Project Name: Skyscraper Energy Predictor

How much energy will a building consume?

Description

Q: How much does it cost to cool a skyscraper in the summer?
A: A lot! And not just in dollars, but in environmental impact.

Thankfully, significant investments are being made to improve building efficiencies to reduce costs and emissions. The question is, are the improvements working? That's where you come in. Under pay-for-performance financing, the building owner makes payments based on the difference between their real energy consumption and what they would have used without any retrofits. The latter values have to come from a model. 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.

In this project, you'll develop accurate models of metered building energy usage in the following areas: chilled water, electric, hot water, and steam meters. The data comes from over 1,000 buildings over a three-year timeframe. With better estimates of these energy-saving investments, large scale investors and financial institutions will be more inclined to invest in this area to enable progress in building efficiencies.

Evaluation Metric

The evaluation metric for this competition is Root Mean Squared Logarithmic Error.

The RMSLE is calculated as

$$\epsilon=1n\sum i=1n(\log(pi+1)-\log(ai+1))2$$

Where:

ϵ is the RMSLE value (score)
n is the total number of observations in the (public/private) data set,
pi is your prediction of target, and
ai is the actual target for i.
$\log(x)$ is the natural logarithm of x

Note that not all rows will necessarily be scored.

Submission File

For each id in the test set, you must predict the target variable. The file should contain a header and have the following format:

```
id,meter_reading
0,0
1,0
```

```
2,0
etc.
```

Data Description

Assessing the value of energy efficiency improvements can be challenging as there's no way to truly know how much energy a building would have used without the improvements. The best we can do is to build counterfactual models. Once a building is overhauled the new (lower) energy consumption is compared against modeled values for the original building to calculate the savings from the retrofit. More accurate models could support better market incentives and enable lower cost financing.

This project challenges you to build these counterfactual models across four energy types based on historic usage rates and observed weather. The dataset includes three years of hourly meter readings from over one thousand buildings at several different sites around the world.

Files

train.csv

- `building_id` - Foreign key for the building metadata.
- `meter` - The meter id code. Read as `{0: electricity, 1: chilledwater, 2: steam, 3: hotwater}`. Not every building has all meter types.
- `timestamp` - When the measurement was taken
- `meter_reading` - The target variable. Energy consumption in kWh (or equivalent). Note that this is real data with measurement error, which we expect will impose a baseline level of modeling error. UPDATE: as discussed **here**, the site 0 electric meter readings are in kBTU.

building_meta.csv

- `site_id` - Foreign key for the weather files.
- `building_id` - Foreign key for `training.csv`
- `primary_use` - Indicator of the primary category of activities for the building based on **EnergyStar property type definitions**
- `square_feet` - Gross floor area of the building
- `year_built` - Year building was opened
- `floor_count` - Number of floors of the building

weather_[train/test].csv

Weather data from a meteorological station as close as possible to the site.

- `site_id`
- `air_temperature` - Degrees Celsius

- `cloud_coverage` - Portion of the sky covered in clouds, in **oktas**
- `dew_temperature` - Degrees Celsius
- `precip_depth_1_hr` - Millimeters
- `sea_level_pressure` - Millibar/hectopascals
- `wind_direction` - Compass direction (0-360)
- `wind_speed` - Meters per second

  test.csv

  The submission files use row numbers for ID codes in order to save space on the file uploads. `test.csv` has no feature data; it exists so you can get your predictions into the correct order.
- `row_id` - Row id for your submission file
- `building_id` - Building id code
- `meter` - The meter id code
- `timestamp` - Timestamps for the test data period

  sample_submission.csv

  A valid sample submission.

- All floats in the solution file were truncated to four decimal places; we recommend you do the same to save space on your file upload.
- There are gaps in some of the meter readings for both the train and test sets. Gaps in the test set are not revealed or scored.