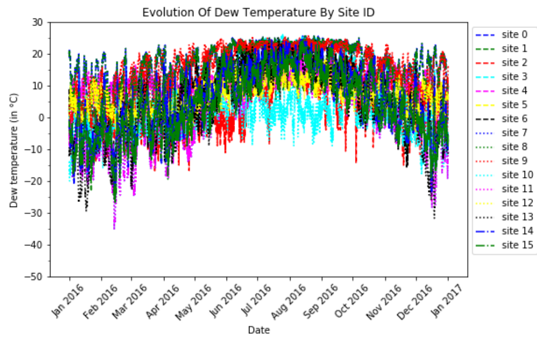

Figure 7: Evolution of dew temperature by site ID (2016)

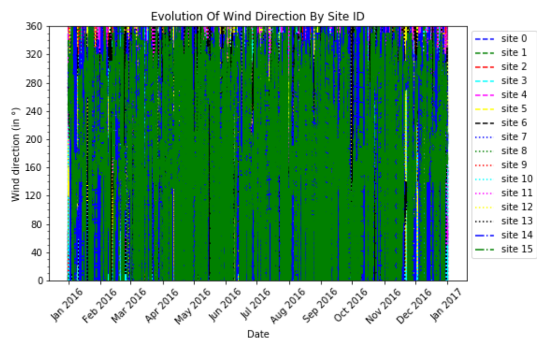

Figure 8: Evolution of wind direction by site ID (2016)

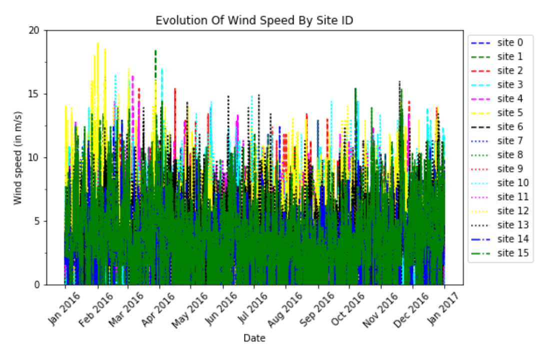

Figure 9: Evolution of wind speed by site ID (2016)

On this tabular data file, some conclusions can be stated:

- By site ID, evolution of air temperature, dew temperature, wind direction and wind speed covers un important spectrum of possibilities, nonetheless, for air temperature, for exemple, depending of the month of the year (and season), some common tendencies can be noted whatever the site ID considered: This will have an impact on the prediction models we will build, indeed, they will be trained only with data related to year 2016, and they won't probably be able to catch year periodicity on the data and associated energy consumption (they won't probably be able to catch, too, certain epiphenomena, as Christmas and New Year holidays, element which provocs a discontinous and brutal change on energy consumption);
- Features *cloud_coverage*, *precip_depth_1_hr*, and, to a lesser extent, *sea_level_pressure*, count an import proportion of missing values (respectively 49.49%, 35.98% and 7.60%), while this is not the case for the other features (less than 5%). This is an important issue for these features (for *cloud_coverage*, e.g., we can reasonably assume that the impact on electricity consumption to light a building is not insignificant).

2.1.3 train.csv

The third—and last—tabular data file (see Table 3) contains, over the year 2016, the various metered energy consumption registered.

First, we can consider Figure 10.

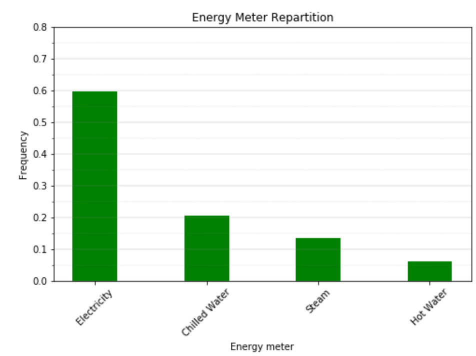

Figure 10: Energy meter repartition (2016)

As it can be observed, the energy meter repartition is not well balanced on the data set, this is not really an issue because we are going to build 4 prediction models (one for each energy type), nevertheless, we can already suppose that the one we will build for electricity consumption prediction will probably be better than the one we will build for hot water consumption prediction (at least, we will benefit from more data to train the first one than we will benefit for training the second one).

Another interesting graph we can plot concerns the evolution of each energy type consumption over the year 2016 (see Figures 11, 12, 13 and 14).

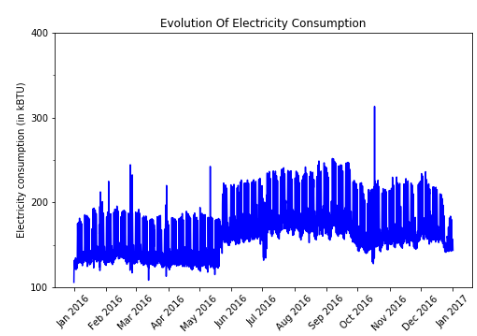

Figure 11: Evolution of electricity consumption (2016)

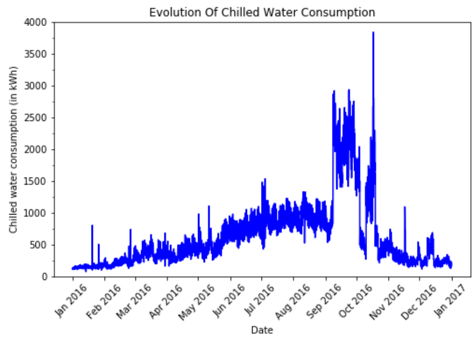

Figure 12: Evolution of chilled water consumption (2016)

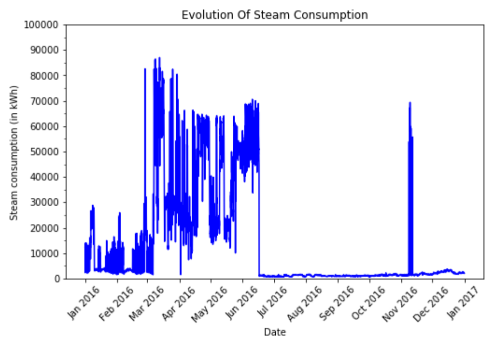

Figure 13: Evolution of steam consumption (2016)

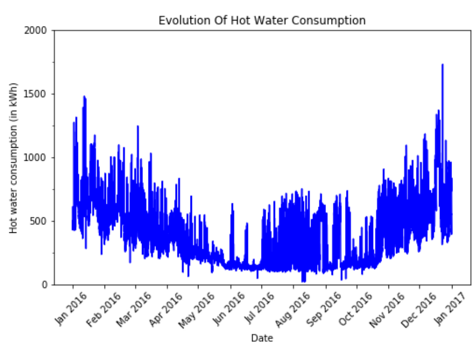

Figure 14: Evolution of hot water consumption (2016)

3 types of statements can be done respectively to these graphs:

- We can observe an high versatility on energy consumption data we benefit from, element that can be observed for each one of the 4 energy types (we note, too, a proportion of outliers superior to 10% for each one of them), we can thus already say that this will be a challenging point to handle for the prediction models we will build;
- For this study, we benefit from one year of registered data, as we have previously noted when discussing weather conditions, this will have an impact on the prediction models we will build: Indeed, they will be trained only with data related to year 2016, and they won't probably be able to catch year periodicity on the data and associated energy consumption (they won't probably be able to catch, too, certain epiphenomena, as Christmas and New Year holidays, element which provocs a discontinous and brutal change on energy consumption);
- The graph exposing the evolution of steam consumption over the year used for this study seems quite "strange". Indeed, from june to december (with a very strange "Dirac epiphenomena" in November), it seems like if steam consumption meters have been inactive.

2.2 *Preprocessing*

We have now a better vision of the problem and the way we are going to tackle it:

1. The target variable is clearly identified, *meter_reading*, and is closely linked to variable *meter*, which allows to indentify with which one of the 4 energy types (electricity, chilled water, steam and hot water) it is related;
2. Before merging the 3 tabular data files—*building_metadata.csv*, *train.csv* and *weather_train.csv*—into one consistent data set, we need to perform independent operations:
   - On tabular data files *building_metadata.csv* and *weather_train.csv*, we take the decision to drop features with too much missing values (features *year_built* and *floor_count* on tabular data file *building_metadata.csv*, and features *cloud_coverage*, *precip_depth_1_hr* and *sea_level_pressure* on tabular data file *weather_train.csv*);
   - On tabular data file *weather_train.csv*, we reconstruct missing values for features *air_temperature*, *dew_temperature*, *wind_direction* and *wind_speed*, replacing them by their previous value in the chronological process used to register data during the study;
3. After merging the 3 tabular data files, 4 steps need to be completed:
   - Common outliers, independently from energy type, are removed (features *square_feet* and *meter_reading*);
   - Cyclical features are transformed to take into account this aspect (features *timestamp* and *wind_direction*);
   - We handle *primary_use* categorical feature, making it a binary dummy variable;
   - We standardize continuous features (*square_feet*, *air_temperature*, *dew_temperature* and *wind_speed*);
4. We terminate the preprocessing with the splitting of the data set into 4 parts, each one of them corresponding to one particular energy type (electricity, chilled water, steam and hot water), and with removing outliers for meter reading registrations.

This way, we obtain a consistent data set—to split, for each one of the 4 energy types, between a training set (80%) and a testing set (20%) in a shuffled way—composed by:

- $9,868,987$ data points, 39 feature variables and 1 target variable for building electricity meter consumption forecasting model;
- $2,922,250$ data points, 39 feature variables and 1 target variable for building chilled water meter consumption forecasting model;
- $1,895,906$ data points, 39 feature variables and 1 target variable for building steam meter consumption forecasting model;
- $791,788$ data points, 39 feature variables and 1 target variable for building hot water meter consumption forecasting model.

3 PROBLEM MODELING

3.1 General Methodology

As it was precised in Subsection 1.2, in this project, our main focus is to perform a "benchmarking" of 2 current popular and very performant methods: Extreme Gradient Boosting (also known as XGBoost, see [2]) and Light Gradient Boosting Machine (also known as LightGBM, see [7]). For that, using Root Mean Squared Logarithmic Error (RMSLE) as quality metric to evaluate produced prediction models (see Subsection 1.4) and a "naive" forecasting model (a forecasting prediction model which always predicts the mean value of all meter readings registered in each energy type data set), to appreciate the value provided by these 2 methods, we are going to put them in action, and tune their hyperparameters as the hardware at our disposal (see Table 1) allows us to do it.

Respectively to XGBoost, as precised in its documentation, we can note that:

"XGBoost is an optimized distributed gradient boosting library designed to be highly efficient, flexible and portable. It implements machine learning algorithms under the Gradient Boosting framework. XGBoost provides a parallel tree boosting (also known as GBDT, GBM) that solve many data science problems in a fast and accurate way."[9]

In respect LightGBM, as written in its documentation, the following key points can be noted:

"LightGBM is a gradient boosting framework that uses tree based learning algorithms. It is designed to be distributed and efficient with the following advantages: Faster training speed and higher efficiency, lower memory usage, better accuracy, support of parallel and GPU learning, capable of handling large-scale data."

Recently, both XGBoost and LightGBM have been widely-used in many winning solutions of machine learning competitions.

[9]To obtain further information about Gradient Boosting, an interested reader can consult [4].

3.2 Hyperparameters Tuning

As discussed in Subsection above, the hardware at our disposal for this project constitutes a pain point, and, to a certain—and limited—point, we have only been able to perform hyperparameters' tuning with training data set dedicated to buildings' hot water meter energy consumption forecasting, the smallest of the 4 ones for the problem to tackle.

Although, obviously, it's not the best solution to take, we have conserved the tuned hyperparameters we have obtained and used them for the 3 other energy types: Indeed, same patterns can be observed in all energy types consumption, so, although it's not the best solution to take, it's not the worst either.

The "best" key hyperparameters we have obtained for XGBoost and LightGBM methods are expressed respectively in Table 5 and in Table 6.

Table 5: XGBoost "best" key hyperparameters

Number of gradient boosted trees	77
Maximum tree depth for base learners	9
Boosting learning rate	0.4
Min loss reduction for tree's leaf node partition	0.5
L_1 regularization term on weights	5
L_2 regularization term on weights	0.01

Table 6: LightGBM "best" key hyperparameters

Boosting type	DART
Number of boosted trees to fit	100
Boosting learning rate	1
Maximum tree depth for base learners	5
L_1 regularization term on weights	0.01
L_2 regularization term on weights	0.01

3.3 Buildings' Electricity Meter Consumption Forecasting

The results we obtained constructing buildings' electricity meter consumption forecasting models are summarized in Table 7.

Table 7: Buildings' electricity meter consumption forecasting models "benchmarking"

Forecasting Model	RMSLE Score Training Set	Testing Set
"Naive"	1.426128	1.425909
XGBoost	0.739577	0.742597
LightGBM	0.783755	0.783860

As it can be observed, best scores, both on training and testing sets, are provided by XGBoost model.

An illustration of this benchmarking situation can be visualized on Figure 15.

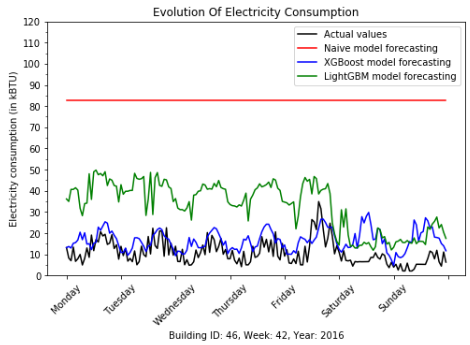

Figure 15: Buildings' electricity meter consumption forecasting models benchmarking illustration

Finally, to explain XGBoost model predictions on testing set—data not used for building the model, so, data which allows to observe model's behaviour when confronted with "unknown" data, "real-life" data—, SHAP (SHapley Additive exPlanations) values have been mobilized, allowing us a better comprehension of the forecasting processing: This game theoretic approach is being increasingly used to explain the output of any machine learning model (see [10] and [9]).

An overview of which features are most important for the model is plotted on Figure 16, where the SHAP values of every feature for every sample are summarized.

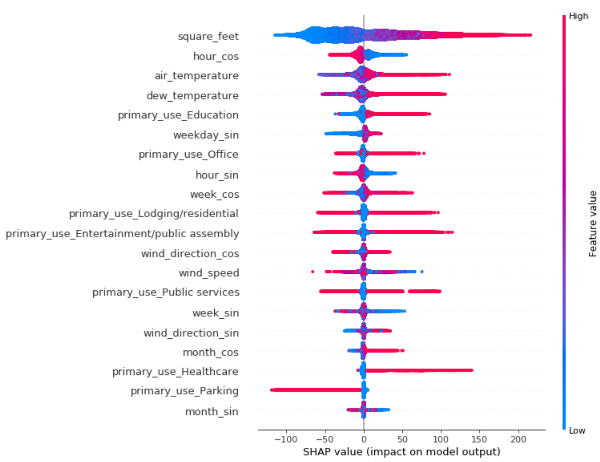

Figure 16: XGBoost model predictions on testing set explanation with SHAP values

The plot above sorts features by the sum of SHAP value magnitudes over all samples, and uses SHAP values to show the distribution of the impacts each feature has on the model output.

Here, we can observe what features drive the predictions on the testing set, and, thus, if we consider the "top 5" features, by order of importance, we have:

1. Surface of the building;
2. Hour's position in the day;
3. Air temperature;
4. Dew temperature;
5. Education as primary use.

So, for example, analyzing the 2 most important features, in accordance with what was intuitively expected, we can note that the highest the building's surface is, the more electricity consumption will be, and that being during labour hours, electricity consumption increases. It is equally interesting, too, to observe that air temperature and dew temperature have an important impact with the predictions of our model. Furthermore, if building's primary use belongs to education, office, residence, public assembly, public services, healthcare and parking, this aspect seems to be very discriminating for electricity consumption forecasting model's processing.

3.4 Buildings' Chilled Water Meter Consumption Forecasting

The results we obtained constructing buildings' chilled water meter consumption forecasting models are summarized in Table 8.

Table 8: Buildings' chilled water meter consumption forecasting models "benchmarking"

Forecasting Model	RMSLE Score Training Set	Testing Set
"Naive"	1.894896	1.894816
XGBoost	0.919122	0.927294
LightGBM	1.216306	1.217206

As it can be observed, best scores, both on training and testing sets, are provided by XGBoost model.

An illustration of this benchmarking situation can be visualized on Figure 17.

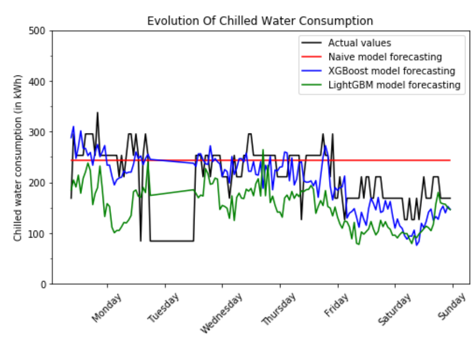

Figure 17: Buildings' chilled water meter consumption forecasting models benchmarking illustration

As it has been done in previous Subsection, exploiting SHAP values, we are now going to explain XGBoost model predictions on testing set.

An overview of which features are most important for the model is plotted on Figure 18, where the SHAP values of every feature for every sample are summarized.

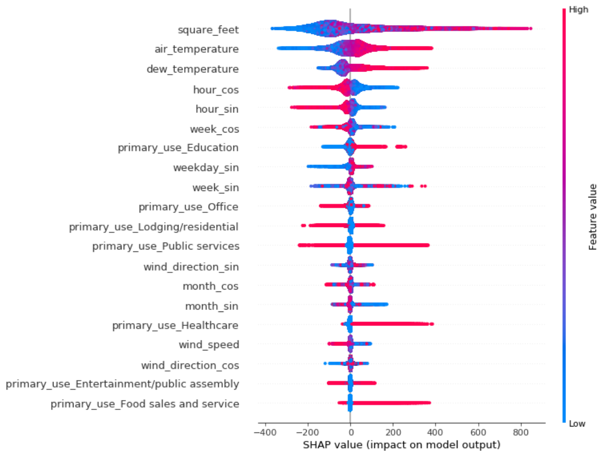

Figure 18: XGBoost model predictions on testing set explanation with SHAP values

In this graph, we can observe what features drive the predictions on the testing set, and, thus, if we consider the "top 4" features, by order of importance, we have:

1. Surface of the building;
2. Air temperature;
3. Dew temperature;
4. hour's position in the day.

So, as previously, for example, analyzing the 2 most important features, in accordance with what was intuitively expected, we can note that the highest the building's surface is, the more chilled water consumption will be, and, the hotest the air temperature is, the more chilled water consumption will be. It is equally interesting, too, to observe that dew temperature is the third most important feature for the model we have built, and that afternoons use to be moments in the day when chilled water consumption is high. Furthermore, if building's primary use belongs to public services, healthcare and food sales and service, this aspect seems to be very discriminating for chilled water consumption forecasting model's processing.

3.5 Buildings' Steam Meter Consumption Forecasting

The results we obtained constructing buildings' steam meter consumption forecasting models are summarized in Table 9.

Table 9: Buildings' steam meter consumption forecasting models "benchmarking"

Forecasting Model	RMSLE Score Training Set	Testing Set
"Naive"	1.760635	1.759955
XGBoost	0.818102	0.831114
LightGBM	1.026755	1.030987

As it can be observed, best scores, both on training and testing sets, are provided by XGBoost model.

An illustration of this benchmarking situation can be visualized on Figure 19.

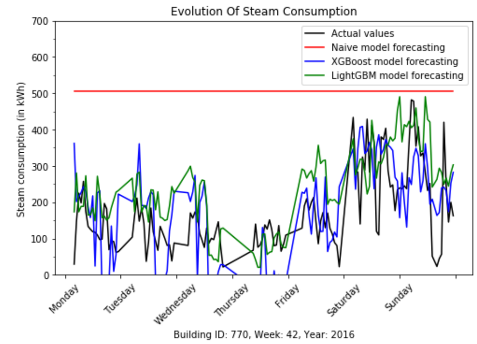

Figure 19: Buildings' steam meter consumption forecasting models benchmarking illustration

As it has been done in previous Subsections, exploiting SHAP values, we are now going to explain XGBoost model predictions on testing set.

An overview of which features are most important for the model is plotted on Figure 20, where the SHAP values of every feature for every sample are summarized.

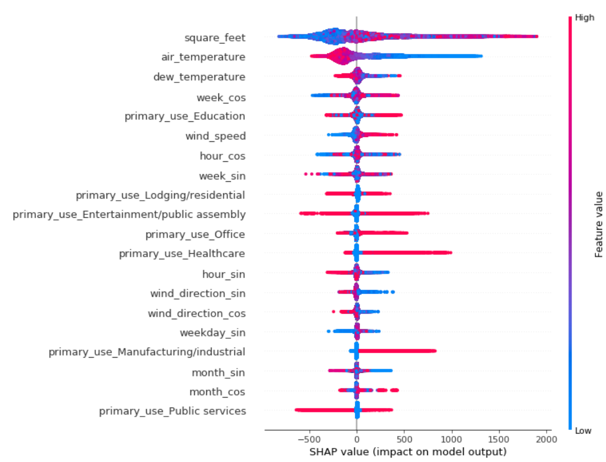

Figure 20: XGBoost model predictions on testing set explanation with SHAP values

In this graph, we can observe what features drive the predictions on the testing set, and, thus, if we consider the "top 5" features, by order of importance, we have:

1. Surface of the building;
2. Air temperature;
3. Dew temperature;
4. Week's position in the year;
5. Education as primary use.

So, for example, analyzing the 2 most important features, in accordance with what was intuitively expected, we can note that the highest the building's surface is, the more steam consumption will be, and, the hotest the air temperature is, the less steam consumption will be. It is equally interesting, too, to observe that dew temperature is the third most important feature for the model we have built. Furthermore, if building's primary use belongs to public assembly, healthcare, industry and public services, this aspect seems to be very discriminating for steam consumption forecasting model's processing.

3.6 Buildings' Hot Water Meter Consumption Forecasting

The results we obtained constructing buildings' hot water meter consumption forecasting models are summarized in Table 10.

Table 10: Buildings' hot water meter consumption forecasting models "benchmarking"

Forecasting Model	RMSLE Score Training Set	Testing Set
"Naive"	1.979692	1.982237
XGBoost	0.850988	0.881483
LightGBM	1.030566	1.035321

As it can be observed, best scores, both on training and testing sets, are provided by XGBoost model.

An illustration of this benchmarking situation can be visualized on Figure 21.

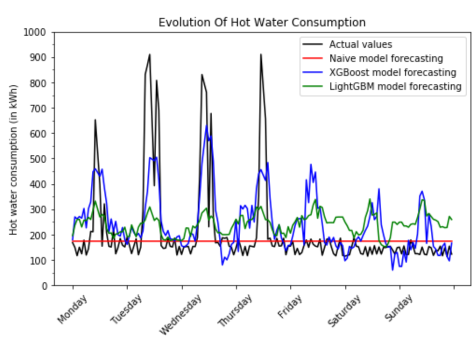

Figure 21: Buildings' hot water meter consumption forecasting models benchmarking illustration

As it has been done in previous Subsections, exploiting SHAP values, we are now going to explain XGBoost model predictions on testing set.

An overview of which features are most important for the model is plotted on Figure 22, where the SHAP values of every feature for every sample are summarized.

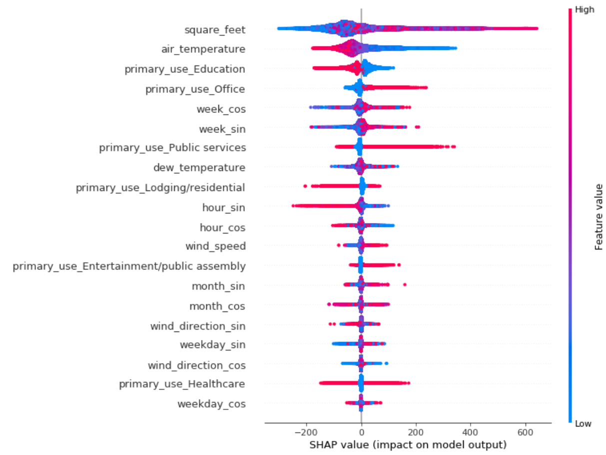

Figure 22: XGBoost model predictions on testing set explanation with SHAP values

In this graph, we can observe what features drive the predictions on the testing set, and, thus, if we consider the "top 5" features, by order of importance, we have:

1. Surface of the building;
2. Air temperature;
3. Education as primary use;
4. Office as primary use;
5. Week's position in the year.

So, for example, analyzing the 2 most important features, in accordance with what was intuitively expected, we can note that the highest the building's surface is, the more hot water consumption will be, and, the hotest the air temperature is, the less hot water consumption will be. It is interesting to note, too, that building's primary use (especially education, office, public services and healthcare) seems to have disruptive impact for hot water consumption forecasting model's processing.

4 PROPOSED SOLUTION

To conclude this report and our analysis on the challenge to tackle, based on the various statements and conclusions that have been established, we are led to propose XGBoost models to forecast each one of the buildings' 4 energy types metered consumption (electricity, chilled water, steam and hot water). Indeed, it's with these models—despite the limited hyperparameters tuning the hardware has allowed us to perform—the best scores have been reached.

Nevertheless, some indications can be provided to go further this work and deepen it:

- Firstly, a more complete work can be performed about the data put at disposal, principally about missing data and ways to explore to try te reconstruct it. For example, respectively to the data linked to weather conditions, trying to figure out the exact localizations of the 16 site IDs mobilized for this study, it's probably possible to gather the needed information to fill missing values, or at least some of them.

- Secondly, with less limited hardware, a consistent work about hyperparameters tuning—for both XGBoost and LightGBM models—could be realized. Here, we have operated a randomized search on key hyperparameters of both XGBoost and LightGBM, but we have been very limited in our action. An interesting perspective to explore—and a smarter way to proceed—is to combine randomized search and grid search: In a first step, randomized search can be used with a large hyperparameters grid, then, in a second step, exploiting the results that have been obtained, a focused hyperparameters grid around the best performing hyperparameters values can be established, and, after that, in a third step, a grid search can be run on this reduced hyperparameters grid, process that can be reapeted (until maximum computational/time budget is exceeded). Bayesian hyperparameters optimization could equally constitute an interesting way to follow (see, e.g., [1]).
- Thirdly, and lastly, more complex architectures for the forecasting process can be tried to reach better scores (e.g., using a voting regressor, or decomposing data domain used for training).

REFERENCES

[1] J. Bergstra, D. Yamins, and D. D. Cox. Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures. In *Proceedings of the 30th International Conference on Machine Learning*, 2013.

[2] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. 2016.

[3] Robert Dodier and Gregor Henze. Statistical analysis of neural networks as applied to building energy prediction. *Journal of Solar Energy Engineering*, 126:592–600, 2004.

[4] Jerome H. Friedman. Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 29:1189–1232, 2001.

[5] Aurélien Géron. *Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*, chapter Machine Learning Project Checklist. O'Reilly Media, 2017.

[6] J. S. Haberl and S. Thamilseran. The great energy predictor shootout ii: Measuring retrofit savings. *ASHRAE Journal*, 40:49–56, 1998.

[7] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. Lightgbm: A highly efficient gradient boosting decision tree. In *Advances in Neural Information Processing Systems*, pages 3146–3154. Curran Associates, Inc., 2017.

[8] Jan F. Kreider and Jeff S. Haberl. Predicting hourly building energy usage. *ASHRAE Journal*, 36:72–81, 1994.

[9] Scott M. Lundberg, Gabriel Erion, Hugh Chen, Alex DeGrave, Jordan M. Prutkin, Bala Nair, Ronit Katz, Jonathan Himmelfarb, Nisha Bansal, and Su-In Lee. From local explanations to global understanding with explainable ai for trees. *Nature Machine Intelligence*, 2:2522–5839, 2020.

[10] Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems 30*, pages 4765–4774. Curran Associates, Inc., 2017.

[11] David J.C. MacKay. Bayesian nonlinear modeling for the prediction competition. In *ASHRAE Transactions*, volume 100, pages 1053–1062, 1994.