Leveraging Graph Neural Networks to Forecast Electricity Consumption

Context

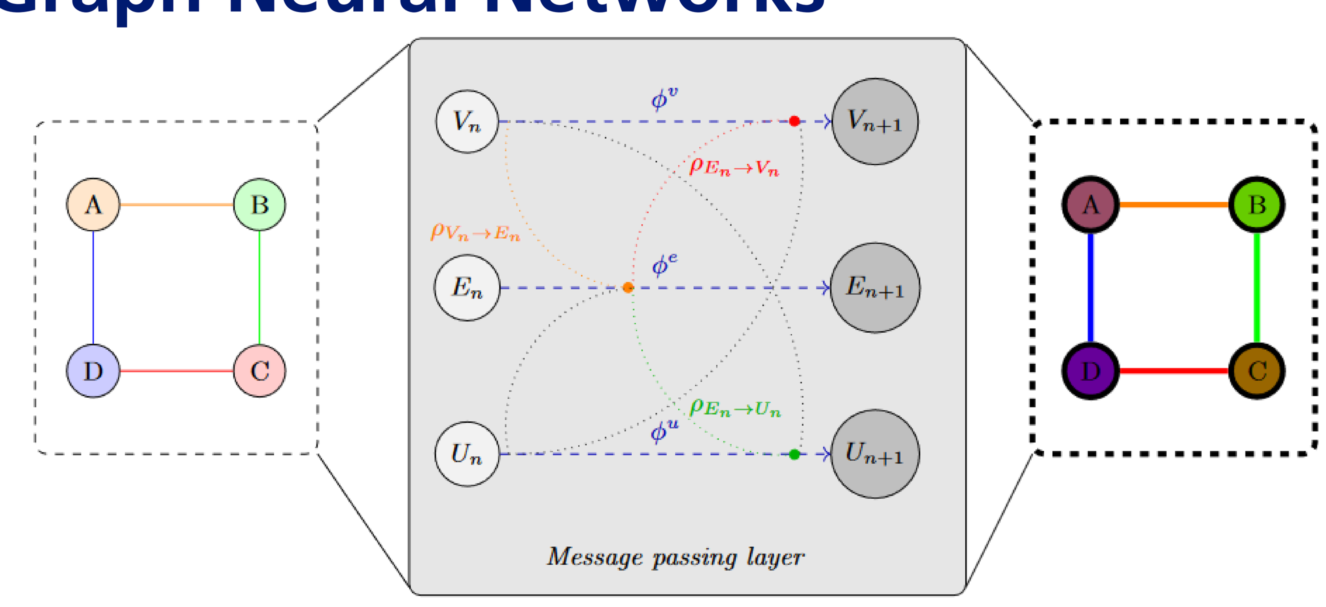
Accurate electricity demand forecasting is essential for several reasons, especially as the integration of renewable energy sources and the transition to a **decentralized network paradigm** introduce greater complexity and uncertainty. The proposed methodology leverages **graph-based representations** to effectively capture the **spatial distribution and relational intricacies** inherent in this decentralized network structure. We conduct experiments on electricity load forecasting, in both a **synthetic and a real framework** considering the **French mainland regions**.

Notations

Denote by

- $\mathcal{N}_v$ the neighborhood of node v,
- $h_v^{(k)}$ the representation vector of node v at iteration k,
- $\boldsymbol{X}_v$ the feature vector of node v,
- $\boldsymbol{W}$ the learned weight matrices, and $\boldsymbol{b}$ the learned bias vector.

Graph Neural Networks



$$h_v^{(k)} = \mathrm{UPDATE}^{(k)}\left(h_v^{(k-1)}; \mathrm{AGGREGATE}^{(k)}\left(h_v^{(k-1)}; \left\{h_u^{(k-1)} \mid u \in \mathcal{N}_v\right\}\right)\right)$$

$$h_v^{(0)} = \boldsymbol{X}_v$$

Classical GNN models

GCNs were introduced by *Kipf and Welling* in 2016 and aim at **learning representations**. The update rule of a GCN is given by:

$$h_i^{(0)} = \boldsymbol{X}_i,$$

$$h_i^{(l+1)} = \sigma\left(\sum_{j \in \mathcal{N}_i} \frac{1}{c_{ij}} \boldsymbol{W}^{(l)} h_j^{(l)} + \boldsymbol{b}\right)$$

$$\text{where } c_{ij} = \sqrt{|\mathcal{N}_i||\mathcal{N}_j|}$$

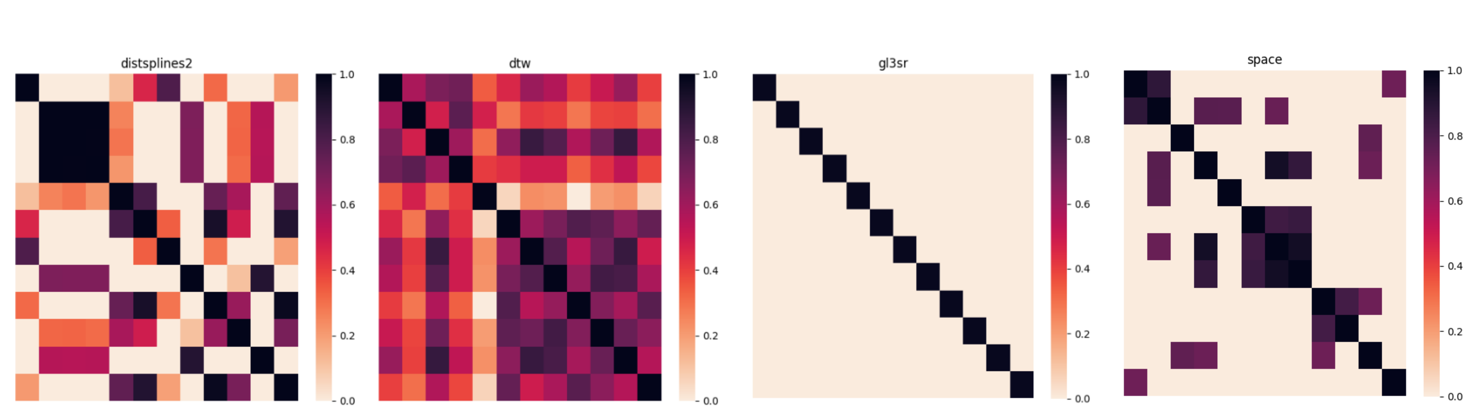
SAGE was introduced by *Hamilton* in 2018 and aim at learning **aggregation functions**. The learned aggregation function is given by:

$$h_i^{(l+1)} = \sigma\left(\mathbf{W}^{(l)}\left[h_i^{(l)} || \max\left\{\sigma\left(\mathbf{W}_{\text{pool}} h_j^{(l)} + \mathbf{b}\right), \forall v_j \in \mathcal{N}_i\right\}\right]\right)$$

where $\mathbf{W}_{\text{pool}}$ is a learnable pooling weight matrix.

Inferring Graphs from Data

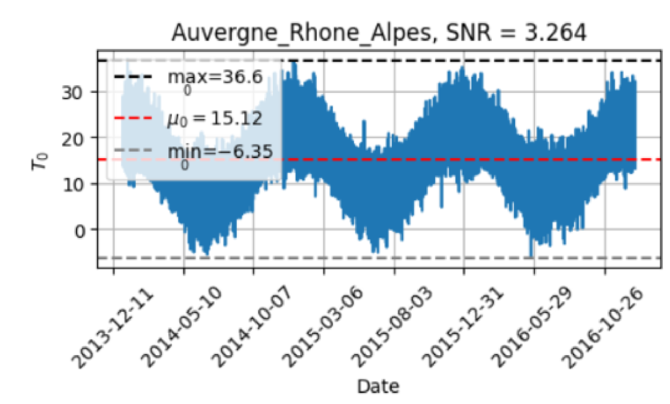
- **Geographical Data**: we compute a similarity matrix based on the geographical positions of the nodes without considering the topography
- **Electricity & Weather Data**: we first project the d-dimensional signal into a 1-dimensional space and then we can apply algorithms (DTW, GL3SR), distances…



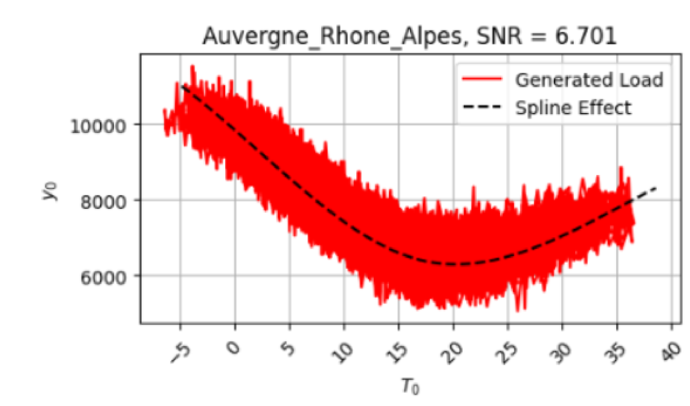
Datasets

To analyze our models and the contribution of GNNs when there is an **underlying graph structure**, we generated datasets **with and without links** between regions.

We **fitted some splines** based on the **observed temperatures and loads**, and then **we evaluated the learned splines** with some **generated temperatures**, to which we added a **multivariate centered random noise** to represent the links between the regions.



(a) Temperature generated in Auvergne-Rhône-Alpes.



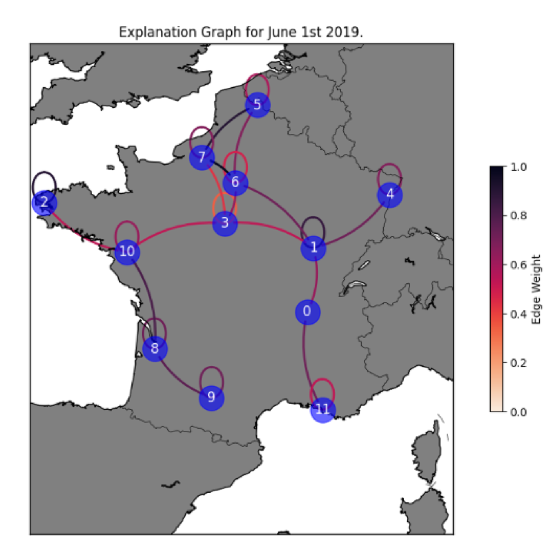
(b) Load generated with a cubic spline basis of rank 10 in Auvergne-Rhône-Alpes.

Explainability

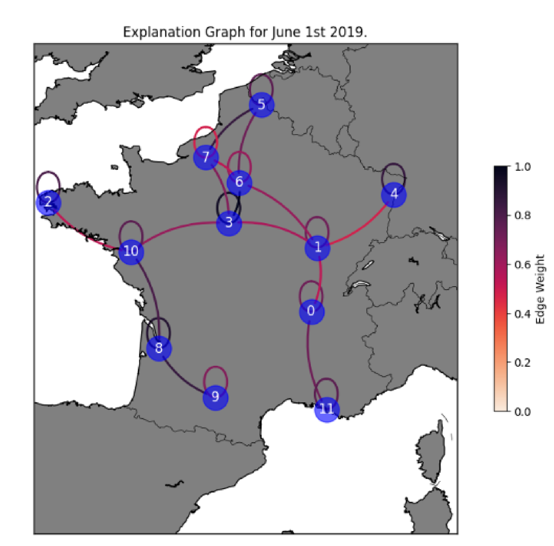
We want to be able to extract important sub-graphs to find links between regions. To do this, we used **GNNExplainer**, developed by *Ying et al.*, which calculates a mask on the edges of the graph by maximizing a mutual information criterion:

$$\max_{\mathcal{G}_S} \mathrm{MI}\left(\boldsymbol{Y}_{\mathcal{G}}, \mathcal{G}_S\right) = H\left(\boldsymbol{Y}_{\mathcal{G}}\right) - H\left(\boldsymbol{Y}_{\mathcal{G}} \mid \mathcal{G} = \mathcal{G}_S, \mathbf{X} = \mathbf{X}_S\right)$$

where $\mathcal{G}_S$ and $\mathbf{X}_S$ are the explaining subgraph and features, H is the entropy, and $\boldsymbol{Y}_{\mathcal{G}}$ is the vector of predictions made with graph $\mathcal{G}$.



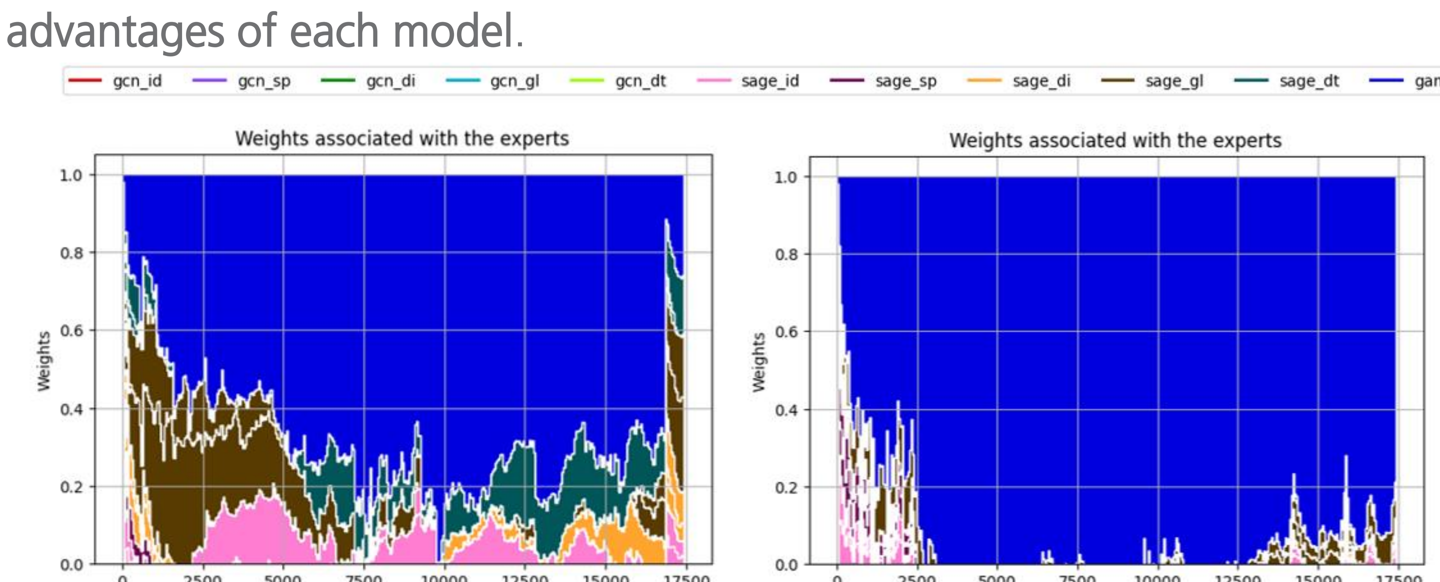
(a) Synthetic dataset ($\boldsymbol{\Sigma} = \rho(\mathbf{W}_\lambda)$).



(b) Real dataset.

Expert Aggregation

Several models have been developed in the literature, each with its **own distinctive features**, which may also **complement** each other. Expert aggregation is an ensemble technique that allows to **benefit from the advantages of each model**.



Regardless of the dataset, GAM remains **dominant in aggregation**, but GNNs prove particularly useful in the **dataset with information between regions**.

Results

For each model/matrix pair, we ran a **grid-search hyperparameter optimization**, then retained the best model on the **validation set**. GNNs bring **diversity** to the aggregation when datasets contain **information between regions**. Moreover, **SAGEs perform better** on average than **GCNs**.

Model	Real Dataset MAPE (%)	Real Dataset RMSE (MW)	Synthetic Dataset ($\boldsymbol{\Sigma} = \rho(\mathbf{W}_\lambda)$) MAPE (%)	Synthetic Dataset ($\boldsymbol{\Sigma} = \rho(\mathbf{W}_\lambda)$) RMSE (MW)	Synthetic Dataset ($\boldsymbol{\Sigma} = \boldsymbol{I}$) MAPE (%)	Synthetic Dataset ($\boldsymbol{\Sigma} = \boldsymbol{I}$) RMSE (MW)
GAM-Regions	**1.48**	**1018**	**1.11**	**662**	**1.75**	**1043**
Feed Forward	1.54	1071	3.82	3141	4.49	3213
GCN-identity	5.66	3949	1.43	834	2.16	1259
GCN-space	2.07	1452	1.26	749	1.98	1169
GCN-distsplines	2.04	1404	1.29	764	2.01	1185
GCN-gl3sr	5.95	4210	**1.25**	**743**	**1.97**	**1160**
GCN-dtw	**1.82**	**1276**	1.26	753	1.99	1171
SAGE-identity	4.38	3021	1.25	755	**1.78**	**1066**
SAGE-space	1.96	1350	1.29	778	1.85	1112
SAGE-distsplines	2.06	1410	1.22	741	1.84	1116
SAGE-gl3sr	**1.78**	**1234**	**1.15**	**701**	1.92	1171
SAGE-dtw	1.90	1335	1.21	735	1.86	1127
Mixture (Baseline)	1.31	925	1.11	662	1.76	1044
Mixture (GNNs)	1.48	1092	1.14	683	1.98	1171
Mixture (Baseline + GNNs)	**1.13**	**844**	**1.10**	**661**	**1.76**	**1050**

Contacts

CAMPAGNE Eloi, eloi.campagne@edf.fr AMARA-OUALI Yvenn, yvenn.amara-ouali@edf.fr GOUDE Yannig, yannig.goude@edf.fr KALOGERATOS Argyris, argyris.kalogeratos@ens-paris-saclay.fr