Forecasting Net Load in France:
The EDF Data Challenge

Eloi Campagne, Yvenn Amara-Ouali, Yannig Goude, Argyris Kalogeratos
EDF R&D, Centre Borelli (ENS Paris-Saclay)

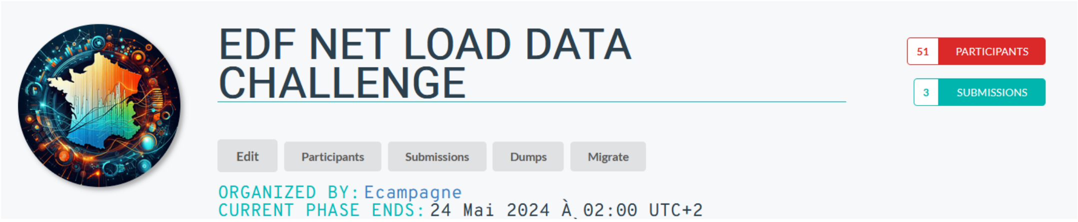
Outline

- Designing a Challenge
- Subject & Data
- Approaches
- Solutions
- Perspectives

Designing a Challenge (1/4)

- « **CodaBench** is an open-source platform for hosting data science challenges, benchmarks and competitions »

Codabench



- **130+ participants** from EDF Group in 46 teams, on **virtual machines** provided by EDF
- Starting kits and GPU available
- **Animate a data-science community** at EDF, get people working on a **subject of interest**

Designing a Challenge (2/4)

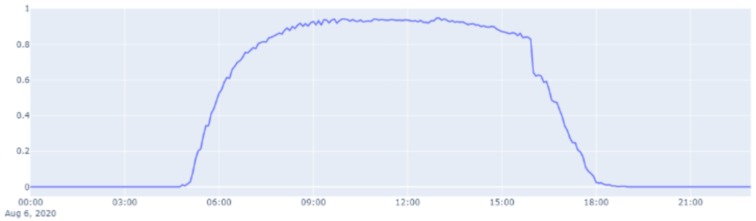
Motivations

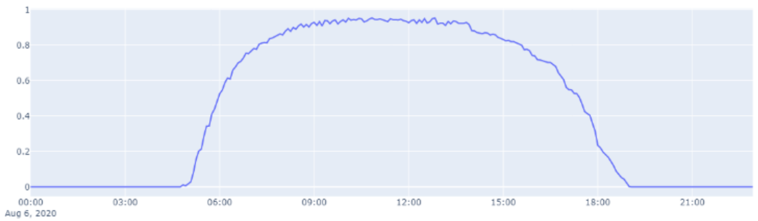
- Maintaining a **balance between electricity supply and demand** is important for **grid stability**
- Providing **accurate forecasts** for short-term electricity load is therefore crucial for all participants in the energy market
- The availability of new **geolocalized data and individual electricity consumption data** can be exploited by models that are able to take advantage of additional information and help in **minimizing forecast error**

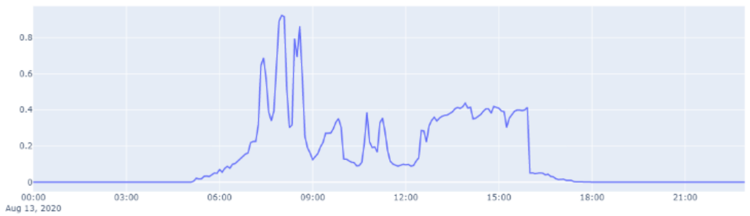
Designing a Challenge (3/4)

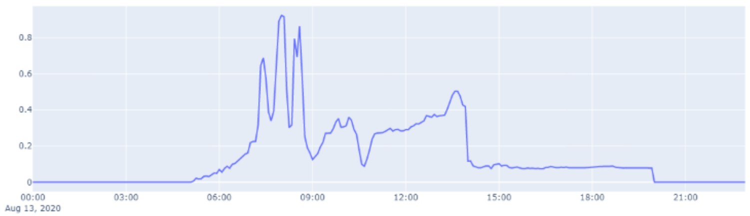
- The increasing contribution of **renewable energy sources brings fluctuations and intermittency** to the electricity market

  ➢ Complex representation of meteorological variables such as clouds



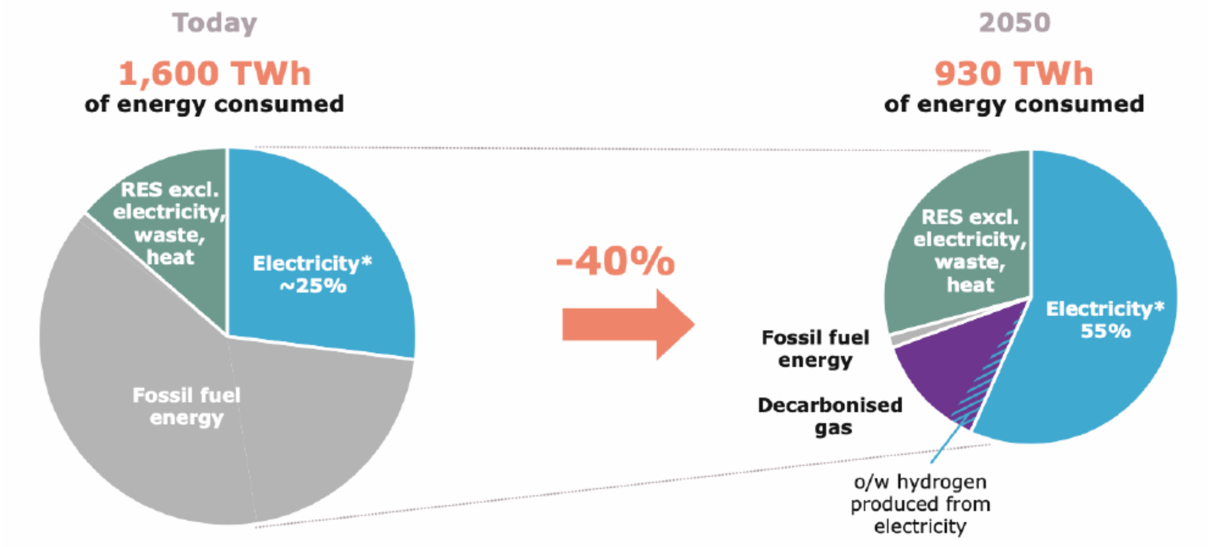


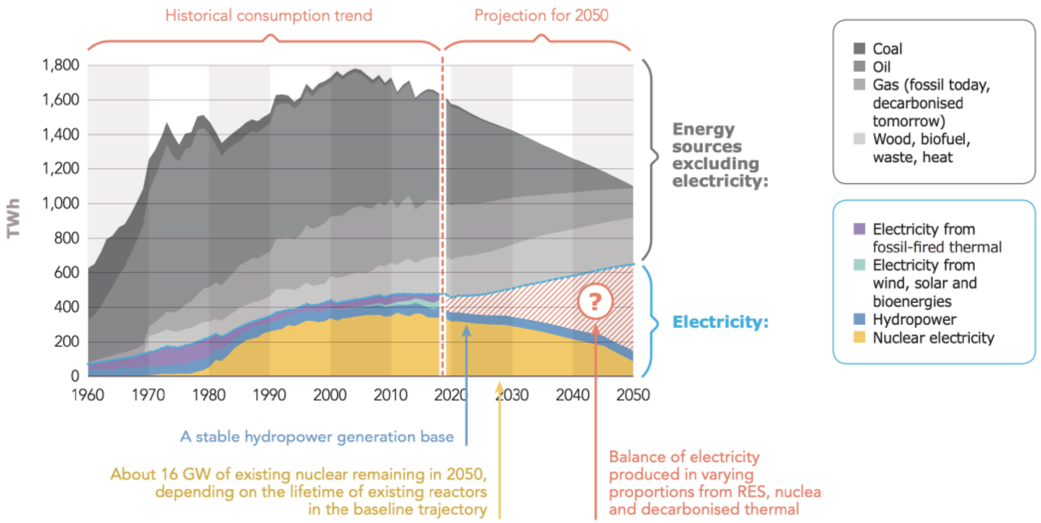




  ➢ Installed capacity is not precisely known and production measures are imperfect

Designing a Challenge (4/4)





Subject & Data (1/3)

Objective

- Develop methods that can take into account regional data to forecast **daily minimum and maximum net load* over France**

Perimeter

- Mainland France (excluding Corsica)

Challenges

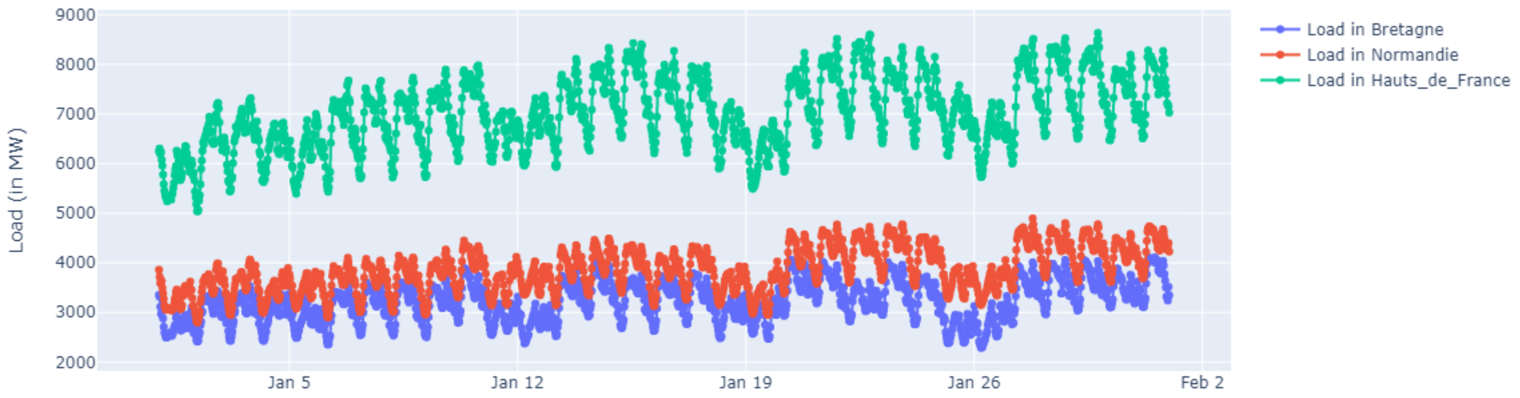
- Model renewable electricity production as accurately as possible
- Use data at different scales: regional and national

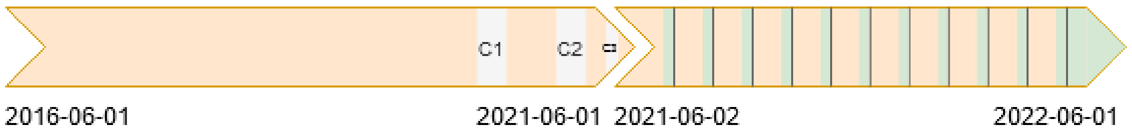
*Note that net load is defined as the difference between demand and renewable generation.

Subject & Data (2/3)

3 datasets: regional, national and price

- 12 administrative regions of France are considered
- 32 weather stations (appearing as black dots)
- Half-hourly data

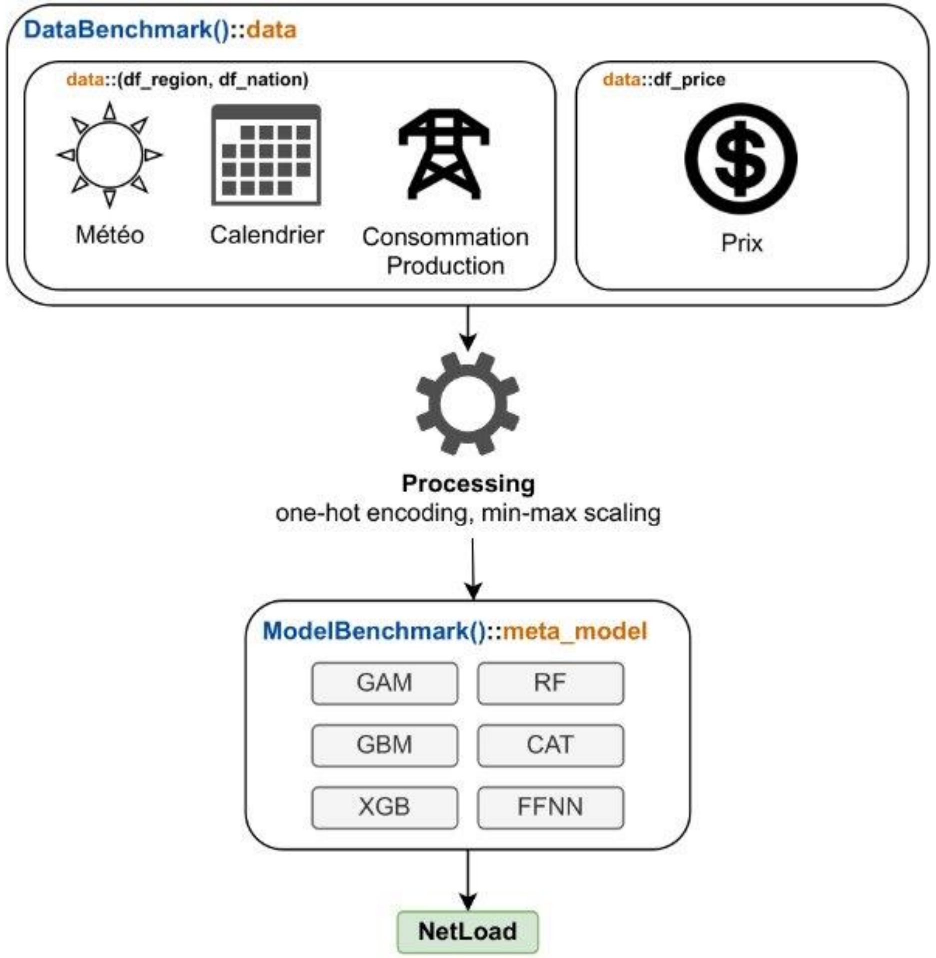




Training and test periods

- Train = orange, test = green
- Test period = last week of each month + May 2022
- Covid containment periods (grey) have been invalidated

Subject & Data (3/3)



Features

- Climatic, calendar and load data
- Spot price of electricity

Processing

- One-hot encoding
- Min-max scaling per region

Approaches

Geolocalized Data

- A natural approach consists in **predicting the national net load using national data**
- However, some **regions do not behave in the same way (weather, economy,…)** which can be hidden in a national forecast
- Aggregation of the regional forecasts can be done with a simple sum, but there are connexion problems

Net Load, Load – Production, Load – (Solar Power + Wind Power)?

- In practice, predicting all the **individual components give better results**

Mean predictions or Extreme predictions?

- **Small or large resolution**: predicting the mean and then taking the extremes or directly predicting the extremes (Scaled Student, Generalized Extreme Value Distribution,…)?
- Number of data points is reduced when directly trying to predict the extremes
- Best approach is to consider a multi-resolution prediction: **Amara-Ouali, Y., Fasiolo, M., Goude, Y., & Yan, H. (2022). Daily peak electrical load forecasting with a multi-resolution approach. International Journal of Forecasting**

Solutions (1/3)

Evaluation

- Participants were evaluated using the following loss:

$$\ell(y,\hat{y}) = \sqrt{\frac{1}{n}\sum_{d=1}^{n}\left(\min y_{48d:48(d+1)} - \min \hat{y}_{48d:48(d+1)}\right)^2} + \sqrt{\frac{1}{n}\sum_{d=1}^{n}\left(\max y_{48d:48(d+1)} - \max \hat{y}_{48d:48(d+1)}\right)^2}$$

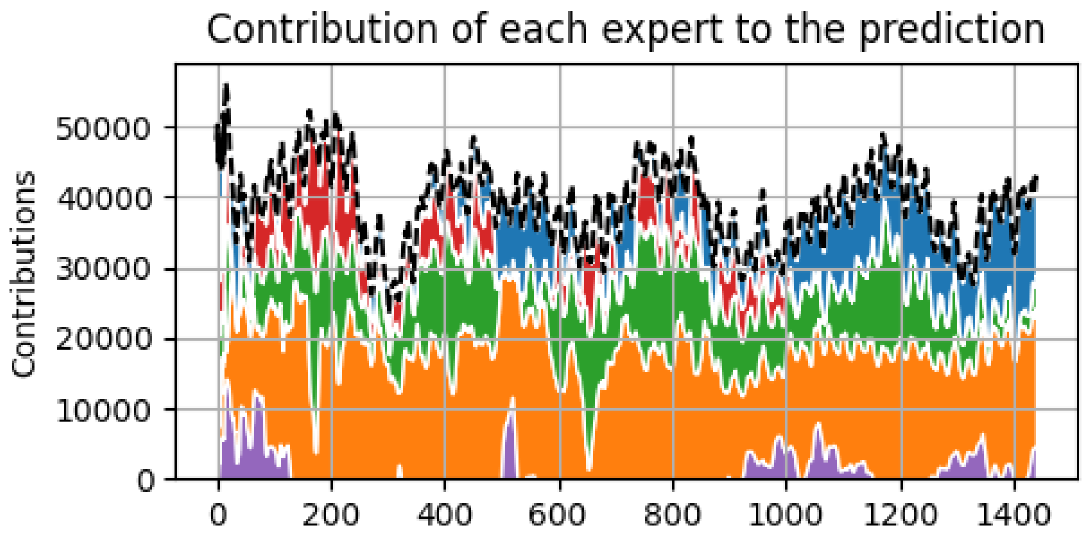
- The loss function is **the sum of the RMSE on the min and the max predictions** of all days (a day = **48 instants**)

Benchmark models

Participants were competing against basic benchmark models (trained on the RMSE of the mean predictions) to help them evaluate the quality of their own models:

- **GAM-1**: a GAM (Generalized Additive Model) forecasting the **net load**
- **GAM-3**: a GAM forecasting the **load**, the **wind production** and the **solar production**
- **CAT & xGB**: a CatBoost model & a xGBoost model
- **FF:** a Feed-Forward neural network model
- **Mixture**: a mixture of the above models

Solutions (2/3)



Benchmark models

- In average, **GAMs are the best performing** individual experts
- **ML-Poly aggregation outperform** the individual estimators on both the **validation and test sets** (except for GAM-3)
- Weights are **optimized on a validation set** and then **frozen for the test set**

Model	Loss (MW) Public Test	**Loss (MW) Private Test**	Loss (MW) Average Test
GAM-1	6226	8304	7265
GAM-3	**3946**	**4929**	**4438**
xGB	14364	14383	14374
CAT	18969	19569	19269
FF	14824	13342	14083
Aggregation ML-Poly	*4805*	*5091*	*4948*
Aggregation Uniform	9200	8823	9012

Solutions (3/3)

Participants models

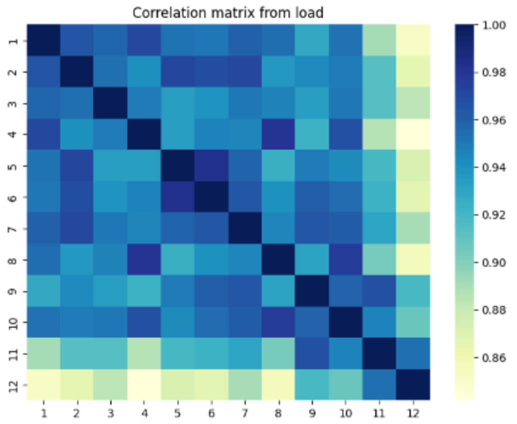
- **14 teams**/46 did a better score than the baseline on the private test set
- The best solutions **compute new features**, perform a **cross validation with multiple models** while **tuning their hyperparameters**, and select the **best model in average** on the folds
- The Final Strike used a **Light Gradient Boosting Machine Regressor (LGBMR)**
- Les Equilibristes used **two xGB models**, one for the load, and one for the production
- The Data Rangers aggregated a **GAM per instant** (7%), an **xGB** model (80%) and a **Bidirectional Recurrent Neural Network per instant** (**BRNN**, 13%)

Rank	Teams	Model	Loss (MW) Public Test	**Loss (MW) Private Test**	Loss (MW) Average Test
1	**Final Strike**	**LGBMR**	**2580**	**2747**	**2664**
2	*Les Equilibristes*	*xGB*	3352	*3040*	3196
3	Data Rangers	Mixture	2994	3098	*3046*
4	Les Green Code	xGB	3153	3172	3163
5	Les Syracusains	GAM	*2845*	3257	3051
6	Kung Fu Pandas 2	Mixture	3093	3370	3231
7	SoDataSum	GAM	3251	3442	3347
…	…		…	…	…
15	Baseline	GAM-3	3946	4929	4438

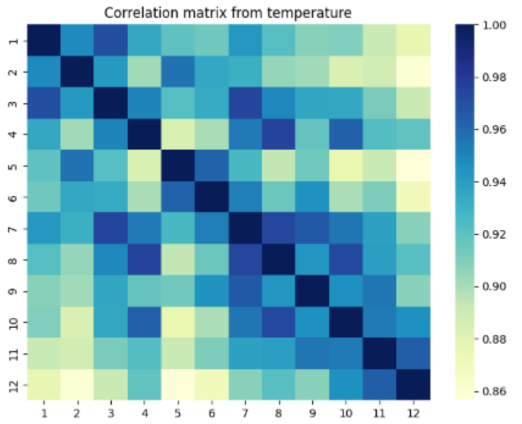
Perspectives (1/3)

Graph Neural Networks

- **CIFRE thesis** between EDF and Centre Borelli (Y. Amara-Ouali, A. Kalogeratos, M. Mougeot)
- GNNs can **handle spatial data using graph structures**
- We want to make use of the deep relationships that exist between the regions as their **features are strongly correlated**
- GNNs effectively **compute representations through convolutions** using the relationships within the data:
  - **Spatial** convolutions (adjacency matrix)
  - **Spectral** convolutions (Laplacian matrix)



(a) Correlation matrix obtained from consumption data.



(b) Correlation matrix obtained from temperature data.

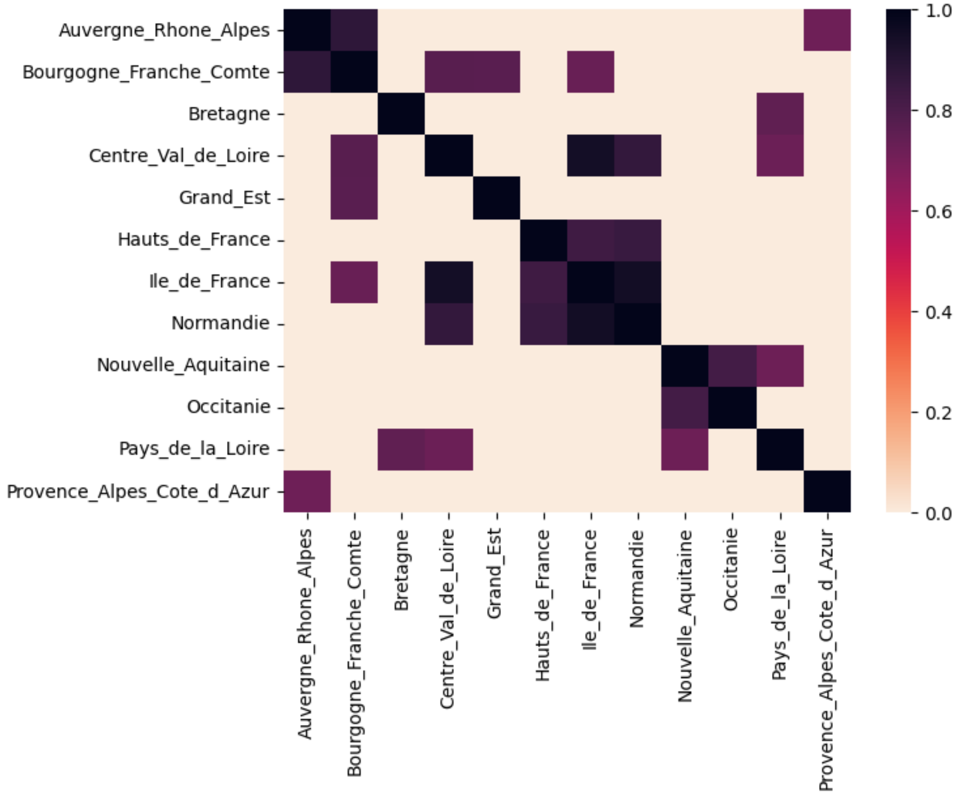
$$h_v^{(0)} = X_v,$$
$$h_v^{(k)} = \mathbf{UPDATE}^{(k)}\left(h_v^{(k-1)}; \mathbf{AGGREGATE}^{(k)}\left(\left\{h_u^{(k-1)} \mid u \in \mathcal{N}_v\right\}\right)\right)$$

Spatial

$$h^{(0)} = X,$$
$$h^{(k)} = \sigma\left(g_\theta \star_G h^{(k-1)}\right) = \sigma\left(g_\theta(\mathbf{L})h^{(k-1)}\right) = \sigma\left(\sum_{\ell=0}^{N-1} \theta_\ell \mathbf{L}^\ell h^{(k-1)}\right)$$

Spectral

Perspectives (2/3)



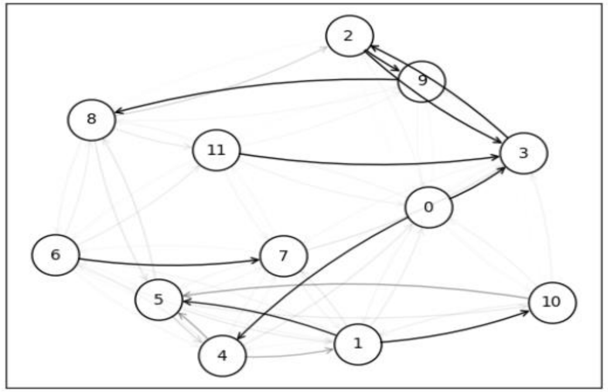
Graph structures

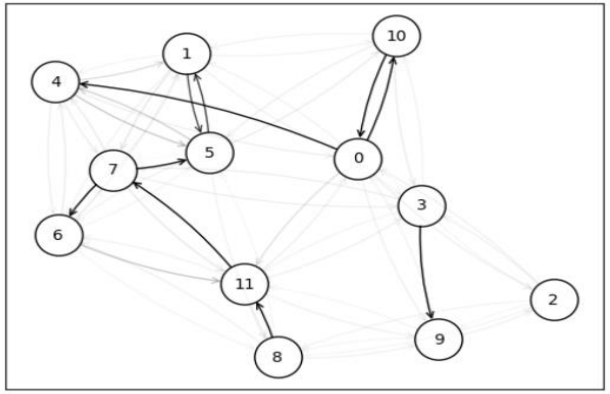
- **Statistics oriented**: correlation and precision matrices
- **Distance oriented**: DTW and Exponential similarity matrices
- **Time dependent structures**

Explainability

- Making **explainable forecasts** is a crucial point for EDF, GAM models **give both good results and interpretable forecasts**
- GNNs can highlight links between nodes and **therefore important subgraphs can be extracted**:

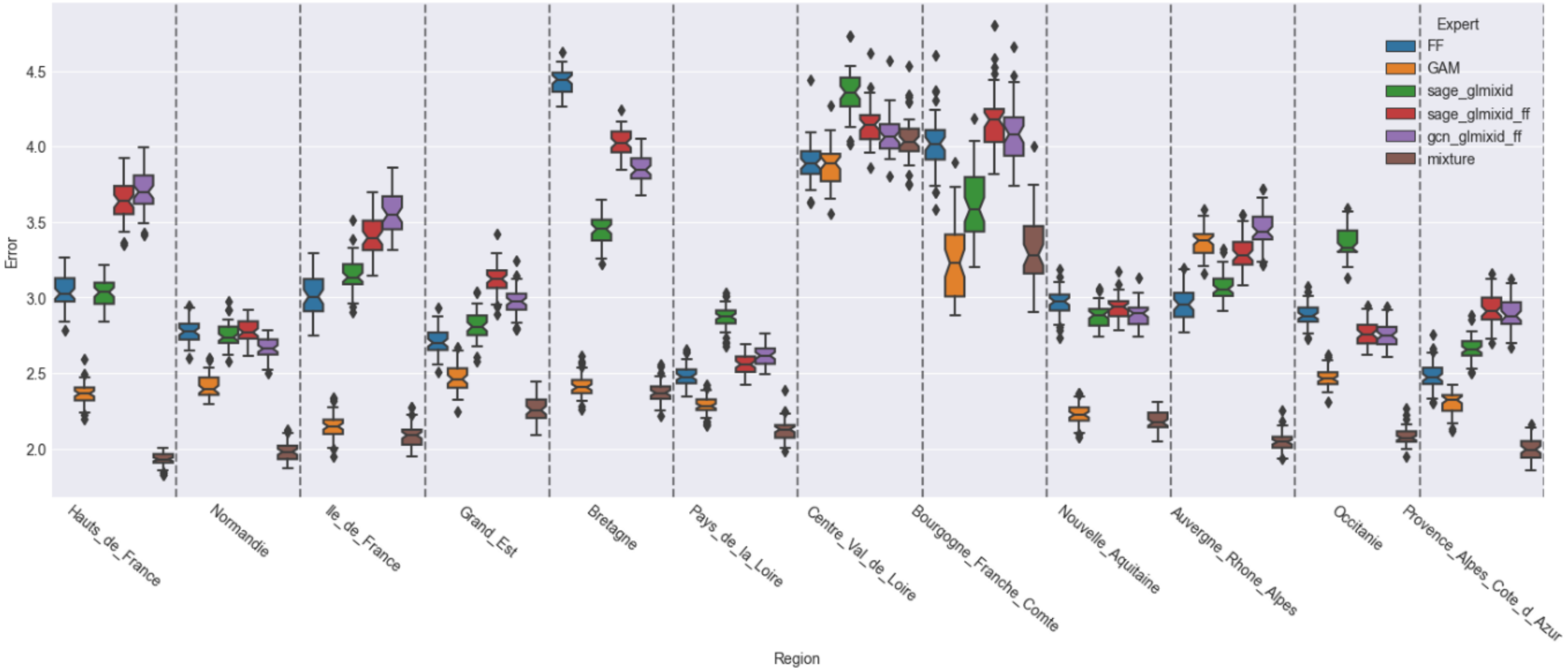
$$\max_{G_S} \mathbf{MI}(Y_G, G_S) \stackrel{\text{def}}{=} H(Y_G) - H(Y_G \mid G = G_S,\ X = X_S)$$





Perspectives (3/3)

- **Promising results** on the load



References

Alasali, F., Nusair, K., Alhmoud, L., and Zarour, E. (2021). Impact of the covid-19 pandemic on electricity demand and load forecasting. *Sustainability*, 13(3):1435.

Antoniadis, A., Gaucher, S., and Goude, Y. (2022). Hierarchical transfer learning with applications for electricity load forecasting.

Brégère, M. and Huard, M. (2022). Online hierarchical forecasting for power consumption data. *International Journal of Forecasting*, 38(1):339–351.

Cesa-Bianchi, N. and Lugosi, G. (2006). *Prediction, learning, and games*. Cambridge university press.

Chen, T. and Guestrin, C. (2016). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16. ACM.

Daigavane, A., Ravindran, B., and Aggarwal, G. (2021). Understanding convolutions on graphs. *Distill*, 6(9):e32.

de Vilmarest, J. and Goude, Y. (2021). State-space models win the ieee dataport competition on post-covid day-ahead electricity load forecasting.

Gaillard, P., Stoltz, G., and Van Erven, T. (2014). A second-order bound with excess losses. In *Conference on Learning Theory*, pages 176–196. PMLR.

Hamilton, W. L., Ying, R., and Leskovec, J. (2018). Inductive representation learning on large graphs.

Hansen, N., Müller, S. D., and Koumoutsakos, P. (2003). Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (cma-es). *Evolutionary computation*, 11(1):1–18.

Hastie, T. and Tibshirani, R. (1986). Generalized additive models (with discussion), statistical science.

Hastie, T. J. (2017). Generalized additive models. In *Statistical models in S*, pages 249–307. Routledge.

Kipf, T. N. and Welling, M. (2017). Semi-supervised classification with graph convolutional networks.

LeCun, Y., Bengio, Y., et al. (1995). Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, 3361(10):1995.

Obst, D., de Vilmarest, J., and Goude, Y. (2021). Adaptive methods for short-term electricity load forecasting during covid-19 lockdown in france. *IEEE Transactions on Power Systems*, 36(5):4754–4763.

Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., and Gulin, A. (2019). Catboost: unbiased boosting with categorical features.

Waikhom, L. and Patgiri, R. (2022). A survey of graph neural networks in various learning paradigms: methods, applications, and challenges. *Artificial Intelligence Review*.

Wood, S. N., Pya, N., and Säfken, B. (2016). Smoothing parameter and model selection for general smooth models. *Journal of the American Statistical Association*, 111(516):1548–1563.

Thank you!

Eloi Campagne, Yvenn Amara-Ouali, Yannig Goude, Argyris Kalogeratos

EDF R&D, Centre Borelli (ENS Paris-Saclay)

Annexes (1/3)

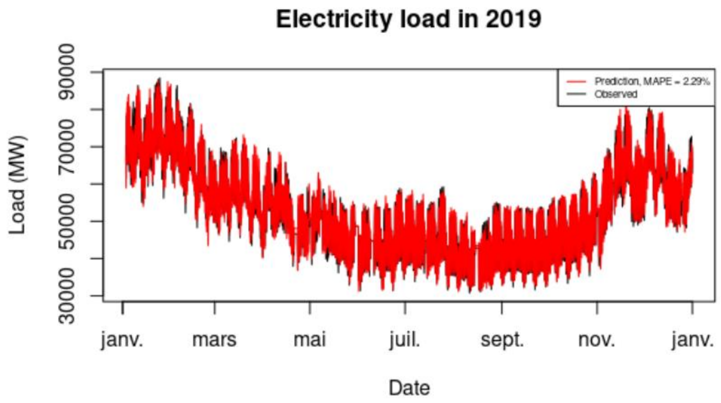
Generalized Additive Models

$$y_t = X_t \beta_0 + \sum_{j=1}^{d} f_j(x_{t,j}) + \varepsilon_t$$

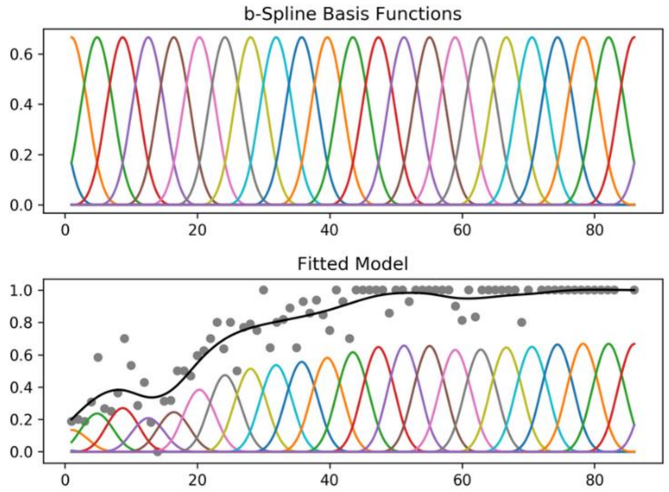
where β_0 is the intercept, $X_t = [x_{t,1}, \ldots, x_{t,d}]$ and (ε_t) is an i.i.d. random noise.

$$f_j(x) = \sum_{k=1}^{m_j} \beta_{j,k} B_{j,k}(x)$$

with coefficient $\mathbf{B_j} \in \mathbb{R}^{m_j}$ where m_j is the chosen spline basis dimension



(a) Load prediction for 2019 using GAM.



(b) A GAM model and its spline basis.

Annexes (2/3)

Boosting Models

- Class of machine-learning models that **combine sequentially weak learners** (e.g. decision trees) building a complex regression model,

- Each new simple model added to the ensemble **compensates for the weaknesses** of the current ensemble,

- **CatBoost** is usually chosen for its **fast optimization** and ability to handle **categorical variables** data (especially calendar data),

Annexes (3/3)

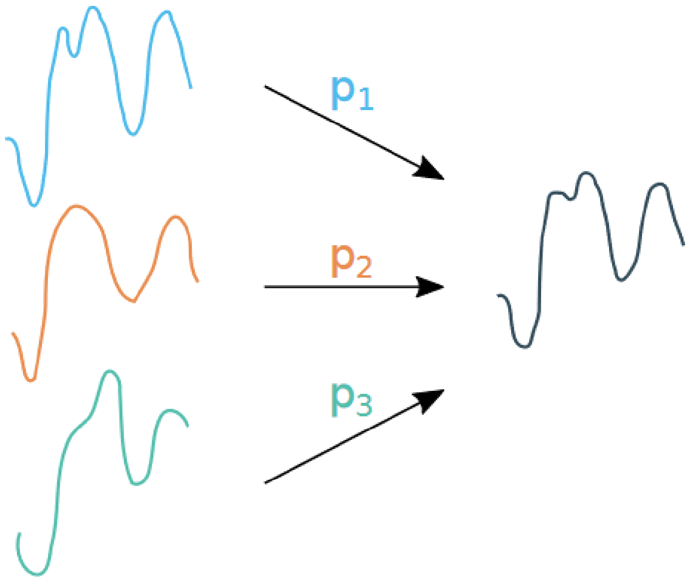
Aggregation of Experts

- Exponentially Weighted Average (EWA)

$$\widehat{p}_{k,t} = \frac{e^{-\eta \sum_{s=1}^{t-1} \ell_s(x_{k,s})}}{\sum_{i=1}^{K} e^{-\eta \sum_{s=1}^{t-1} \ell_s(x_{i,s})}}$$

- Polynomial weighted averages with multiple learning rates (ML-Poly)

  - set $\eta > 0$
  - set initial weights to $p_{j,1} = 1/N$
  - initialize $\widehat{y}_1 = \sum_{j=1}^{N} p_{j,1} f_{j,1}$
  - for $t = 2, \ldots, T$
    - for each expert j, pick the learning rates: $\eta_{j,t-1} = 1/\left(1 + \sum_{s=1}^{t-1} (l(\widehat{y}_s, y_s) - l(f_{j,s}, y_s))^2\right)$
    - update the weights: $p_{j,t} = \eta_{j,t-1} \frac{R_t(\delta_j)^+}{R_t(MLpol)^+}$
    - then aggregation: $\widehat{y}_t = \sum_{j=1}^{N} p_{j,t} f_{j,t}$



$$\hat{y}_t = \sum_k p_{k,t} x_{k,t}$$