IISE PG&E Analytics Challenge

Team Optix

Columbia University

Millend Roy*, Vladimir Pyltsov*, Yinbo Hu, Agostino Capponi and Vijay Modi

October 26, 2025

INFORMS Annual Meeting 2025

Introduction

The setting is regressive: $y = f(X)$, where X represents exogenous variables (tabular data). The challenge assumes that sequential window lags are not available. Hence, our focus is quantifying the relationship between *Load* and exogenous variables. However, how do we actually model this relationship?

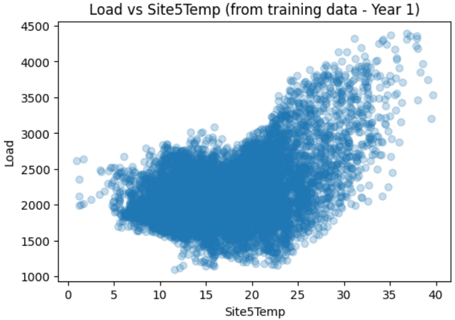

Figure: Temperature vs Load for Year 1 of training data.

Question: how do we make sense of the relationship presented above?

Model Intuition

We adopt the modeling approach from: **Hu, Yinbo** *et.al. "A data-driven approach for the disaggregation of building-sector heating and cooling loads from hourly utility load data." Energy Strategy Reviews 49 (2023)*.

Main idea: disaggregation of the data by hour, i.e., binning. Load responds to temperature: heating and cooling, while the temporal pattern is isolated.

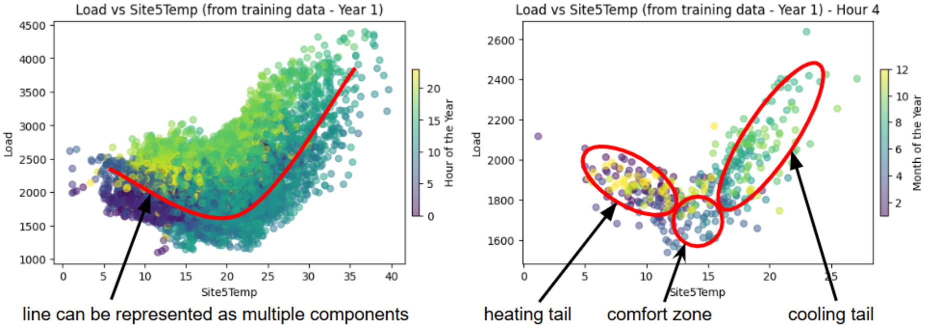

Figure: Temperature vs Load disaggregated for one hour.

Model - Step 1

Problem: No single site consistently correlates best with load. Including all site variables introduces high multicollinearity (checking Vairance Inflation Factor).

Feature Engineering using Principal Component Analysis (PCA):

- Applied separately to temperature and GHI variables.
- Just keeping the first principal component explains approximately 99% of total variance.
- Effectively replaces multiple correlated site features with low-dimensional, informative representations.

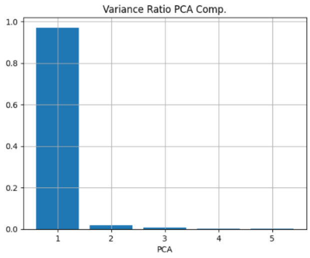

Figure: Variance Ratio PCA Components.

Model - Step 2

We define :

1. $y_{t,h}$ as the load at time t for hour h, where $h \in \{0, 1, 2, \ldots, 23\}$.
2. $\mathbf{X}_{t,h}$ represents the exogenous features, including PCA-transformed temperature and GHI.
3. $f_h(\cdot)$ is the function representing the predictive model trained for each specific hour h.

Each model f_h is trained independently on historical data:

$$y_{t,h} = f_h(\mathbf{X}_{t,h}) + \epsilon_{t,h} \tag{1}$$

where $\epsilon_{t,h}$ represents the residual error for the respective hour h.

Stacking Predictions: The final daily prediction sequence is constructed as:

$$\hat{Y}_t = \{\hat{y}_{t,0}, \hat{y}_{t,1}, \ldots, \hat{y}_{t,23}\} \tag{2}$$

$$\hat{y}_{t,h} = f_h(\mathbf{X}_{t,h}), \quad h \in \{0, 1, 2, \ldots, 23\}$$

This process is iteratively repeated for each day in the forecasting horizon, constructing a full-year prediction.

Model - Step 2

How can we fit the curve? We iterate through different classes of models.

- Classical models (segmented regression, polynomial) - easy and interpretable, but can be susceptible to outliers, require more preprocessing, hard to tailor for comfort zone (i.e., finding the breaking points)
- Tree-based models (Random Forest, XGBoost) - easy to use, have some degree of interpretability (i.e., Gini index), capture non-linear relationships.
- Probabilistic models (Gaussian Process) - estimates an uncertainty range rather than point estimates, but can be computationally heavy
- Sequential models (LSTM, Transformers, and their variants) - good for sequence modeling, but usually not used for regressive settings, can be computationally heavy and susceptible to hyperparameter optimization
- Pretrained/Foundation models (TimeGPT) - emerging paradigm, can give really accurate results, but expensive

Other aspects:

- We created multiple training cases for a rigorous comparison of models (Year 1 $\rightarrow$ Year 2, Year 2 $\rightarrow$ Year 1, Both years with 5 CV folds)
- Created temporal features and iterated exogenous lag features to explore their impact

Result Comparisons

All models achieve a reasonable MAPE/sMAPE error within 10%. This is a positive result considering the limited training data and a high prediction horizon.

		Models										
Metric	Test Case	Poly	XGBoost	RF	MLP	GP	LSTM	TF	NHITS	TCN	TFT	TimeGPT
R^2	Year 1 → Year 2	0.6	**0.8**	0.8	0.8	0.8	0.7	0.7	0.7	0.7	0.3	0.18
	Year 2 → Year 1	0.8	**0.9**	0.8	0.9	0.9	0.7	0.7	0.6	0.7	0.2	0.32
	Both years	-750e6	**0.7**	0.6	0.5	0.5	0.5	0.5	0.3	0.4	0.4	×
RMSE	Year 1 → Year 2	263.4	**159.6**	174.8	160.6	168.9	220.9	239.5	233.6	222.8	329.3	369.03
	Year 2 → Year 1	181.6	178.6	189.6	**170.7**	177.2	254.8	270.7	294.4	237.7	408.0	383.18
	Both years	12e6	**263.7**	286.5	312.5	308.1	311.6	309.9	368.5	334.9	348.7	×
MAPE	Year 1 → Year 2	6.9	**5.5**	5.9	5.6	5.9	7.5	8.3	8.3	7.6	11.5	14.31
	Year 2 → Year 1	5.7	**5.6**	5.9	5.8	5.9	8.5	8.9	9.3	8.0	13.1	14.79
	Both years	37e3	**7.4**	8.0	9.0	8.1	9.3	9.2	10.5	9.7	10.1	×
sMAPE	Year 1 → Year 2	7.0	**5.5**	5.9	5.6	6.0	7.5	8.2	8.2	7.6	10.9	13.19
	Year 2 → Year 1	5.7	**5.6**	5.8	5.7	5.8	8.4	8.8	9.3	7.9	12.8	13.63
	Both years	14.0	**7.7**	8.3	9.3	8.6	9.6	9.6	10.9	10.2	10.5	×

Our iterations show that tree-based models generally achieve more accurate results.

Thank you!

For questions and collaborations, please email:

millend.roy@columbia.edu or v.pyltsov@columbia.edu

Interesting Upcoming Models : TimeGPT Model by Nixtla

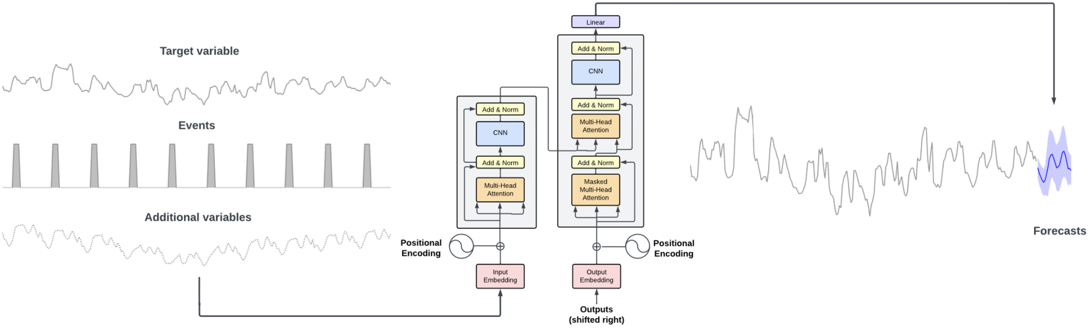

Figure: TimeGPT model

TimeGPT Results

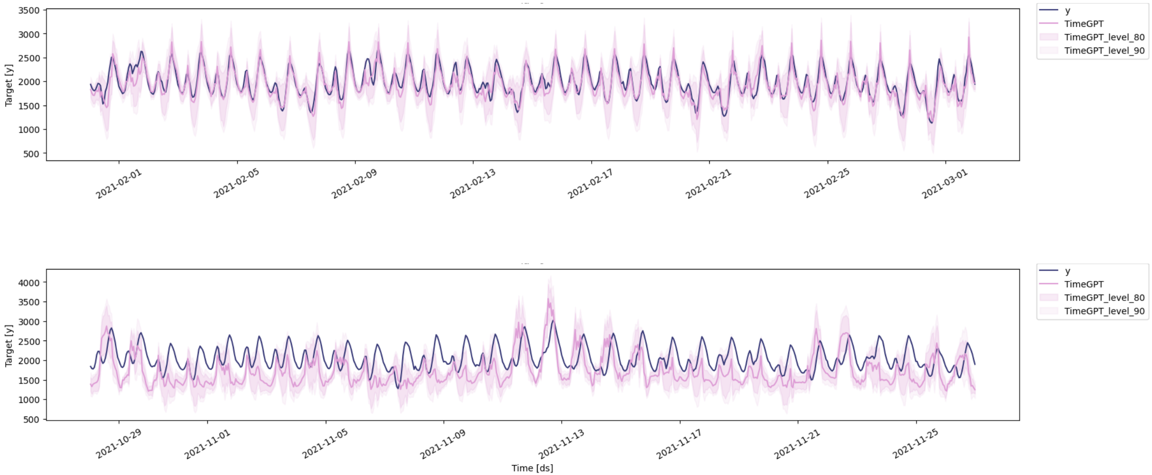

Figure: Long-Horizon TimeGPT-1 Forecasting