HD ENERGY

Detecting Faulty Energy Consumption

Programming Challenge for HD Energy

François St-Hilaire
July 18th 2025

Context

Buildings are major consumers of energy → monitoring usage is key for cost reduction and sustainability.

Early detection of abnormal energy consumption can:

- Help prevent system failures
- Reduce energy waste
- Optimize operational efficiency and eventually help businesses reach net zero through better performance monitoring.

In this project:

- We analyze daily energy consumption of a store in relation to outside air enthalpy
- The training data contains only **normal behavior**; the test data contains a mix of normal and **anomalous behavior**.

The goal is to **automatically detect abnormal energy consumption** using machine learning.

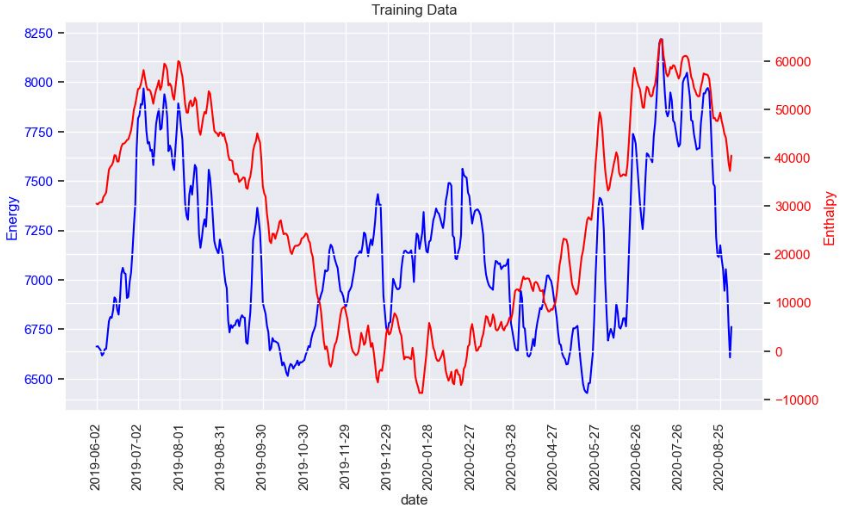
Assumptions and Simplifications

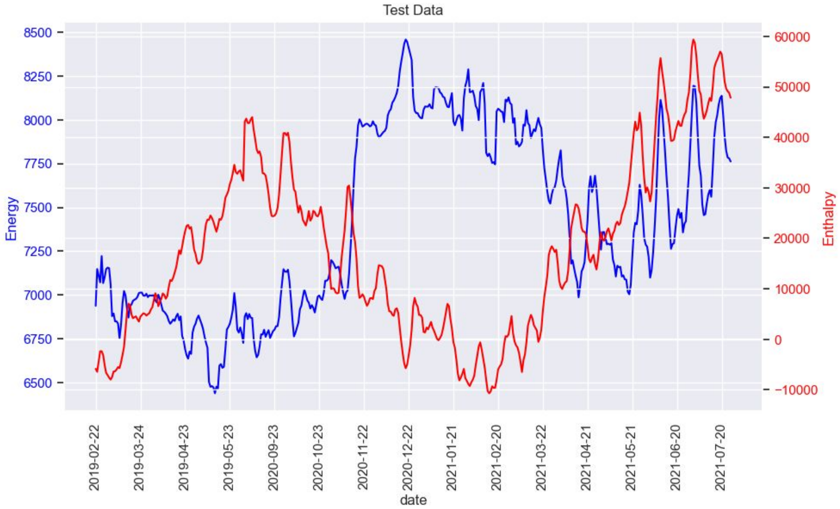
- All the data in the training set is considered normal (store working at an acceptable level)

- All factors unknown factors that could affect store energy consumption are considered constant throughout the data for now.

- Abnormally low energy consumption won’t be flagged. We are only interested in abnormally high energy consumption

Data Exploration - Energy Consumption in Time

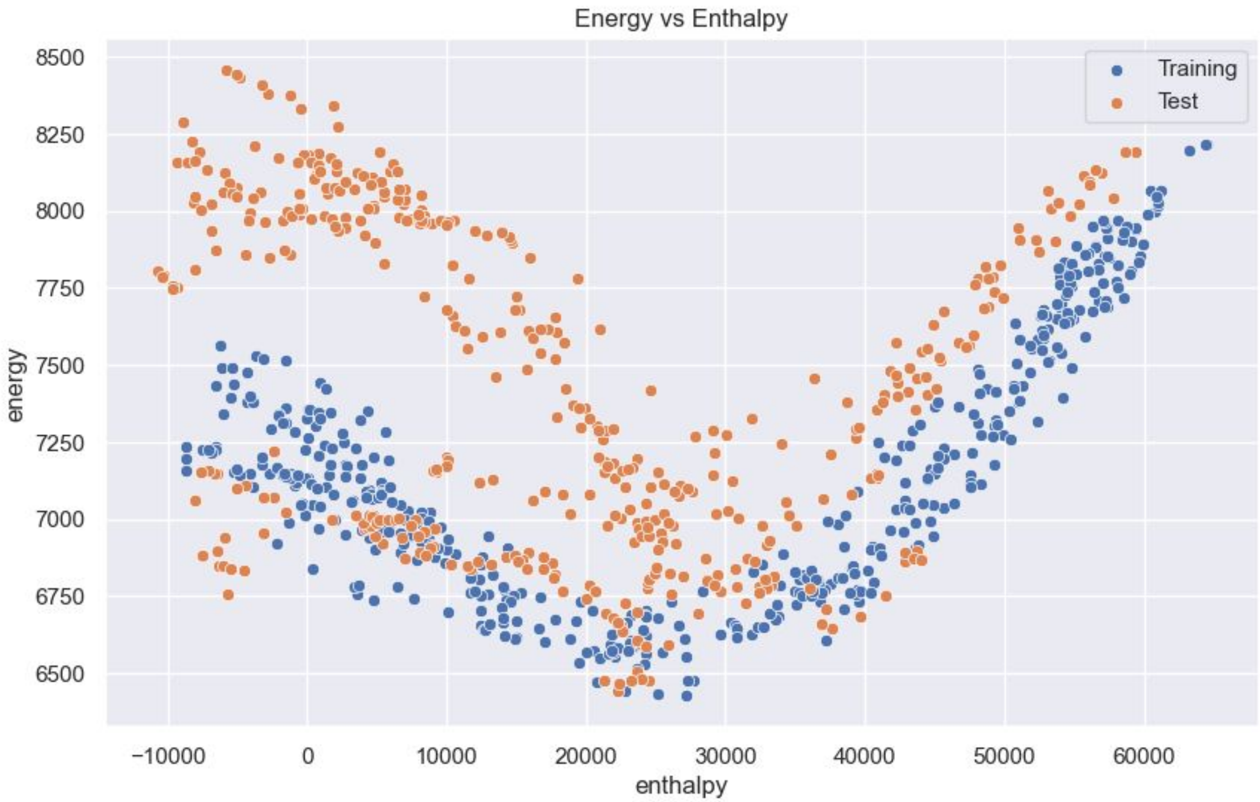




Training data: 459 points (date, enthalpy, energy)

Test data: 427 points (date, enthalpy, energy) [there's missing data from 2019-06-01 to 2020-09-03]

Data Exploration - Energy Consumption vs Enthalpy



- Looking at the training data, there is a nonlinear U-shape relationship between energy and enthalpy. This makes sense: at high outside temperature (high enthalpy) the store needs more energy to keep cool temperatures inside. At low outside temperature (low enthalpy), the store needs also more energy to keep the inside temperature adequate.

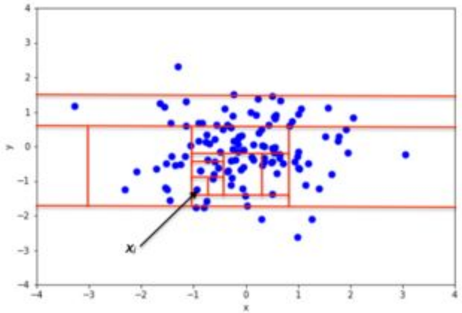
- We can already see that some of the training data follows a different relationship

Machine Learning Approach - Isolation Forest

Since we assume the whole training data is labeled as “normal”, this is a semi-supervised anomaly detection task

This can be solved with an **isolation forest**:

- Works well with few features (two in our case)
- Can be trained on only normal data
- Does not require labels for abnormal data
- Simple to implement and interpret

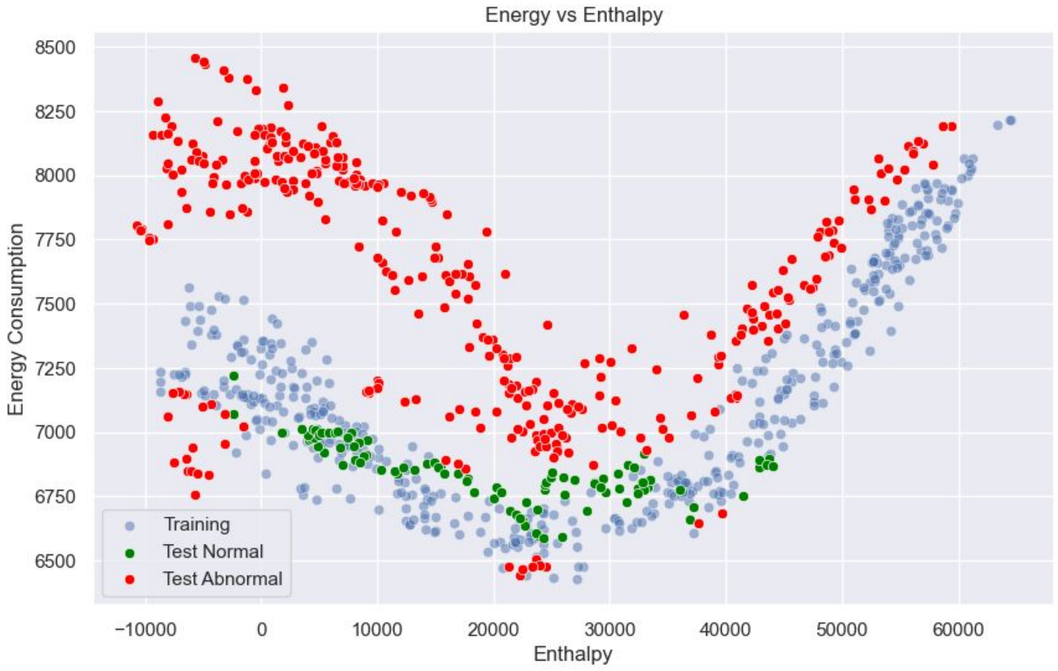


Visualization of Isolation Forest in 2D

Results - Initial Model Results

Model: `IsolationForest(contamination='auto', random_state=42)`

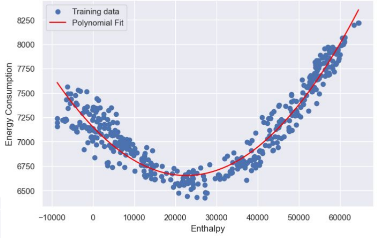
Since the number of anomalies in the test set is unknown, we set contamination = ‘auto’ for now



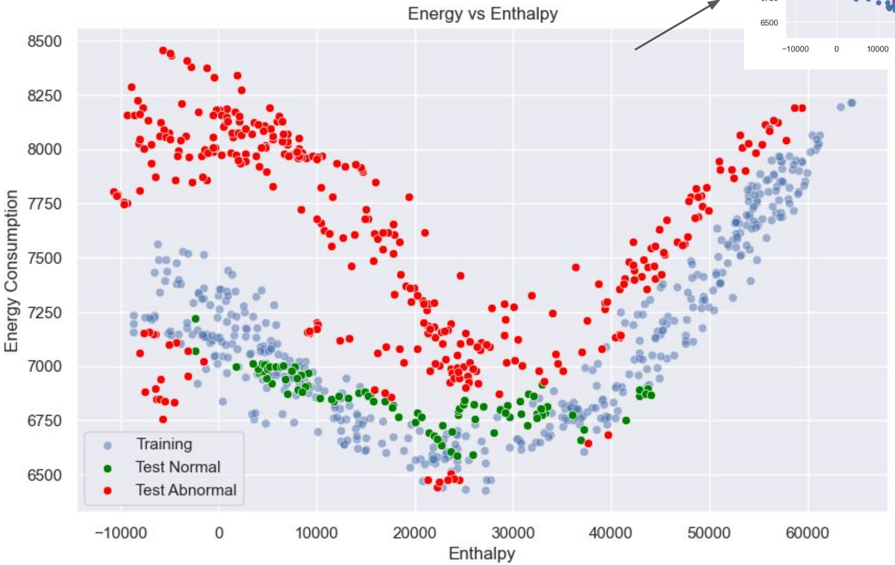
- The model seems to perform well at detecting anomalies relative to the training data
- Post-processing is needed to only flag the abnormally high energy consumption, and not the abnormally low

Results - Post-Processing with Polynomial Fit

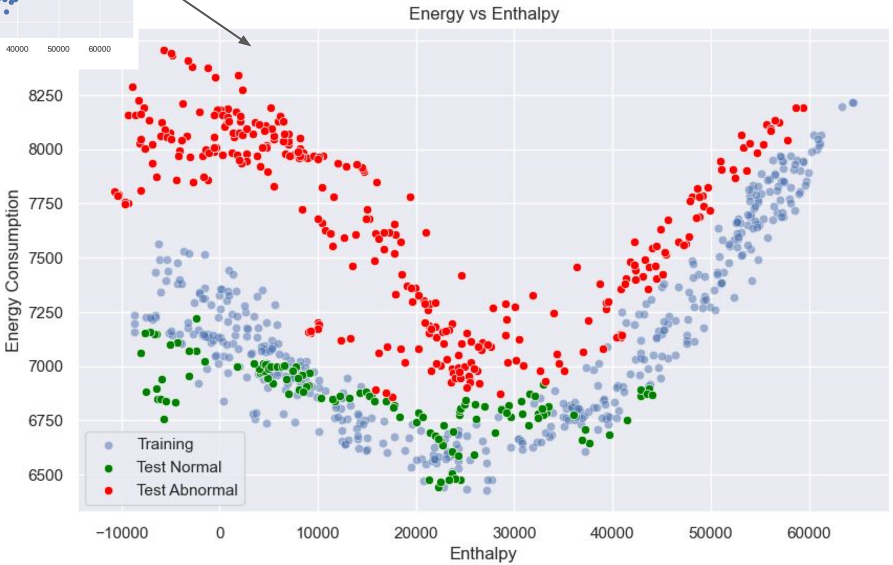
Justification: we want the final model to only flag energy consumption that is abnormally high. We can fit a polynomial through the training data and set any energy consumption lower that the fit as “normal”



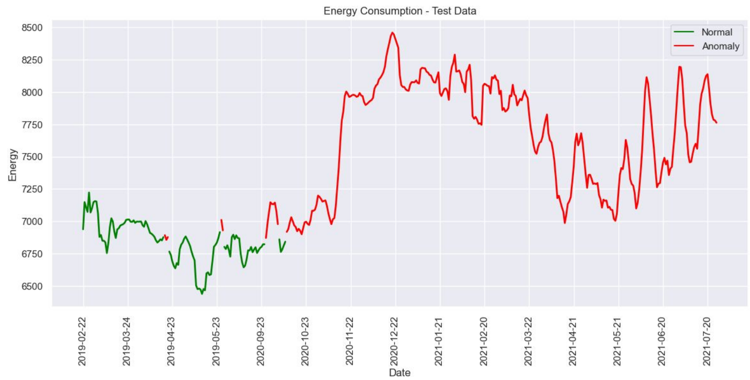
Initial Model

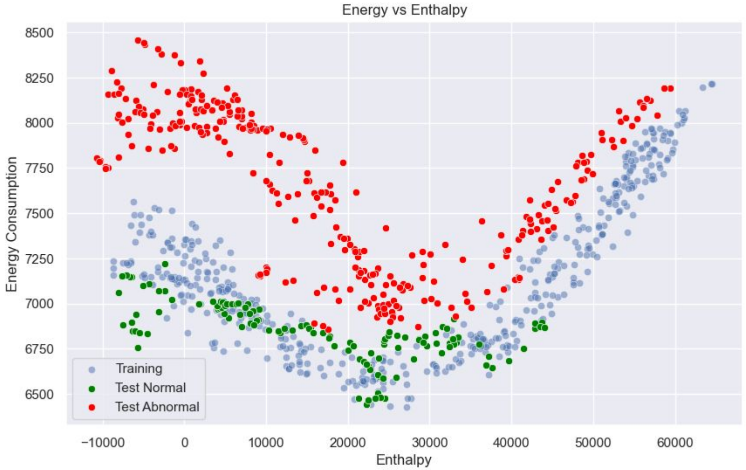


Final Model



Results - Assessment





- We can assess visually from the energy vs enthalpy scatter plot that the model does a good job of flagging energy consumptions that are higher than the normal training data

- The model could be further fine-tuned to be slightly less sensitive. It seems that some flagged points are very close to the edge of the "normal" cloud. This depends on the application and how sensitive we need the model to be.

- From the time series view, it seems that something occurred around 2020-09: all energy consumptions are abnormally high after that date

Conclusion and Next Steps

- Isolation forest model with polynomial fit for post-processing does a good job of detecting abnormally high energy consumption
- The model could be fine-tuned depending on the sensitivity needs

- **Next steps:** fine-tune model, test other options like one-class SVM, or simple polynomial fit with MSE,...