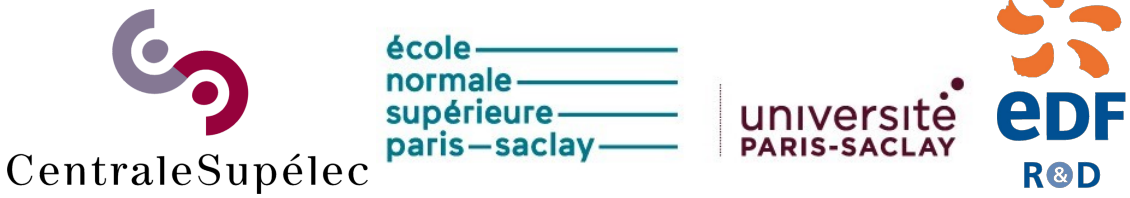
End-of-studies internship report

Rapport de stage de fin d'études

Interpretability of load forecasting models

Interprétabilité des modèles de prédiction de consommation électrique

Gaspard Berthelier
Avril - Octobre 2024



Responsables :
Margaux Brégère (tutrice EDF)
Gilles Faÿ (responsable mention CS)
Pierre-Henri Multon (responsable filière CS)
Stéphanie Allassonnière (responsable MVA)

1 Executive Summary

This report describes my end-of-studies internship at the EDF Lab, the main research and development facility of Électricité de France (EDF). It lasted from the 29th of april to the 18th of october 2024, and consisted of a semester-long research internship, during which I studied the "Interpretability of load forecasting models". The subject stands at the intersection of various fields: explainable artificial intelligence, time series forecasting and electricity production. It concluded my engineering studies at CentraleSupélec as well as my double degree at École Normale Supérieure Paris-Saclay (MVA master).

I was attached to EDF's R&D in the R39 team, within the OSIRIS departement (Optimization, SImulation, RIsks and Statistics). The team develops models to forecast electricy demand in the short and mid-term range, a crucial function in the overall production pipeline, since electricy cannot easily be stored. The study is part of the teams's desire to deepen its knowledge and tools in interpretability. This skill is indeed fundamental to gain insight regarding the various models' predictions.

During the internship, I studied various methods to render complex statistical and machine learning models more interpretable. In particular, I experimented on the Shapley values, which is a popular framework inspired by game theory. The method decomposes a prediction into a sum of distinct contributions for each variable. The end-goal was to study the relevance of Shapley values to estimate the climate and non-climate parts of the global demand (the "load"), as well as to produce a ready-to-use tool for the team to interpret their models.

To that end, we conducted a bibliographic review of interpretability methods for machine learning. After choosing to focus on Shapley values, we looked into the various algorithms to efficiently estimate these values and coded our own library for the computations which we used to conduct experiments on the various parameters. A comprehensive study was made possible by our synthetic dataset which we obtained using the SMACH plateform (Simulation Multi-Agent de l'Activité humaine et des Consommations dans l'Habitat), a multimodal agent simulator developed at the SEQUOIA departement (Services, Economie, Questions hUmaines, Outils Innovants et IA). It allowed us to obtain temporal consumption data for each furniture in all households, thus allowing us to derive the climate and non-climate parts of the load. We validated the algorithms on Generalized Additive Models, which are commonly used to forecast energy loads, as well as on a online mix of models to introduce more complexity.

In the end, we expect promising results which we will gather in an upcoming paper as well as a documented python package for future use.

Contents

2 Introduction

2.1 EDF

EDF (Electricité de France) is a French company which is entirely owned by the State and whose mission is to produce and provide electricity for the country as well as in Europe. It is particularly renowned for its nuclear power, which accounts for about 80% of its production and 70% of the French electricity mix. Its revenue was 140 billion euros in 2023, and it employs approximately 180,000 people. This includes its various subsidiaries such as Enedis and Framatome. EDF's R&D department has 2,000 employees and represents about 1% of the group's expenditures, which amounts to a budget of around 700 million euros. Its purpose is to conduct studies and develop tools for the various operational and decision-making units. The R&D is spread across 9 research centers, including 3 in France: Chatou, Les Renardières, and Paris-Saclay. My internship took place on the Saclay plateau, in Palaiseau (view of the EDF Lab on Figure 1).

During my internship, I was attached to the OSIRIS department (Optimization, Simulation, Risks, and Statistics). Its main mission is to develop decision-support software to forecast energy needs and optimize electricity production. The department comprises about 130 employees. Within this department, different teams coexist: some focus on financial risks and trading on the electricity market, others work on optimizing production means or predicting the renewable energy generation. The R39 group, in which I was located, aims to predict the medium and short-term consumption of businesses and individuals. I was under the supervision of Margaux Brégère, member of R39 and associate research professor at the Sorbonne University. I also colaborated with the SEQUOIA department (Services, Économie, Questions Humaines, Outils Innovants et IA), which provided us with simulated load data from the SMACH platform (Multi-Agent Simulation of Human Activity in Households, Albouys et al., 2019).

Figure 1: EDF Lab on the Saclay Plateau.

2.2 Load forecasting

Electricity is difficult to store at large scale, except at prohibitive cost, which is why the balance between generation and consumption at large scale (the "load") must be maintained at all times. It is traditionally managed by anticipating the demand as well as the intermittent production (wind, solar), in order to adjust the flexible production (hydro dams, nuclear power plants, coal and gaz plants) accordingly. Accurate forecasts of electricity load and renewable production are therefore essential to guarantee grid performance and stability.

State-of-the-art load prediction models are complex machine learning or statistical models. They take in data, usually weather and calendar variables, and output a prediction for the overall demand. With online learning methods (e.g using Kalman filtering, see Obst D, 2021), models are regularly updated to make up for past errors and distributional shifts. In practice, the final prediction is a weighted combination of the various forecasts (called Mixture of Experts, see Section 4.2 as well as Gaillard and Goude, 2015). The following figures illustrate an annual load curve (Fig. 2) and the predictions provided by a Generalized Additive Model (see Section 4.1). This model takes a dozen of variables as input: time of the year, day of the week, temperature, wind speed, humidity, etc. Supplementary graphs on the dataset and the model ouputs are provided in Appendix A.

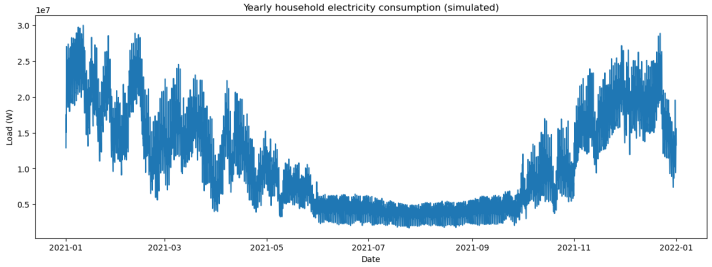


Figure 2: French 2021 households' yearly load (simulated with SMACH). Demand is higher in winter than summer (heating is more widespread than cooling in France).

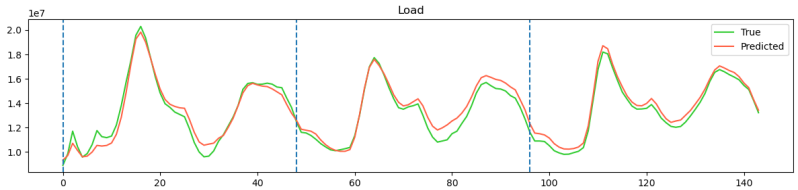


Figure 3: Load predictions for the first three days of 2023, provided by a Generalized Additive Model. It was trained on 4 years of data (2018-2022).

2.3 Interpretablity

Downstream entities use the predictions provided by the models to manage production. However, they do not solely rely on one given prediction. Based on their expertise, they may recalibrate certain results or decide to trust different models. This requires insight regarding the provided predictions. In particular, they will analyze the models' behavior with respect to specific features, for instance the temperature. In practice, this is currently only feasible if the model proposes an intrinsic decomposition according to each variable. That is one of the reasons why Generative Additive Models (GAMs) are mostly used, since their additive structure can be used for explainability (more details in Section 4.1 and inWood, 2017). Yet, the more recent and performant models such as deep neural networks (Keisler et al., 2024) actually suffer from a lack of interpretability. Indeed, these models are no longer additive are present a huge number of parameters. Moreover, the online recalibration of traditionally explainable models such as GAMs renders the interpretability obsolete, since we can no longer easily identify the influence of each variable independently.

For this reason, developing agnostic methods to render any class of model more explainable are much appreciated. There are actually many different methods which have been developed in the domain of explainable AI across the years. Many are catalogued in the very popular book by Molnar, 2022. The main methods are:

- Using readily interpretable models: linear regression, decision trees, Generalized Additive Models (GAMs, Wood, 2017).
- Global methods which describe the mean effect of each variable: Permutation Feature Importance, Partial Dependence Plots (PDP, see Fig. 4), Accumulated Local Effects (ALE, see Fig. 5). More illutrations in Appendix B.
- Locals methods which describe the effect of a variable at a given point: linear surrogate (LIME Ribeiro et al., 2016), counterfactual explanations, and the most popular framework for machine learning: Shapley values (Shapley, 1953, see Section 3).

There are also methods specific to certain architectures, such as TreeShap (Laberge and Pequignot, 2022 for decision trees or GradCAM for Convolutional Neural Networks (Selvaraju et al., 2019). In our case, we wanted to focus on a model-agnostic method.

Figure 4 illustrates the PDP for the temperature. Light blue lines are Invidivual Conditional Expectations (ICE), which correspond to the outputs of the model for a given data point, when varying the variable of the x-axis. The PDP corresponds to the darker line, which is the mean of the ICE lines, i.e the expectation of the model output when only the current variable is fixed. We may also plot expectations conditionally to the current value. This called a Marginal plot (M plot).

Figure 5 illustrates an ALE graph for temperature and position. The method consists in integrating the partial derivatives of the output along the corresponding variable. As expected, electricity demand is sensitive to extreme temperatures, especially in the winter. There is also a second-order version of ALE, as illusrated in Appendix B.

In our setting, we would prefer a local method, to explain individual predictions by the model. LIME is a first candidate, and consists in a weighted linear regression, with a weight that decreases with the distance to the point of interest. Unfortunately, the resulting model is very dependant on the choice of the metric. Counterfactual explanations on the other hand are not well suited for time series predictions.

The most promising method turns out to be Shapley values. It is in fact one of the most popular framework for explainability and is backed with longstanding theory. The method is based on game theory and conditional expectations, and decomposes the prediction into a sum of individual contributions for each feature. We will see in the next section a brief state of the art of Shapley values for machine learning and then we will formalize them more rigorously and see how they are approximated in practice.

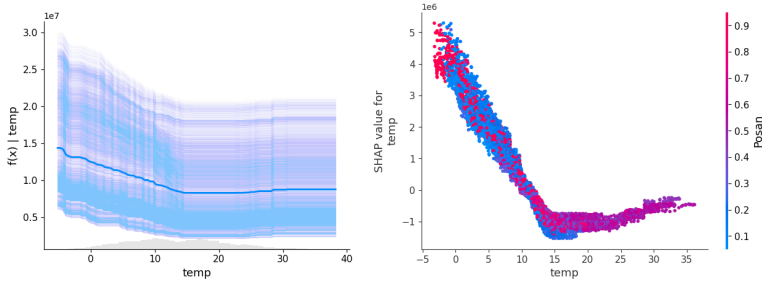


Figure 4: Left: Partial Dependence Plot (PDP) for the temperature. Right: dependence plot (individual points without averaging) for temperature and position in the year.

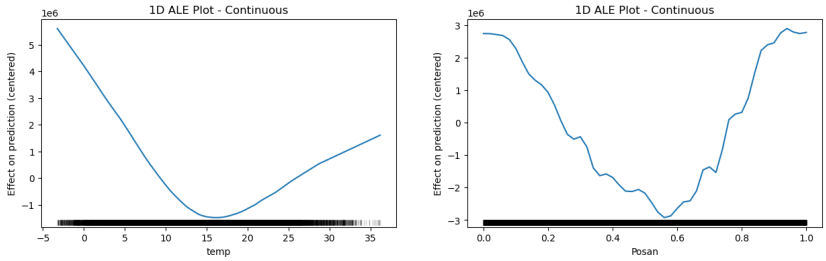


Figure 5: Accumulated Local Effects (ALE) for temperature and position in the year.

2.4 State of the art for Shapley values

Shapley values were introduced in 1953 by Lloyd Shapley as a method to fairly distribute the payoff of a game among a set of players (Shapley, 1953). They consist in computing the average marginal impact of a player to the game, when considering all the possible subsets of participants (called coalitions). Later, they were adapted as a feature selection method, by considering features as players and the payoff as the model's performance, as seen in Cohen et al., 2007. Transposing the original Shapley value definition to machine learning is not straightforward (see Section 3). It depends on how you define the presence of a subset of players and may lead to a wide range of applications (see Olsen et al., 2024). Also, due to the computational complexity of calculating exact Shapley values, many studies proposed approximation techniques using Monte Carlo sampling and permutations (e.g., Castro et al., 2009; Strumbelj and Kononenko, 2010). It was not until relatively recently that Shapley values gained prominence as a local explanation method for machine learning models, particularly through the works of Scott Lundberg (Lundberg and Lee, 2017a). To explain local predictions, different assumptions and approximations can be made (see Rozemberczki et al., 2022). The most common assumption is to consider independent features; this is the case for the two major methods in the Python *shap* libray (Lundberg and Lee, 2017b): Permutation SHAP and Kernel SHAP (see Section 3.3). Shapley values computed in this manner are called interventional values. Efficient approximations like Tree SHAP exist for specific architectures such as tree-based models (Laberge and Pequignot, 2022). In practice, *shap* automatically selects the most appropriate method based on the model architecture. An example of SHAP output is provided in Figure 6. A more general setting involves calculating conditional Shapley values, which are much more difficult to estimate. Some approaches simplify this problem by assuming a Gaussian copula between variables (Aas et al., 2020), while others introduce causal reasoning to enhance the explanation (Heskes et al., 2020). In this paper, we focus on approximation methods for both interventional and conditional Shapley values, primarily using Monte Carlo sampling. Our goal is to obtain a fast and local explanation for a series of predictions. To our knowledge, we are the first to compare our computed explanations to true variable contributions, thanks to our simulated SMACH dataset (see Section 5.1).

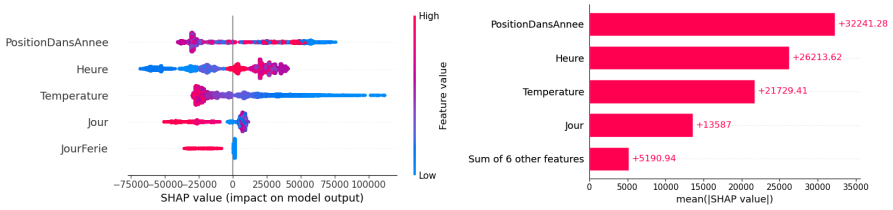


Figure 6: Left : Local Shapley values using the Python *shap* library. Variables may have a positive or negative contribution. Right : Global Shapley values (mean absolute values).

3 Computing Shapley Values

3.1 Definition

We consider a black-box model f which produces predictions $f(x) \in \mathbb{R}$ when inputed with a feature vector $x = (x_1, \dots, x_p) \in \mathbb{R}^p$. In our case, the prediction is the load at a given instant (half-hour time step) and the features are weather and calendar values at this instant. The model may belong to any arbitrary class such as GAMs or neural networks. We suppose the model was fitted on a training dataset $(X_{\text{train}}, Y_{\text{train}}) \in \mathbb{R}^{N_{\text{train}} \times p} \times \mathbb{R}^{N_{\text{train}}}$, where X_{train} is the feature matrix and Y_{train} are the true values which are approximated by $\{f(x) \,|\, x \in X_{\text{train}}\}$. We have access to a distinct dataset $X_{\text{test}} \in \mathbb{R}^{N_{\text{test}} \times p}$, and we would like to explain the predictions $\{f(x) \,|\, x \in X_{\text{test}}\}$. Our aim is to obtain a model-agnostic and local explanation:

- Model-agnostic : the method does not depend on the model class
- Local : we have a distinct explanation for each $x \in X_{\text{test}}$, contrary to a global explanation such as feature importance (Molnar, 2022).

In particular, we would like to obtain for each $x \in \mathbb{R}^p$ an additive decomposition according to each feature, meaning a set of contributions $\phi_j(x) \in \mathbb{R}$ such that:

$$f(x) = \phi_0 + \sum_{j=1}^{p} \phi_j(x) \,. \tag{1}$$

For a given time series $t \to f_t(x^t)$ (i.e the sequential load predictions), this would give us a set of p explainable time series $t \to \phi_j(x^t)$, which add up to the original (see Figure 25).

But how should we distribute a given prediction among the p features? In 1953, Shapley explored a similar problem in the field of coalitional games (Shapley, 1953). Suppose we have a game v and a set of players $J = \{1, \dots, p\}$. After the game is played, the players obtain a payoff $v(J)$ which should be distributed fairly. To do so, we must evaluate the contribution ϕ_j of each player $j \in J$. We could use the difference in the outcome when a player is present or absent from the game. However, one's contribution may be conditional to the presence of another. Thus, we should consider all the possible outcomes $v(S)$ for each coalition of players $S \subset J$. Furthermore, we expect the attribution scheme to be fair, *i.e.* that it satisfies the following fairness properties:

- Efficiency : $\phi_0 + \sum_{j \in J} \phi_j = v(J) \quad$ with $\quad \phi_0 = v(\emptyset) \,.$
- Symmetry : $\forall S \subset J \backslash \{i, j\},\ v(S \cup \{j\}) = v(S \cup \{i\}) \ \Rightarrow \phi_j = \phi_i \,.$
- Dummy : $\forall S \subset J \backslash \{j\},\ v(S \cup \{j\}) = v(S) \ \Rightarrow \phi_j = 0 \,.$
- Linearity : a game $v = v^1 + v^2$ has contributions $\phi_j = \phi_j^1 + \phi_j^2 \,.$

Shapley proved that the unique attribution satisfying these properties is:

$$\phi_j = \sum_{S \subset J \backslash \{j\}} \pi_S (v(S \cup \{j\}) - v(S)) \quad \text{with} \quad \pi_S = \frac{1}{p} \binom{p-1}{|S|}^{-1}. \tag{2}$$

The term $\Delta v(S, j) = v(S \cup \{j\}) - v(S)$ represents the maginal contribution of j to the game with respect to coalition S and π_S would be the probability of choosing S when sampling from coalitions of $J \backslash \{j\}$, by first choosing a coalition size $k \in \{0, \dots, p-1\}$ and then k random features in $\{1, \dots, p\} \backslash \{j\}$. In the next sections, we will use the notation $\mathcal{P}_j = \mathcal{P}(J \backslash \{j\}))$ for the set of coalitions of $J \backslash \{j\}$ and $S \sim P_j$ when S is drawn from $\mathcal{P}_j$ via the method described above. Note that this sampling strategy gives equal weight to each coalition size, contrary to uniform sampling. We can also easily verify that the Shapley values satisfy the fairness properties. In particular, we notice that the sum of contributions is telescopic: $\sum_{j \in J} \phi_j = v(M) - v(\emptyset) + \sum \dots$ where the sum on the right is equal to 0.

In our machine learning setting, we can consider the local prediction as a game where the players are the feature values $(x_1, \dots, x_p)$ and the payoff is the prediction $f(x)$. However, how do we define the outcome of the game when a subset of players are absent? This is not straightforward since f necessarily takes p inputs.

Let's note $v_x(S)$ the outcome of the game associated with $f(x)$, when only S is considered present. Various choices for v_x have been proposed in the literature, depending on the interpretation we desire (Rozemberczki et al., 2022, Olsen et al., 2024). In our case, we consider three versions:

- Baseline

$$v_x(S) = f(x_S, z_{\overline{S}}) \quad \text{where } z \text{ is a reference point (e.g } z_j = \mathbb{E}[X_j]) \,.$$

- Interventional

$$v_x(S) = \mathbb{E}[f(x_S, X_{\overline{S}})] = \int f(x_S, z) P_{X_{\overline{S}}}(dz) \,.$$

- Conditional

$$v_x(S) = \mathbb{E}[f(X_S, X_{\overline{S}}) | X_S = x_S] = \int f(x_S, z) P_{X_{\overline{S}} | X_S = x_S}(dz) \,.$$

where X_S represents the random feature vector $(X_j)_{j \in S}$ and $X = (X_j)_{j \in J}$ has a distribution defined by the empirical distribution of X_{train}, noted $P_{X_{\text{train}}}$ (approximation of real data distribution). Note that this implies $\int f(x_S, z) P_{X_{\overline{S}}}(dz) = \frac{1}{N_{\text{train}}} \sum_{z \in X_{\text{train}}} f(x_S, z)$. This approximation if faithful as long as N_{train} is big enough and X_{train} unbiased.

Both the baseline and interventional method have an issue which is the use of potentially unrealistic data points. Indeed, if features from S and $\overline{S}$ are not independent, a fixed x_S value may be associated with an unlikely $x_{\overline{S}}$ value. For instance, the temperature and calendar variables are highly correlated features. Integrating over the whole scope of temperatures at a given point in time may result in combinations such as (month = January, T = 30°C), which is fairly unrealistic (for now at least). The conditional method appears to be more revelant since it samples $X_{\overline{S}}$ values conditionnally to $(X_S = x_S)$. But this requires the knowledge of the conditional distribution $P_{X_{\overline{S}}|X_S=x_S}$ for all coalitions S, which greatly increases the complexity. Note that if X_S and $X_{\overline{S}}$ were actually independent, we would obtain $P_{X_{\overline{S}}|X_S=x_S} = P_{X_{\overline{S}}}$ and conditional values would be equal to interventional values. This is a common assumption in the litterature (Lundberg and Lee, 2017a). Some studies have tried simplifying the conditional values by setting a gaussian copula between the variables (Aas et al., 2020), whereas some have gone as far as to introduce causality (Heskes et al., 2020). In our case, we will focus on approximation methods for the interventional and conditional Shapley values, mainly based on Monte Carlo methods, presented in the next section.

3.2 Monte Carlo Approximation

Computing $v_x(S)$ on all X_{train} and for all coalitions S is a computation bottleneck. Indeed, the interventional setting requires $N_{\text{train}} \times p \times 2 \times 2^{p-1}$ calls the model. For 4 years of data at a half hour time step and only 10 features, this amounts to 35 million predictions. That is why we must find an approximation of ϕ_j that is much faster to compute. Thus, our goal is to obtain an estimator $\hat{\phi}_j^n(x)$ of $\phi_j(x)$, where n controls the number of calls to the model, such that $\hat{\phi}_j^n(x) \approx \phi_j(x)$ when n is high enough (but hopefully smaller than $2^{p-1} \times N_{\text{train}}$). We notice that:

$$\phi_j(x) = \mathbb{E}_{S \sim P_j}[v_x(S \cup \{j\}) - v_x(S)] = \mathbb{E}_{S \sim P_j}[\Delta v_x(S, j)]\,. \tag{3}$$

Using the definition of v_x for interventional Shapley values, we get:

$$\phi_j(x) = \mathbb{E}_{(S,X) \sim P_j \otimes P_{X_{\text{train}}}}[f(x_S, x_j, X_{\overline{S}\setminus\{j\}}) - f(x_S, X_{\overline{S}})]\,. \tag{4}$$

This corresponds to an expectation which can be approximated with a Monte Carlo method. We shall call Interventional Monte Carlo (IMC) the following approximator:

$$\hat{\phi}_j^n(x) = \frac{1}{n}\sum_{k=1}^{n}\left(f(x_{S_k}, x_j, z_+^k) - f(x_{S_k}, z_-^k)\right) \quad \text{where} \quad z_+^k = z_{\overline{S}_k\setminus\{j\}}^k,\, z_-^k = z_{\overline{S}_k}^k\,. \tag{5}$$

In Equation (12), $(S^k, z^k)_{1\leq k\leq n}$ is an i.i.d. sample of size n from $\mathcal{P}_j \otimes P_{X_{\text{train}}}$. Coalitions and background values are sampled independently.

For baseline Shapley values, only coalitions need to be sampled, and we define the Baseline Monte Carlo (BMC) approximator as:

$$\hat{\phi}_j^n(x) = \frac{1}{n}\sum_{k=1}^{n}\left(f(x_{S_k}, x_j, z_{\overline{S}_k}) - f(x_{S_k}, z_{\overline{S}_k \setminus \{j\}})\right) \quad \text{where} \quad z = \mathbb{E}[X]. \tag{6}$$

In the case of conditional Shapley values, v_x is defined with conditional expectations so:

$$\phi_j(x) = \mathbb{E}_S\left[\mathbb{E}_X[f(x_{S\cup\{j\}}, X_{\overline{S}\setminus\{j\}})|X_{S\cup\{j\}} = x_{S\cup\{j\}}] - \mathbb{E}_X[f(x_S, X_{\overline{S}})|X_S = x_S]\right] \tag{7}$$

We decide to set the Conditional Monte Carlo estimator (CMC) analagous to (5) but with z_+^k drawn from $\Omega(x_{S\cup\{j\}})$ and z_-^k from $\Omega(x_S)$, where $\Omega(x)$ is the neighborhood of x for a chosen metric. This is inspired by commonly used approximations for ALE (see Molnar, 2022). For the metric, we divided the feature spaces into regularly spaced intervals, taking into account categorical and cyclic variables (see Appendix A for more details). In all three algorithms, two quantities are approximated: the search on the coalitions $S \in \mathcal{P}_j$ and the expectation for $\Delta v_x(S)$. For the CMC estimator, S is sampled first and then the background is sampled conditionally, otherwise both are sampled jointly. The higher n, the more precise estimator, but the computation times will be higher. The simplified Monte Carlo algorithm is shown below. The difference between BMC, IMC and CMC lies on line 9, where the background is sampled either as a fixed reference, a random data point or conditionally to S.

Algorithm 1 Monte Carlo Shapley

```
1:  function DELTA_V(f, x, S, j, z)
2:      return f(x_S, x_j, z_{S̄\{j}}) − f(x_S, z_{S̄})
3:  function SHAPLEYMC(f, x, n, sampling_method)
4:      shapley_values ← { }
5:      for j ∈ [1, p] do
6:          V ← [ ]
7:          for k ∈ [1, n] do
8:              S ← sample_coalition(P_j)
9:              z ← sample_background(X_train, sampling_method)
10:             V. append(DELTA_V(f, x, S, j, z))
11:         shapley_values[j] ← mean(V)
12:     return shapley_values
```

For more control over the approximations, we may add another parameter to approximate $\mathbb{E}_S$ and $\mathbb{E}_{X_{\text{train}}}$ at different speeds, using to the following formula:

$$\hat{\phi}_j^{\,n_1,n_2}(x) = \frac{1}{n_1}\sum_{k=1}^{n_1}\hat{\Delta}_x^{n_2}(S_k, j), \tag{8}$$

$$\hat{\Delta}_x^{n_2}(S_k, j) = \frac{1}{n_2}\sum_{k'=1}^{n_2}\left(f(x_{S_k}, x_j, z^+_{k,k'}) - f(x_{S_k}, z^-_{k,k'})\right).$$

Here, we sample n_1 coalitions, and n_2 background samples for each coalition, leading to $n_1 \times n_2$ calls to the model (instead of $n \times 1$). Yet, whatever the choice for $n_1 \times n_2$, an issue arises with the Monte Carlo algorithms: we no longer satisfy efficiency. Indeed, each $\hat{\phi}_j(x)$ are computed separately, so the sampled S are not the same for each j and neither are their associated $v_x(S)$. Yet efficiency requires the sum of $\phi_j(x)$ to be telescopic.

To mitigate this issue, we may think about reusing previously computed values for $v_x(S)$. For instance, we may constrain $v_x(\emptyset) = \mathbb{E}[f(X)]$ and $v_x(J) = f(x)$ beforehand, since both of these coalition values are easily precomputed and are likely to be sampled. We may decide to force their sampling or simply replace their values when sampled. To go even further, we may keep in memory a list of all the previously sampled coalitions and their values, to reuse their values for different features. When using all coalitions, this ensure efficiency. However, this implies that for a given ϕ_j, $v_x(S \cup \{j\})$ might be approximated with different background values than $v_x(S)$, which may impact convergence.

Nevertheless, when sampling random coalitions, efficiency is still not guaranteed. We must either compute all coalitions, or compute a specific subset of coalitions that cancel out. In the next section, we will describe such algorithms

3.3 *shap* library

A very popular python library to compute Shapley values is *shap* (Lundberg and Lee, 2017b), which is based on the SHAP (Shapley Additive Explanations) framework proposed by Lundberg (Lundberg and Lee, 2017a). It considers various reformulations of the Shapley values in order to compute efficient contributions. The two main approaches, detailed below, are called Permutation SHAP values and Kernel SHAP.

3.3.1 Permutation Shapley

Permutation SHAP is the result of long-standing research on Shapley values, which were first approximated using permutations in Castro et al. (2009), and then continuously improved for better convergence for instance in Strumbelj and Kononenko (2010). The idea is to reformulate Equation (2) with permutations instead of subsets.

At each step, the coalition S to consider is the set of features before j in the current order/permutations of features. Sampling a coalition S is then analogous to reordering the features. Indeed:

$$\pi_S = \frac{1}{p}\binom{p-1}{|S|}^{-1} = \frac{|S|!(p-1-|S|)!}{p!} = \frac{\text{number of orders for coalition } S}{\text{number of orders of } J}.$$

Thus, $\phi_j(x)$ can be rewritten as the average on all orders:

$$\phi_j(x) = \frac{1}{p!}\sum_{\sigma\in\Pi(J)} \big(v_x(S_j^\sigma \cup \{j\}) - v_x(S_j^\sigma)\big), \tag{9}$$

where $\Pi(J)$ is the set of all permutations of J and $S_j^\sigma = (\sigma(J)_k)_{k\in\{1,\dots,j-1\}}$ is the coalition of features before j in order σ. When a permutation is sampled, its reverse permutation is also considered to reduce the variance of the estimator (see antithetic sampling in Staudacher and Pollmann, 2023). For a given σ, we iterate j along $\sigma(J)$ and compute each possible $\Delta(v_x, j)$ values (one for each ϕ_j). This allows to achieve efficiency whatever the approximation. An example is provided in Appendix B). The output of Permutation SHAP is illustrated on a gradient boosted model is shown in Figure 25.

3.3.2 Kernel Shapley

Kernel SHAP is inspired by LIME (Ribeiro et al., 2016), in which is fitted a local linear surrogate model g_x to approximate $f$in the neighborhood of x. It is proven that a certain weighted linear regression solves exactly Equation (1), i.e. the coefficients of the model g_x are equal to the Shapley values. The regression is performed on a space of binary variables z, ($z_j = 1$ indicates the presence of a given feature) equipped with a mapping h_x such that $g(z) = f(h_x(z))$ when $h_x(z) = x$ (i.e. when $z = 1_p$). To train the model, a set of perturbed examples are used; the more training samples, the better the approximation, but whatever the degree of approximation, efficiency is always ensured. A more detailed formulation is provided in Appendix C.

4 Analyzing Shapley Values

The previous section presented various methods to estimate Shapley values. This section describes how we may compare them with exact Shapley values, as well as true contributions. To do so, we must derive exact formulations for a certain class of models - called Generalized Additive Models - and carry out some further processing on the raw values.

4.1 Generalized Additive Models

It is sometimes possible to compute exact Shapley values if the model is simple enough. This is the case for Generalized Additive Models (GAMs). A major reference on this subject is the book by Wood (2017). We make the hypothesis that the target variable Y satisfies $Y = f(X) + \epsilon$, with ϵ a random noise, such that:

$$f(X) = \sum_{k}^{K} f_k(X_{I_k}), \quad \mathbb{E}(Y|X) = f(X), \tag{10}$$

where I_k are subsets of indices for the k^{th} function f_k, which belongs to a certain class of functions, typically splines. $K \in \mathbb{N}$ corresponds to the number of different splines we use in the model. GAMs are fitted very efficiently by imposing certain constraints: the splines are decomposed according to a predefined spline basis. A penalized linear regression then allows to find the best set of smooth functions f_k. More details are provided in Appendix D. In the next sections, we will see typical examples of GAM models used for energy load forecasting. This will allow us to derive exact Shapley values.

4.1.1 Monovariate GAM

The monovariate GAM is a model with only p monovariate functions and $I_j = \{j\}$:

$$f(X) = \sum_{j=1}^{p} f_j(X_j)\,. \tag{11}$$

This model is already explainable in itself, but its simplicity will allow us to study Shapley values in depth, to apply them to increasingly more complex models. Note that splines are not necessarily positive and may compensate each other, which can impact interpretability.

We can now compute the theoretical Shapley values.

- Interventional values $\big(v_x(S) = \mathbb{E}[f(x_S, X_{\overline{S}})]\big)$:

  By linearity of the expectation:

$$v_x(S) = \mathbb{E}[f(x_S, X_{\overline{S}})] = \sum_{k \in S} f_k(x_s) + \sum_{k \in \overline{S}} \mathbb{E}[f_k(X_k)]\,.$$

Thus for Interventional Shapley values, it follows that:

$$\Delta v_x(S) = \mathbb{E}[f(x_{S\cup\{j\}}, X_{\overline{S}\setminus\{j\}})] - \mathbb{E}[f(x_S, X_{\overline{S}})] = f_j(x_j) - \mathbb{E}[f_j(X_j)]\,.$$

Which leads to:

$$\phi_j(x) = \big(f_j(x_j) - \mathbb{E}[f_j(X_j)]\big) \sum_{S\subset J\setminus\{j\}} \pi_S = f_j(x_j) - \mathbb{E}[f_j(X_j)]\,. \tag{12}$$

We also find a simplified form of our IMC estimator:

$$\hat{\phi_j}^{n_1,n_2}(x) = f_j(x_j) - \frac{1}{n_1 n_2}\sum_{k}^{n_1}\sum_{k'}^{n_2} f_j(z_{k',j})\,. \tag{13}$$

- Baseline values $\big(v_x(S) = f(x_S, z_{\overline{S}})$ with $z = \mathbb{E}[X]\big)$:

  Similarly, due to the additive form of the monovariate GAM:

  $$\Delta v_x(S,j) = f(x_{S\cup\{j\}}, s_{\overline{S}\setminus\{j\}}) - f(x_S, z_{\overline{S}}) = f_j(x_j) - f_j(z_j)\,.$$

  This leads to:

  $$\phi_j(x) = \big(f_j(x_j) - f_j(z_j)\big) \sum_{S\subset M\setminus\{j\}} \pi_S = f_j(x_j) - f_j(z_j)\,. \tag{14}$$

Those values are easy to compute for each feature. Unfortunately, for conditional Shapley values, we have conditional expectations which we will not be able to simplify.

- Conditional values $\big(v_x(S) = \mathbb{E}[f(x_S, X_{\overline{S}})|X_S = x_S]\big)$:

  $$\Delta v_x(S,j) = \mathbb{E}[f(x_{S\cup\{j\}}, X_{\overline{S}\setminus\{j\}})|X_{S\cup\{j\}} = x_{S\cup\{j\}}] - \mathbb{E}[f(x_S, X_{\overline{S}})|X_S = x_S]$$

  $$= f_j(x_j) - \mathbb{E}[f_j(X_j)|X_S = x_S] + \sum_{k\in\overline{S}\setminus j} \big(\mathbb{E}[f_k(X_k)|X_{S\cup\{j\}} = x_{S\cup\{j\}}] - \mathbb{E}[f_k(X_k)|X_S = x_S]\big)\,.$$

  We can then write:

  $$\phi_j(x) = \sum_{S\subset M\setminus\{j\}} \pi_S\big(f_j(x_j) - \mathbb{E}[f_j(X_j)|X_S = x_S] + \Sigma(S)\big), \tag{15}$$

  where the sum on $\overline{S}\setminus j$ is denoted as $\Sigma(S)$, which is not equal to 0 due to the difference in conditioning between $(X_S = x_S)$ and $(X_{S\cup\{j\}} = x_{S\cup\{j\}})$. Since $\Sigma(S)$ is difficult to compute, we may simplify the definition by assuming one of the equivalent propositions:

  $$\Delta v_x(S,j) = \mathbb{E}(f(X_{S\cup\{j\}}, X_{\overline{S}\setminus\{j\}}) - f(X_S, X_{\overline{S}})|X_S = x_S) \quad \Leftrightarrow \quad \Sigma(S) = 0\,.$$

Assuming this, we would get:

$$\phi_j(x) = f_j(x_j) - \sum_{S \subset M \setminus \{j\}} \pi_S \, \mathbb{E}(f_j(X_j)|X_S = x_S)\,. \tag{16}$$

We can estimate (18) using the neighborhood metric defined previously:

$$\hat{\phi}_j(x) = f_j(x_j) - \frac{1}{n}\sum_{k}^{n} f_j(z_j^k) \quad (S_k, z_k)_k \sim P_j \otimes P_{\Omega(x_S)}\,. \tag{17}$$

All in all, we've obtained exact formulations of Shapley values for Baseline (Equation 14), Interventional (Equation 12) and Conditional Shapley values with a relaxed (Equation 15 and complex (Equation 16) variations. Unfortunately, those formulations only stand for the monovariate GAM. In the next sections, we present slightly more complex models.

4.1.2 Multivariate GAMs

Naturally, monovariate GAMs are not very expressive and can only capture simple behaviors. To specialize the predictions, different models may be fitted for different instants in a given period T. Usually, the period is chosen to be a day. In our case, the data was generated with a half-hour time step, leading to 48 different GAM models. A more complex model may also include the effect of a variable relative to another (bivariate spline). These more complex models can be formulated as such:

- Monovariate per instant: $f(X) = \sum_{t \in T} 1_t(X_t) \left(\sum_{j \in M \setminus \{t\}} f_j^t(X_j) \right)$.
- Adding a bivariate spline: $f(X) = \sum_{t \in T} 1_t(X_t) \big(\sum_{j \in M \setminus \{t\}} f_j^t(X_j) \quad + f_{uv}^t(X_u, X_v) \big)$.

Details on these models are provided in Appendix E. Note that computing exact Shapley values on these is fairly complex; the majority of our comparison studies were made on the monovariate GAM.

Thus, we have defined three types of models with increased complexity. They may also vary with the chosen set of features. For our experiments (see Section 5), we will train various monovariate GAM models with an increasing number of variables, wihch represents another smaller scale of complexity.

In the next section, we will what happen when we compose the predictions of various models together.

4.2 Mixture of Experts

A common practice in predictive tasks is to average the predictions of a set of expert models. This allows to harness the capabilities of different architectures and specializations at once. Typically, the prediction would be expressed as:

$$f_t(X^t) = \sum_i \omega_t^{(i)} f^{(i)}(X^t), \tag{18}$$

where $f^{(i)}$ is corresponds to expert i and $\omega_t^{(i)}$ is the weight associated to $f^{(i)}$ at time t. Those weights evolve in time based on the expert's previous errors (see Figure 7). The experts may themselves be adapted in time (see Obst D, 2021 and Gaillard and Goude, 2015).

Our goal is to verify that the Shapley values of the mix $\phi_j = \phi_j(f_t, .)$ are equal to the weighted combination of the individual Shapley values $\phi_j^i = \phi_j(f^{(i)}, .)$:

$$\forall t,\, \forall j \in J,\, \phi_j(X^t) = \sum_i \omega_t^{(i)} \phi_j^i(X^t) \tag{19}$$

By linearity, this should true be in theory, but the stochasticity in the estimation of Shapley values may induce errors. Also, the temporal update of $\omega^{(i)}$ depends on past errors, which actually break the direct identifiability.

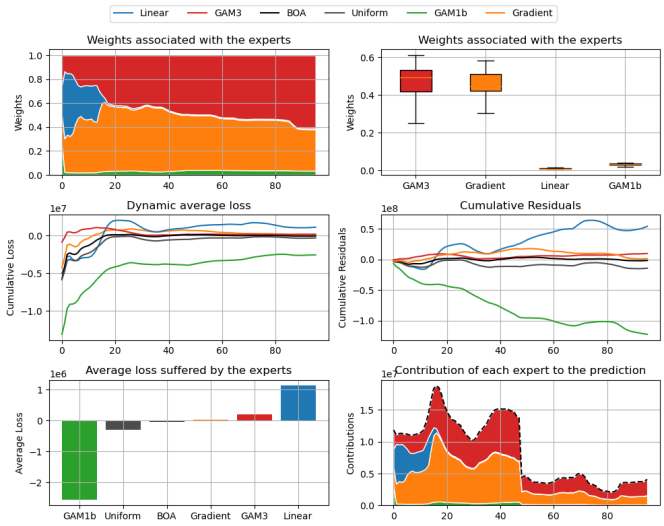


Figure 7: Mix of a linear regression, monovariate GAM, bivariate GAM per instant, and gradient boosted, using the Python opera package (see Gaillard and Goude, 2015).

4.3 Processing of Shapley values

4.3.1 Aggregating

Once we have obtained trustworthy estimations of the contributions of each individual feature, we might like to summarize our results with a reduced number of distinct contributions, for instance a "climate" and "non-climate" part. We partition the set features (e.g $J = C \cup NC$) and introduce the new notation $\tilde{J}$ for the set of reduced coalitions (e.g $\tilde{J} = \{\{C\}, \{NC\}\}$). To compute the Shapley value of $C \in \tilde{J}$, we can simply sum the correspond individual Shapley values:

$$\phi_C(x) = \sum_{j \in C} \phi_j = \sum_{j \in C} \sum_{S \in \mathcal{P}(J \setminus \{j\})} \pi_S \big(v_x(S \cup \{j\}) - v_x(S)\big) \quad \text{with} \quad \pi_S = \frac{1}{p} \binom{p-1}{|S|}^{-1}. \tag{20}$$

This method considers the contribution of a coalition to be the sum of the individual contributions. The advantage of this method is its simple derivation. However, it may not take into account all the joint effects, such as terms of the form $\pi_S(v_x(S \cup C) - v_x(S))$. It also waists a lot of time computing the initial individual contributions when only the aggregates are of interest.

In 1977, Owen introduced the Coalitional value (Owen, 1977), which extends Shapley values in the case of coalitional games, where a prior coalition structure is added to the game. These are very useful for specific data structures such as images, to derive the value of patches of pixels for instance. But this setting is much more general than ours. To obtain the sole contribution of C, we may use the following simplified formula, inspired by Amoukou et al. (2022):

$$\phi_C(x) = \sum_{S \in \mathcal{P}(J \setminus C)} \pi_S^C \big(v_x(S \cup C) - v_x(S)\big) \quad \text{with} \quad \pi_S^C = \frac{1}{p - |C| + 1} \binom{p - |C|}{|S|}^{-1}. \tag{21}$$

In Equation (21), C acts as a single player playing against all the other features. We can simplify the formula even more by considering only two players C and NC, leading to:

$$\phi_C(x) = \frac{1}{2}\big([v_x(C) - v_x(\emptyset)] + [v_x(J) - v_x(NC)]\big). \tag{22}$$

In our experiments, we will derive these three different Shapley values corresponding to the three definitions above. We will designate the Shapley values from Equation (21) as Coalitional Shapley Values and those from Equation (22) as Simplified Shapley Values. Note that all three definitions still satisfy efficiency (see Section 3) and can be further expressed using the different Shapley values' definitions (Baseline, Interventional and Conditional, see Section 3.1) and may be approximated with the different estimators we have studied (see Section 3.2). For Permutation SHAP and Kernel SHAP (see 3.3), only the default Shapley values are available in the *shap* library.

4.3.2 Heightening

Using one of the above-mentioned methods, we have obtained a set of Shapley values which can be expressed in the form of a time series:

$$\begin{array}{rcl} T_{\text{test}} & \to & \mathbb{R} \\ t & \mapsto & \phi_C(x^t) \end{array} \quad \text{where} \quad \left\{ \begin{array}{ll} T_{\text{test}} & \text{is a set of time points} \\ x^t \in \mathbb{R}^p & \text{is a set of features at time } t \end{array} \right. \tag{23}$$

However, the aggregated Shapley values may be negative, which does not make them suitable yet. Indeed, we expect explainable contributions to be positive, since the physical quantity behind is (the load). Also, their sum is only equal to the total load if we add the expectation ϕ_0, which stands as an unexplained offset.

To solve this issue, we may write the offset as:

$$\phi_0 = (\omega_0^C + \omega_0^{NC})\phi_0 \quad \text{with} \quad \omega_0^C, \omega_0^{NC} \in [0, 1]. \tag{24}$$

The fractions ω_0^C and ω_0^{NC} represent respectively the climate and non-climate portions of the expectation. Those may be fixed manually, estimated using a monovariate GAM model, or computed using true contributions.

In our case, true climate and non-climate parts of the load are actually computed using the SMACH dataset (Section 5.1). We consider the consumption related to heating & cooling as climate related ($Load_C$) and the rest as non-climate ($Load_{NC}$):

$$Load_C(x) = heating(x) + cooling(x) \geq 0, \tag{25}$$

with $Load_C(x) + Load_{NC}(x) = Load(x)$. From this, we can compute:

$$\omega_0^C = \frac{\mathbb{E}_{X_{\text{train}}}[Load_C(X)]}{\mathbb{E}_{X_{\text{train}}}[Load(X)]} \in [0, 1]\,. \tag{26}$$

We could also express the output of a monovariate GAM model as a climate and a non-climate part. To do so, we sum the spline outputs of the climate-related variables together $C = \{temperature, sun, wind, \dots\}$, and aggregate the rest as non-climate.

$$f_C(x) = \sum_{j \in C} f_j(x) \in \mathbb{R}\,. \tag{27}$$

Unfortunately, we do not necessarily have $f_C \geq 0$. That is because imposing a constraint of positivity on the weights of a linear regression is not easy, and it is actually an ongoing research topic. Anyhow, unless having positive contributions, we cannot express these terms as percentages directly.

However, there is a small procedure we may apply to mitigate this issue. For a set of time series $t \mapsto f_j(t)$ and an offset f_0, we apply:

$$\tilde{f}_j = f_j - m_j \quad \text{with} \quad m_j = \min(0, \min_t f_j(t)) \leq 0 \tag{28}$$

$$\tilde{f}_0 = f_0 + \sum_j m_j.$$

What we've done is transfer all negative values to the offset, and we are assured that $\forall t,\ \tilde{f}_j(t) \geq 0$. By doing so on the monovariate GAM model, we may compute:

$$\tilde{f}_C(x) = f_C(x) - m_C^{GAM} \geq 0 \quad \text{with} \quad m_C^{GAM} = \min(0, \min_{X_{\text{test}}}(f_C(X)), \tag{29}$$

and $\tilde{f}_C(x) + \tilde{f}_{NC}(x) + f_0 = f(x)$ where $f_0^{GAM} = m_C^{GAM} + m_{NC}^{GAM}$.
Naturally, we may then choose:

$$\omega_0^C = \frac{\mathbb{E}_{X_{\text{train}}}[\tilde{f}_C(X)]}{\mathbb{E}[f(X)] - f_0^{GAM}} \in [0, 1]\,. \tag{30}$$

Whatever the choice for w_0^C, we can now express the total climate contribution as:

$$SHAP_C(x) = \omega_0^C \phi_0 + \phi_C(x)\,. \tag{31}$$

If we assume $\phi_C << \phi_0$, then we can expect with high probability that $SHAP_C(x) \geq 0$. If not, we may heighten this contribution once again, adding a new offset $SHAP_0$ such that:

$$\tilde{SHAP}_C(x) \geq 0, \tag{32}$$

$$SHAP_0 + \tilde{SHAP}_C(x) + \tilde{SHAP}_{NC}(x) = f(x)\,.$$

4.3.3 Normalizing

The objective of the normalization is to be able to compare our estimations together. Our heightening process has indeed introduced some small but unexplainable offsets $\tilde{f}_0$, such that $\tilde{f}_0 + \tilde{f}_C(x) + \tilde{f}_{NC}(x)$. The idea is now to consider the fractions of the form $\omega_C = \frac{\tilde{f}_C(x)}{\tilde{f}_C(x) + \tilde{f}_{NC}(x)}$. This leads to:

$$\omega_{SMACH}^C(x) = \frac{Load_C(x)}{Load(x)} \in [0, 1]\,.$$

$$\omega_{GAM}^C(x) = \frac{\tilde{f}_C(x)}{f(x) - f_0^{GAM}(x)} \in [0, 1]\,.$$

$$\hat{\omega}_{SHAP}^C(x) = \frac{\tilde{SHAP}_C(x)}{\tilde{SHAP}_C(x) + \tilde{SHAP}_{NC}(x)} \in [0, 1]\,.$$

Now, we can easily compare percentage values between each estimated parts and true values. The whole pipeline is summarized in Figure 15.

5 Experiments

5.1 Dataset

The dataset was simulated using the SMACH platform (Simulation Multi-Agent de l'Activité humaine et des Consommations dans l'Habitat), which simulates the behaviors of numerous households in accordance with French behavioral data collected by the INSEE (Institut National de la Statistique et des Etudes Economiques). For more details, refer to the main paper Albouys et al. (2019).

Real climate data was used for the simulation. However, since it was only available at an hourly time step, we had to apply linear interpolation to reduce it to our desired half hour time step. In the simulation, local temperatures are used (67 different centers), but our global models only takes in a single temperature. This temperature is called the "electric temperature", and is a weighted mean of the various local temperatures. The weights correspond to the importance of the node with respect to electrecity demand, which is highly correlated with population density.

After the simulation, we obtained 5 years of data corresponding to load curves from 2018 to 2023 (see Figures 20 and 17). Note that only households' consumption is simulated in SMACH, contrary to public datasets such as RTE's which also measure businesses. The first four years were used as training data for the models (except for covid lockdown periods) and the last year as testing data. Each data point consists of a set of consumptions for each furniture in a household, which we aggregated as a climate (heating and cooling) and non-climate (the rest) load. These correspond to the true climate and non-climate parts which we are trying to estimate using our model and interpretability methods. Their sum correspond to the true load.

In addition to these load values at each point in time, we added the corresponding climate variables (temperature, wind speed, sun exposure, ...) as well as calendar variables (time of the year, day of the week, hour of the day, holidays, ...). This gave us a total of $\sim$ 70k data points and 14 features to train a myriad of supervised models.

5.2 Models

We trained various models to fit the previous data to predict the load. The class of models we focused on were: Gradient Boosted Trees (He et al., 2019), GAMs (Wood, 2017), and Neural Networks (see Keisler et al., 2024). The GAMs were already implemented in R so I adapted my pipeline to be able to execute R scripts. In practice: when a path to an R script is provided, the input data is stored in a local .csv file, the script runs the .rds model, the predictions are saved in another .csv file and processing afterwards in python using the *pandas* library. For the neural networks, hyperparameters were fine-tuned using AutoML (see Keisler et al., 2024), which uses genetic selection techniques to find the best neural structure. These models are trained to output daily consumption data, and use daily data as input. This means using batches of 48 instants at each model call.

Among all these models, we conducted a majority of experiments on a very simple GAM, fitted on the formula:

$$Load = c(Instant) + c(DayType) + s(temp) + s(Posan), \tag{33}$$

where s implies a continuous spline and c a categorical variable (encoded in one-hot encoding). The advantage of this model was the possibility to compare estimated Shapley values to exact Shapley values. We will however conduct more in-depth studies on the more complex models in the upcoming weeks.

The performances of the models on the SMACH data can be found on Table 1. The formulas for each GAM can be found in Appendix F.3. Note that operational GAMs are much more performant (rivalizing with current neural networks) since they include more variables and are adapted online. Ours were simplier on purpose. The metric is the MAPE, computed as: $100 \times \text{mean}_{\text{train/test}}(\frac{|f(x)-y|}{|y|})$.

Model	Train	Test
GAM 1b	14.6	15.8
GAM 1	11.6	12.2
GAM 2	4.0	4.5
GAM 3	3.8	4.5
Gradient Boosted	7.3	7.8
NN + AutoML	0.83	3.3

Table 1: Mean absolute percentage error (MAPE) of trained models.

5.3 Comparative studies

For our comparative studies, we compute aggregated Shapley values on a GAM 1b model (see previous section). We compute MAPE errors between the climate part of the Shapley values and: the true SMACH contribution, the contribution from the GAM model, and the exact Shapley values. We also look at the MAPE between the sum of the Shapley values and the true load (efficiency error), as well as the computation times for each days. The experiments were conducted on 7 or 14 days. This number will be increased in the upcoming experiments.

5.3.1 Impact of n_1 and n_2

In this experiment, we focus on the IMC algorithm to observe the impact of n_1 and n_2.

We can see on Figure 8 that increasing n_1 the number of coalitions naturally decreases the errors but increases the computing time. If we zoom on the SMACH error specifically, we can see that it seems to converge just under 15%, when we reach $n_1 = 1000$.

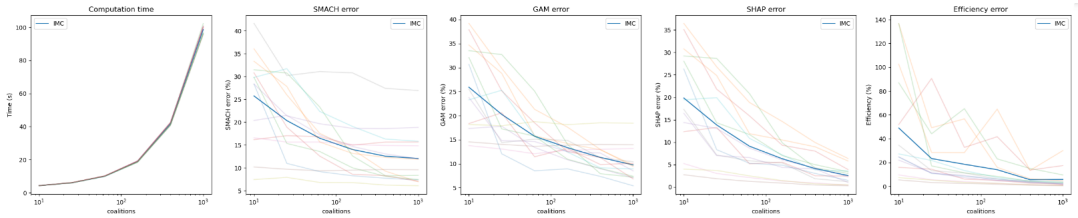


Figure 8: Impact of n_1 on IMC for GAM1b (14 days) with $n_2 = 1$.

We observe a similar behavior when increasing n_2 the number of training data sampled from the background for each coalition. The SMACH error seems to completely plateau after $n_2 = 100$ which could mean we've reached an ireducible bias. On the other hand, the SHAP error keeps decreasing as expected. At $n_1 \times n_2 = 1000$ we are below 5% exact error.

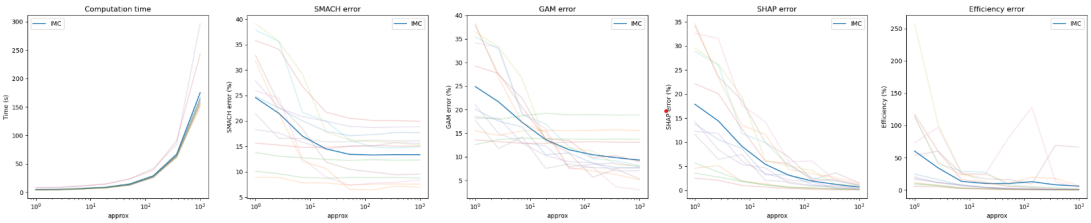


Figure 9: Impact of n_2 on IMC for GAM1b (14 days) with $n_1 = 10$.

On Figure 10, we keep $n_1 n_2$ fixed and observe changing n_1 or n_2 does not have much impact on the errors. This is actually to be expected if we look at Equation 13 which derives the exact Shapley value for a monovariate GAM. This may not be true for more complex models. We notice that increasing n_1 increases computation time more than increasing n_2, which is probably due to the way the loops are coded in the package.

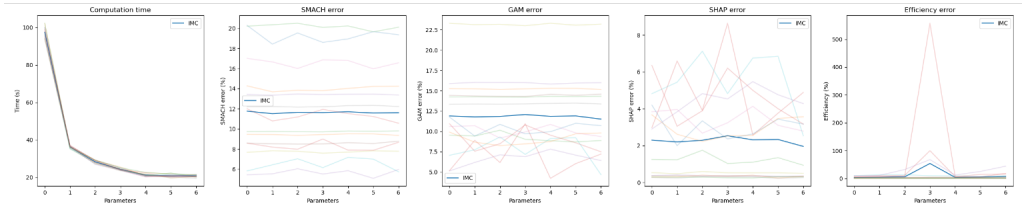


Figure 10: Fixed $n_1 \times n_2 = 1000$ on IMC for GAM1b (14 days).

We will improve this experiment by checking if the n_1, n_2 values at which the errors plateau is the same for all the models, or if it depends on the number of variables they use (we suspect the more variables it uses, the higher the optimal n_1 will be).

5.3.2 Impact of constraining and reusing

Now, we would like to see if our proposed improvements to IMC were successful. In Figure 11, we can see that reusing previously computed values has a huge impact on computation time and on the efficiency error. However, it doesn't seem to effect the other errors much. Constraining on its own (sampled or fixed) did not have much impact either. We also observed that for low values of n_1 and without constraining, reusing values actually destroys the performance, because not enough information has been computed.

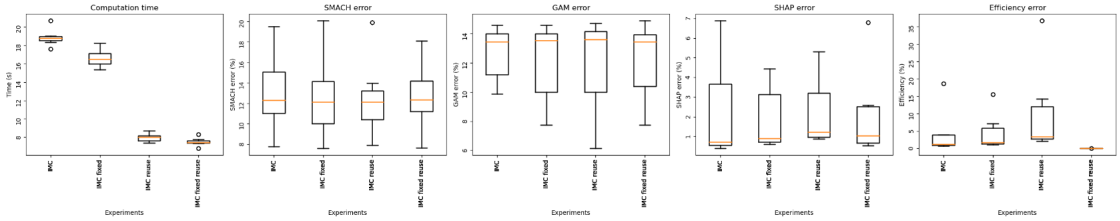


Figure 11: Impact of constrain and reuse with $n_1 = 0$ (no sampling), $n_2 = 1000$ (7 days).

We will improve this experiment by comparing sampled and fixed constraining with different values for n_1. We will also look at the convergence of reusing values with a bigger number of days for more robustness.

5.3.3 Impact of conditioning

For CMC values, there is a new parameter we can play with: the number of intervals to divide the feature ranges in order to select neighbors. Figure 12 shows the impact of the number of intervals. At 1, we are in the case of IMC. We can see that conditioning increases the computing time and decreases the efficiency (we did not find out why). However, it seems to decrease the SMACH error which is a good sign. When computing CMC with all coalitions, $n_2 = 100$ and constraining + reusing, we even get SMACH errors below 5% on average! This was only computed on 7 random days however. We will conduct a more-in-depth study in the days to come.

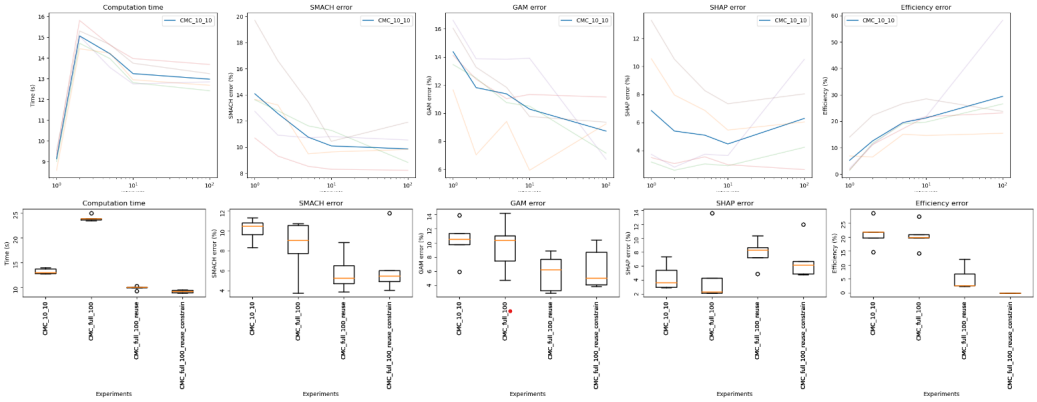


Figure 12: Impact of conditioning (7 days).

5.3.4 Impact of aggregating

On Figure 14, we observe the impact of Coalition Shapley values (agg 1) and Simplified Shapley values (agg 2), on IMC and CMC. The impact on the errors is not huge, agg 2 seems even slightly worse. However, computation times are decreased enormously.

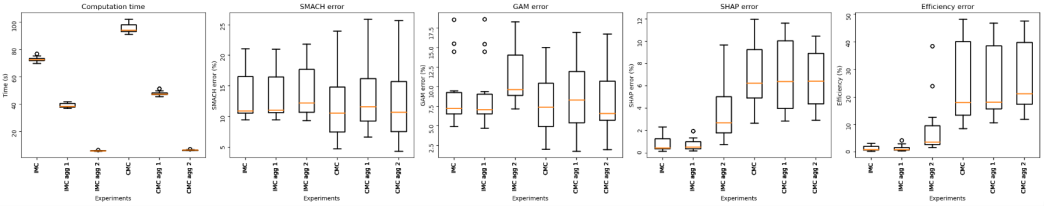


Figure 13: Impact of aggregation with $n_1 = 100, n_2 = 100$ (7 days).

5.3.5 SHAP

We tried varying the parameters used in the *shap* package (number of background samples for Kernel SHAP and number of permutation for Permutation SHAP), but it didn't seem to have much influence. The package is open-source but it is not well documented unfortunately...

Both Kernel and Permutation SHAP have very low efficiency and exact SHAP errors, and a slightly better SMACH error. However, the computation times are insane : 100 times higher than with Monte Carlo for Kernel SHAP, 1000 times for Permutation SHAP!

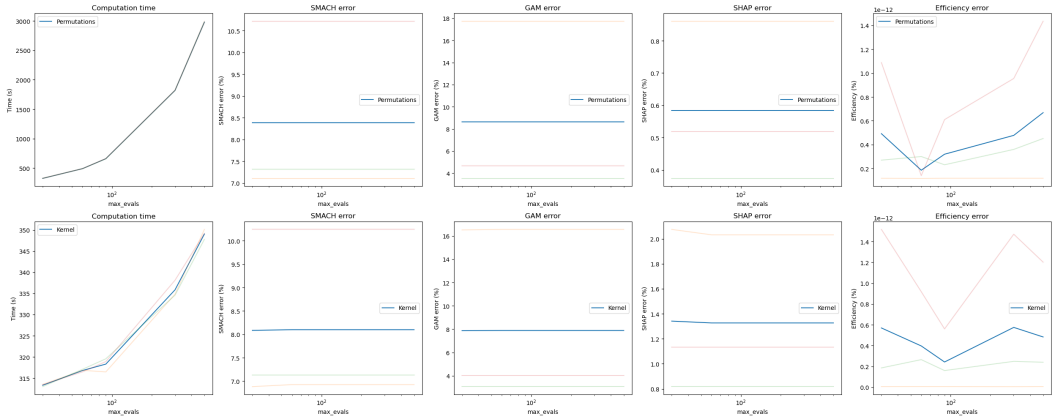


Figure 14: Permutation SHAP and Kernel SHAP (3 days).

5.4 Mixture of experts

Due to software issues (a lot of disconnections during my experiments...), we did not have the time to finish an in-depth study of the mix of experts' Shapley values. I only tested the three sets of parameters on the mix from Figure 7, summarized in Table 2:

IMC	MAPE
$n_1 = 10, n_2 = 1$	446%
$n_1 = 10, n_2 = 100$	20%
$n_1 = 10, n_2 = 100$, reuse, constrain	16 %

Table 2: MAPE between Shapley values of mix and mix of Shapley values (1 day).

5.5 Early conclusions

It is too early to conclude on the true potential of Shapley values to estimate climate and non-climate contributions of the load. There are still 20 days to my internship and much more can be done. Though I will say that from our early experiments shown in the previous pages, my proposed improvements to the default IMC algorithm greatly decreased time complexity without too much loss in performance. Also, conditioning does seem to improve the relation to true contributions.

We must keep in mind that there are biases which we cannot avoid:

- The model only estimates the true load. The better the model, the better the explanations. In our case, we studied very simple models, which enabled to compute exact Shapley values, but which reduced the model's precision.
- The model may mix correlated phenomenons, for instance the load may be caused in part by to the temperature, which is correlated to the position in the year. Thus, the model may learn that position in the year directly impacts the load, which is not exactly true.
- The true contributions themselves from the SMACH dataset are biased. We manually set the climate part to come from the heating and cooling, but we do not know if individuals decide to use heating in november because it is cold, or because they are used to start it at this time of the year.

Nevertheless, we were able to develop a pipeline which returns the usual explainable plots (ALE, PDP, Permutation SHAP) as well as personalized methods with a flexible control of parameters, all of this with interpretable visuals and comparable errors thanks to our processing. This comes in the form of a package (XPC), which is described in the following section.

6 XPC package

In this section, I present the XPC package (eXplainability via Positive Contributions), which I developed during this internship and will be made avalaible for the R39 team as well as publicly on my github.

It was developed in object-oriented Python and inspired by the *shap* library, of which it borrows a few functions (to plot ALEs & PDPs, or to compute Permutation/Kernel Shapley values). Its main contribution is to compute Monte Carlo Shapley values with the whole range of parameters described in the previous sections, and return aggregated explanations according to the pipeline described in Section 4.3 and illustrated in Figure 15. A python notebook is available to showcase the package in action. A typical output can be found on Figure 16.

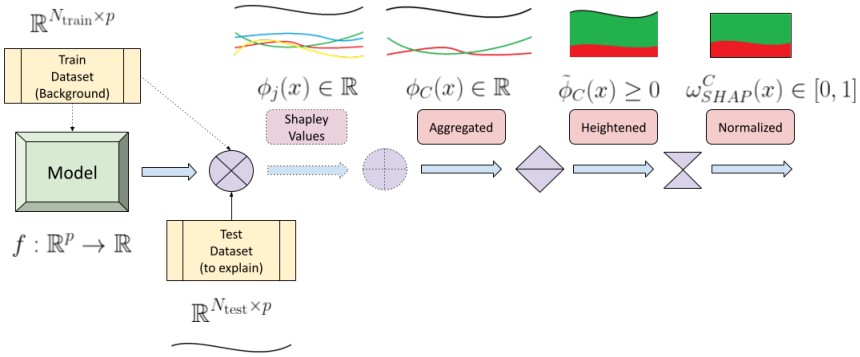


Figure 15: Pipeline to obtain climate and non climate parts of the model predictions.

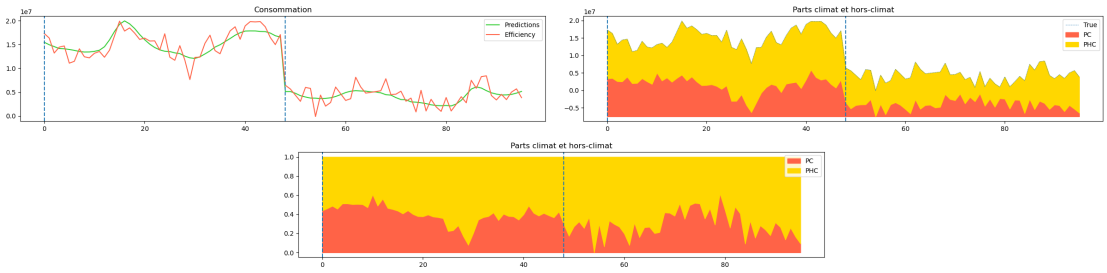


Figure 16: Output of XPC for a day in winter & summer (GAM1B IMC $n_1 = 10$, $n_2 = 1$).

7 Professional takeaways

During this internship, I learned a great deal, both on the theoretical aspects related to the subject, and in general about research as well as working for a big company. EDF's R&D indeed benefits from both worlds, proposing fascinating subjets whilst allowing a confortable work-life balance, ensuring a good pay, material confort and job security.

I really enjoyed working on theoretical aspects of machine learning and focusing on a niche topic which is yet fundamental and has direct applications. More generally, I appreciate the values that EDF showcases: aiming towards a decarbonation of the world's economy, being useful the our country and treating well its employees. For a deeper analysis of the work culture at EDF, I have included my final report for the "Filière Management Opérationnel" in Appendix H.

The main drawback to this internship was the coding environment which I had to use to execute my experiments. For security reasons, getting access to various R and python environments and installing any package was a very tedious task. Also, the software had many issues which lead to quite a few interruptions of my experiments. More generally, long hours of debugging is a side of coding that I do not enjoy at all, but which is a part of any big project, especially in a big company such as EDF. This is not something we are faced with in academic projects and definitely taught me to be more organized than I was before.

After this internship, I will pursue a PhD at EDF's R&D, in the SEQUOIA team. It will be related to Federated Learning for load predictions. I am glad to remain in the sector of electricity production, since I believe it to be of major significance, especially in the years to come. Working in this field whilst deepening my skills in deep learning and statistics is a great opportunity for me. The subject focuses on time series forcasting via the lens of local personalization and privacy. This echoes what I already studied during my gap-year internship at INRIA.

I am very thankful for Margaux Brégère to have guided me during this internship as well as the whole R39 team with whom I had a great time, especially during the numerous sports events.

References

Aas, K., M. Jullum, and A. Løland (2020). Explaining individual predictions when features are dependent: More accurate approximations to shapley values.

Aas, K., M. Jullum, and A. Løland (2021). Explaining individual predictions when features are dependent: More accurate approximations to shapley values. *Artificial Intelligence 298*, 103502.

Albouys, J., N. Sabouret, Y. Haradji, M. Schumann, and C. Inard (2019). Smach: Multi-agent simulation of human activity in households. In *17th International Conference on Practical Applications of Agents and Multi-Agent Systems (PAAMS 2019)*, pp. 227–231.

Amoukou, S. I., N. J.-B. Brunel, and T. Salaün (2022). The shapley value of coalition of variables provides better explanations.

Castro, J., D. Gómez, and J. Tejada (2009). Polynomial calculation of the shapley value based on sampling. *Computers and Operations Research 36*(5), 1726–1730. Selected papers presented at the Tenth International Symposium on Locational Decisions (ISOLDE X).

Cohen, S., G. Dror, and E. Ruppin (2007). Feature selection via coalitional game theory. *Neural Computation 19*(7), 1939–1961.

Gaillard, P. and Y. Goude (2015). Forecasting electricity consumption by aggregating experts; how to design a good set of experts.

He, Z., D. Lin, T. Lau, and M. Wu (2019). Gradient boosting machine: A survey.

Heskes, T., E. Sijben, I. G. Bucur, and T. Claassen (2020). Causal shapley values: Exploiting causal knowledge to explain individual predictions of complex models. *CoRR abs/2011.01625*.

Keisler, J., S. Claudel, G. Cabriel, and M. Brégère (2024). Automated deep learning for load forecasting. *Archive ouverte HAL*.

Laberge, G. and Y. Pequignot (2022). Understanding interventional treeshap : How and why it works.

Lundberg, S. M. and S. Lee (2017a). A unified approach to interpreting model predictions. *CoRR abs/1705.07874*.

Lundberg, S. M. and S.-I. Lee (2017b). A game theoretic approach to explain the output of any machine learning model. `https://github.com/slundberg/shap`.

Molnar, C. (2022). *Interpretable Machine Learning*.

Obst D, de Vilmarest J, G. Y. (2021). Adaptive methods for short-term electricity load forecasting during covid-19 lockdown in france. *IEEE Trans Power Syst 36*(5), 4754–4763.

Olsen, L. H. B., I. K. Glad, M. Jullum, and K. Aas (2024, March). A comparative study of methods for estimating model-agnostic shapley value explanations. *Data Mining and Knowledge Discovery*.

Owen, G. (1977). Values of games with a priori unions.

Ribeiro, M. T., S. Singh, and C. Guestrin (2016). "why should i trust you?": Explaining the predictions of any classifier.

Rozemberczki, B., L. Watson, P. Bayer, H.-T. Yang, O. Kiss, S. Nilsson, and R. Sarkar (2022). The shapley value in machine learning. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, pp. 5572–5579. International Joint Conferences on Artificial Intelligence Organization.

Selvaraju, R. R., M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra (2019, October). Grad-cam: Visual explanations from deep networks via gradient-based localization. *International Journal of Computer Vision 128*(2), 336–359.

Shapley, L. S. (1953). *A Value for n-Person Games*. Princeton, NJ: Princeton University Press.

Staudacher, J. and T. Pollmann (2023, 12). Assessing antithetic sampling for approximating shapley, banzhaf, and owen values. *AppliedMath 3*, 957–988.

Strumbelj, E. and I. Kononenko (2010, mar). An efficient explanation of individual classifications using game theory. *J. Mach. Learn. Res. 11*, 1–18.

Wood, S. N. (2017). *Generalized Additive Models*. Boca Raton, FL: Chapman and Hall/CRC.

A Supplementary figures for load forecasting

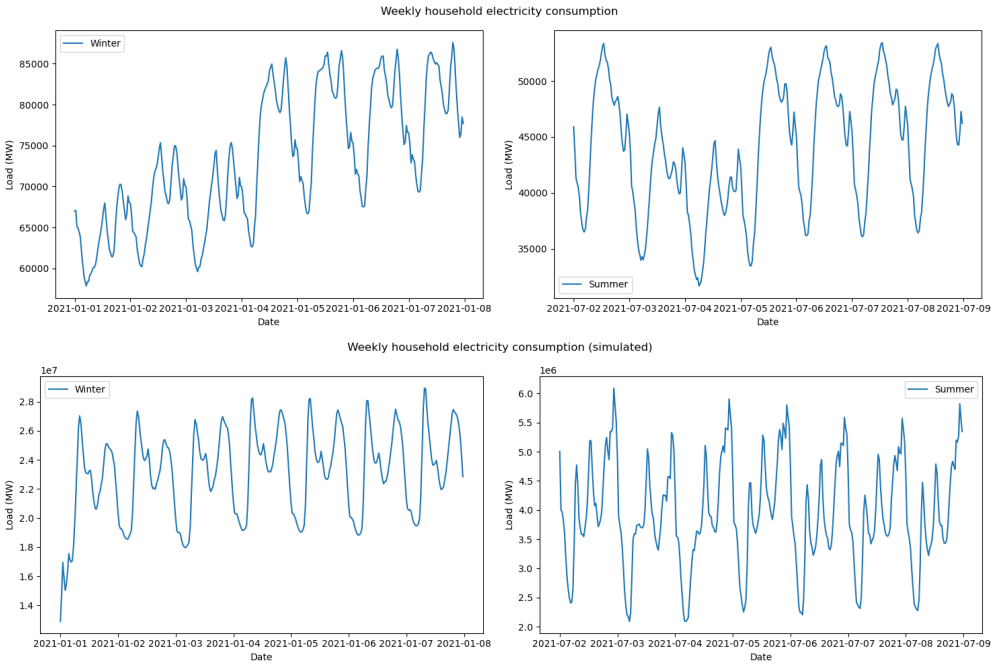


Figure 17: French 2021 weekly load, (winter & summer). SMACH only simulates household consumption whereas the RTE dataset contains both individuals and businesses. Simulated load curves are realistic but not a reproduction of true load curves.

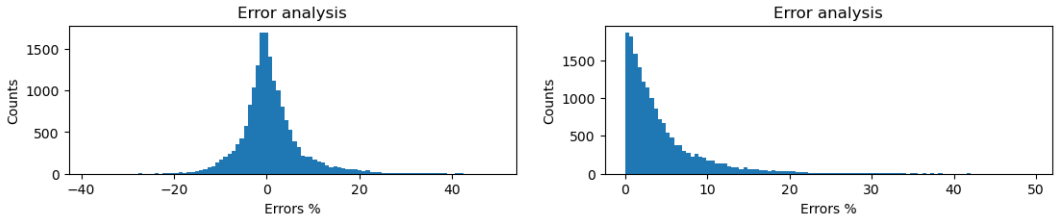


Figure 18: Relative percent errors of the model (absolute values on the right). The absolute mean is 4.48% (the best models of R39 are at roughly 2%).

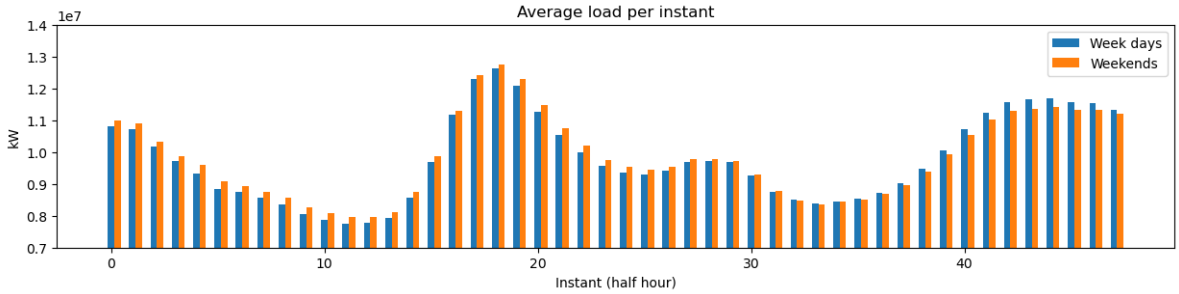


Figure 19: Average load per instant for SMACH dataset.

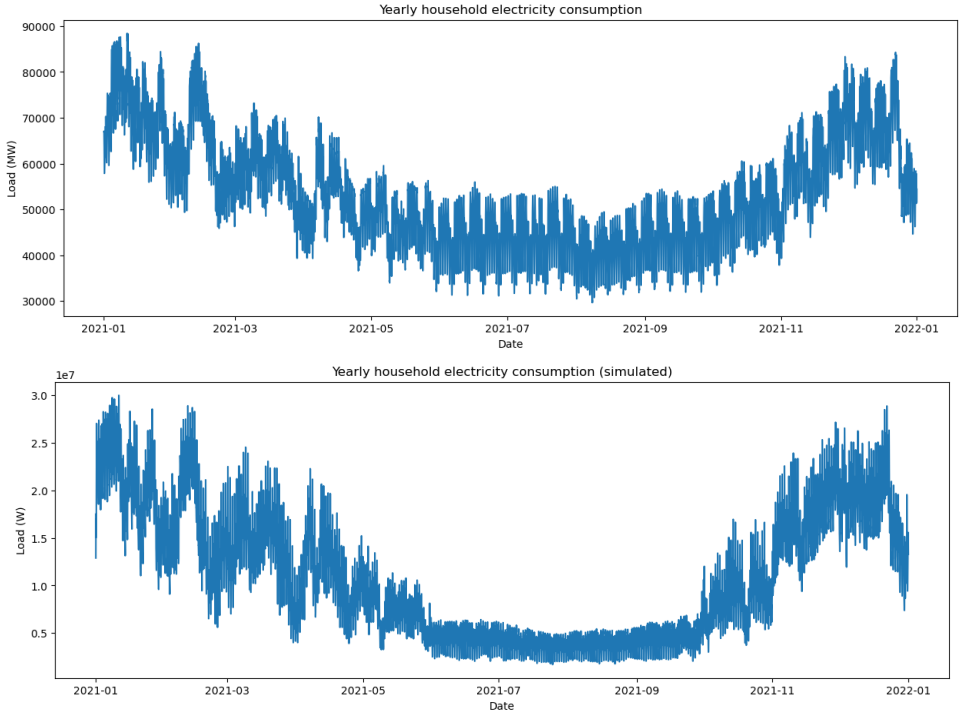


Figure 20: French 2021 yearly load (RTE dataset & simulated with SMACH). We can see that the demand is higher in winter than summer (due to heating). This may vary geographically: more cooling in summer is used in America for instance.

B Supplementary figures for interpretability methods

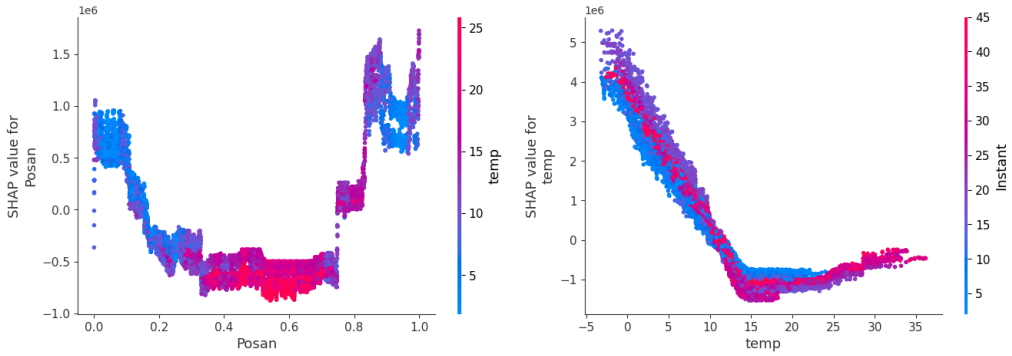


Figure 21: Various dependance plots. We can see that temperature is very correlated to the position in the year whereas instants cover the whole range of temperatures

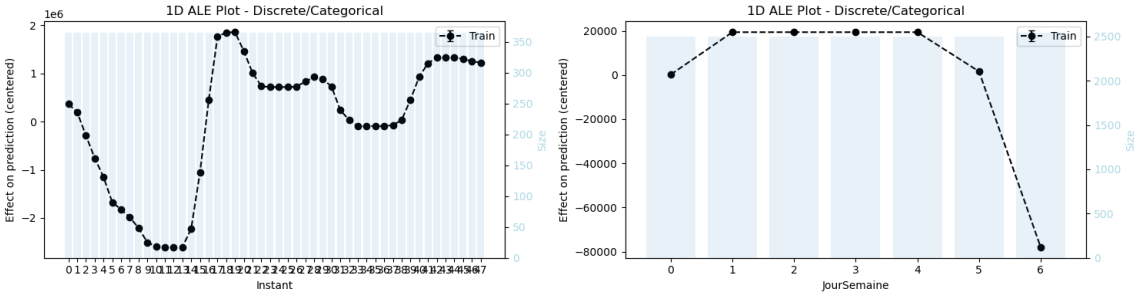


Figure 22: Discrete ALE for times of the day (half-hour time step) and days of the week. Electricity demand is more important at noon and at the end of day, lowest on sunday.

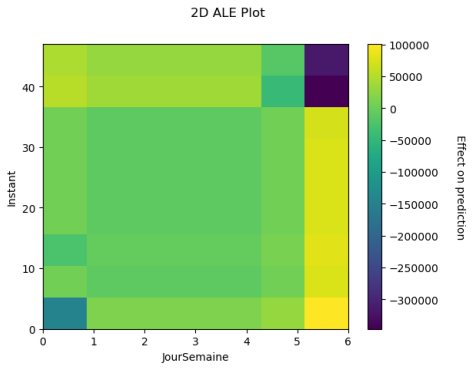


Figure 23: Second order ALE, computed using second order derivatives. Effects seem to the compensate the first oder effects (sunday has a high contribution). Shows that explanation methods can be misleading if not analyzed to their full extent.

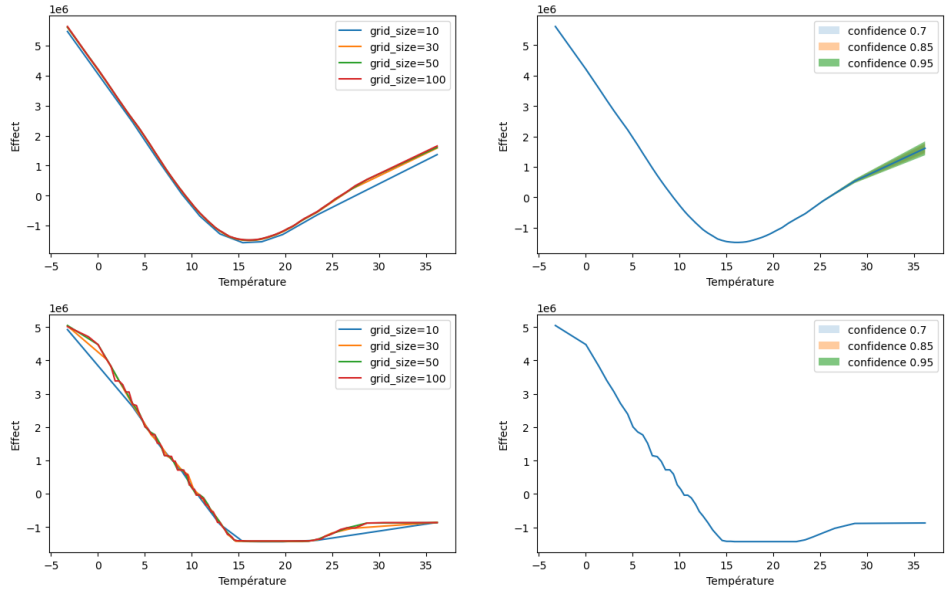


Figure 24: Influence of the grid size on the ALE. On top: GAM model, bottom: gradient boosted. We notice the discrete structure for the boosted model.

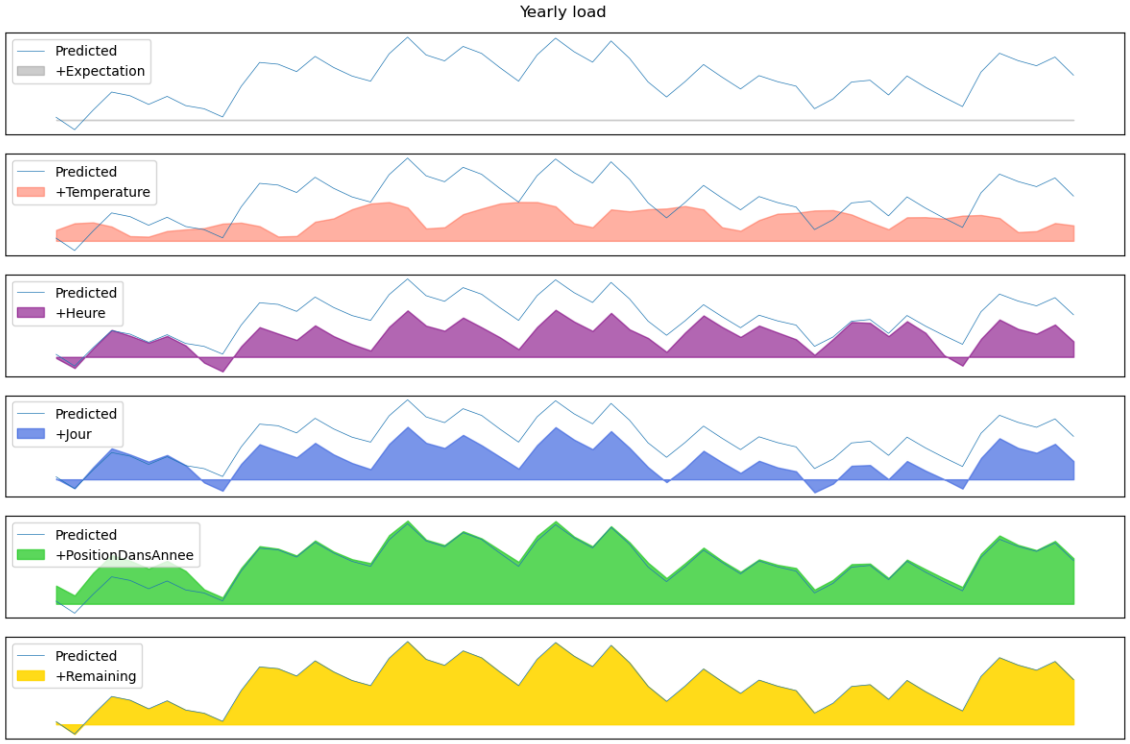


Figure 25: Successive Shapley values for a weekly load. We can see that the temperature's contribution is very high, but decreases with the negative contribution of the hour.

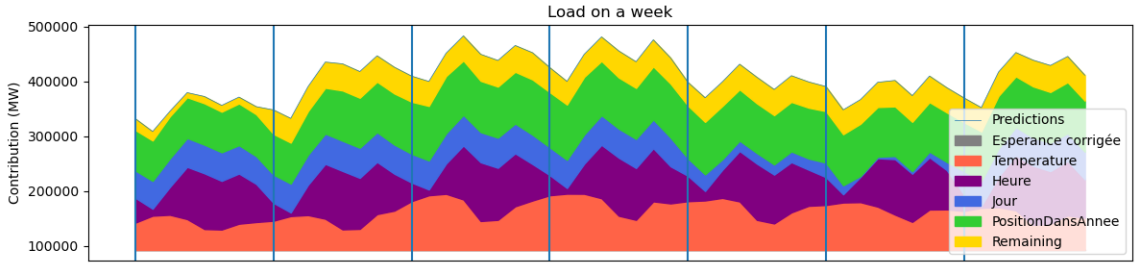


Figure 26: Heightened Shapley values. We substract each feature by its minimum and compensate in the expectation (see section 4.3.2).

C Neighbordhood selection

We describe in this section how we select neighbors to approximate the conditional distributions of the conditional Shapley values.

For continuous features, we divided the features' range into a fixed number of intervals, the same for all features, such that we obtain an interval length L_j for each variable. Then, we consider a data point z to be in the neighborhood of x if: $\forall j \in J, z_j \in [x_j - \frac{L_j}{2}, x + \frac{L_j}{2}]$.

For categorical and cyclic features, the metric is slightly more complex. Indeed, january and december are close in time but their encoding as integers is not. We let the interval around x cycle back to the begining of the range when $x + \frac{L_j}{2} > \max_{X_{\text{train}}} X_j$. For categorical features with no ordering or few unique values, we constrain the neighborhood to be at the exact value of x_j.

Figure 27 illustrates the neighborhood selection around a single data point. We can observe the cyclicity of position of the year and the categorical selection of the instant.

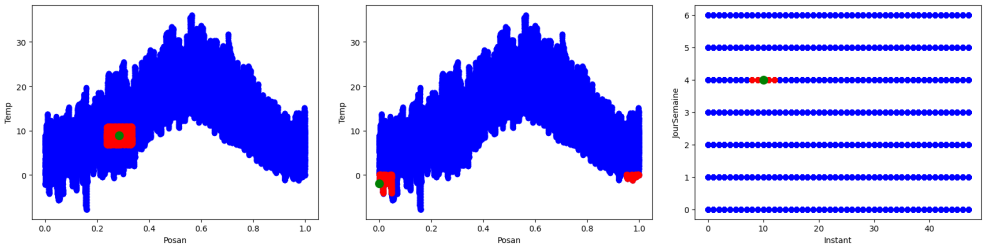


Figure 27: Neighborhood selection; 10 intervals for each non-categorical feature

More specialized metrics would perhaps improve the performances of the conditional methods. For instance, we could vary the number of intervals attributed to each feature, based on feature density or feature importance. We could also use a kernel to smoothly decrease the importance of far away points.

D Efficiency of permutation shap

In this paragraph, we illustrate the practical implementation of Permutation SHAP. As an example, let's consider $p = 4$ features: $J = [1, 2, 3, 4]$ and a given permutation, for instance $\sigma(J) = [2, 1, 3, 4]$. We can compute various coalitions, by iterating over j, using S_j^σ and $S_j^\sigma \cup \{j\}$ as well as $S_j^{-\sigma}$ and $S_j^{-\sigma} \cup \{j\}$ ($-\sigma$ is the antithetic permutation). With those orders, we find:

$$v(\emptyset),\ v(2),\ v(12),\ v(123),\ v(1234) \quad \text{(coalition values approximations)}$$

$$v(\emptyset),\ v(4),\ v(34),\ v(1,3,4),\ v(1,2,3,4) \quad \text{(antithetic permutation)}$$

Now we update the Shapley values according to j:

- $\hat{\phi}_2 \leftarrow \hat{\phi}_2 + [v(2) - v(\emptyset)] + [v(1,2,3,4) - v(1,3,4)]$
- $\hat{\phi}_1 \leftarrow \hat{\phi}_1 + [v(1,2) - v(2)] + [v(1,3,4) - v(3,4)]$
- $\hat{\phi}_3 \leftarrow \hat{\phi}_3 + [v(1,2,3) - v(1,2)] + [v(3,4) - v(4)]$
- $\hat{\phi}_4 \leftarrow \hat{\phi}_4 + [v(1,2,3,4) - v(1,2,3)] + [v(4) - v(\emptyset)]$

We've computed two permutations (counting the antithetic), and we get:

$$v(\emptyset) + \frac{1}{2}[\hat{\phi}_1 + \hat{\phi}_2 + \hat{\phi}_3 + \hat{\phi}_4] = v(1,2,3,4)$$

We would obtain such a result for any permutation and thus for their averages. All in all, we find that we do indeed satisfy efficiency whatever the number of permutations we compute.

E Kernel SHAP

The idea behind Kernel SHAP is to build a local surrogate model which is linear with respect the binary variables $z_j \in \{0, 1\}$:

$$g_x(z) = \phi_0(x) + \sum_j \phi_j(x) z_j, \tag{34}$$

where $z_j = 1$ indicates the presence of the associated feature (player in the game). We want g_x to approximate f in the proximity of x, i.e $g_x(z) \approx f(x)$ when $z \approx 1_p$. The paper introduces a function $h_x : \{0, 1\}^p \to \mathbb{R}$ to map binary features to the original input space, which allows to compute values for the game when features are absent: $v_x(z) = f(h_x(z))$. Note that we use z indistinctly as the binary input variable for g_x and h_x, and as a notation for the coalition $S = \{j \in J \,|\, z_j = 1\}$.

Now, we would like the surrogate model to satisfy properties equivalent to the fairness properties of Shapley:

- Local accuracy : $g_x(1_p) = \phi_0(x) + \sum_j \phi_j(x) = f(x)$ (analogous to efficiency)
- Missingness : x_j absent in original features $x \Rightarrow \phi_j(x) = 0$ (analogous to dummy)
- Consistency : $\forall z \in \{0,1\}^p$, $v_x^1(z + e_j) - v_x^1(z) \geq v_x^2(z + e_j) - v_x^2(z) \Rightarrow \phi_j^1(x) \geq \phi_j^2(x)$

Once again, the only $\phi_j(x)$ that satisfy those properties and Equation 1 are:

$$\phi_j(x) = \frac{1}{p} \sum_{z,\, z_j=0} \binom{p-1}{||z||_0}^{-1} \big(v_x(z + e_j) - v_x(z)\big) \quad \text{with} \quad ||z||_0 = \#\big\{j \,|\, z_j = 1\big\} \tag{35}$$

In the original paper, v_x is chosen such that $f(h_x(z)) = \mathbb{E}[f(x_z, X_{\overline{z}})|X_z = x_z]$, which corresponds to conditional Shapley values, but they make the hypothesis that features are independant and actually use interventional Shapley values. In this case, it turns out the solution g_x is also the solution to the weighted least squares regression problem:

$$\underset{g \in G}{\operatorname{argmin}}\, L_2(f, g, \pi_x), \tag{36}$$

where G is the class of linear models with respect to binary variables z and:

- $L_2(f, g, \pi_x) = \sum_{z \in Z} (v_x(z) - g_x(z))^2 \pi_x(z)$ where Z is the finite set of binary variables
- $\pi_x(z) = \frac{p-1}{\binom{p}{||z||_0}(p - ||z||_0)||z||_0}$ the weight kernel.

In LIME (Ribeiro et al. (2016)), the kernel is arbitrary which yields different possible local surrogate models. In our case, π_x is specifically chosen to be infinite when $||z||_0$ is close to 0 or p (many or few players). This enforces the values of $g(1_p)$ and $g(0_p)$ to be equal to $f(x)$ and $\mathbb{E}_{X_{\text{train}}}[f(X)]$. It is then the unique solution which satisfies the fairness properties.

All in all, this means finding the exact Shapley values can be solved using regression. In practice, a dataset $\hat{Z}$ of sampled binary values is used. The bigger $\hat{Z}$ is, the better the approximation. To improve convergence, we can sample hierarchically, by completing smaller layers, where $\text{layer}_k = \{z,\, ||z||_0 = k\} \cup \{z,\, ||z||_0 = p - k\}$). Note that whatever the degree of approximation, efficiency is always satisfied.

In theory, both the interventional and conditional Shapley values can be computed with this method. The main advantage of Kernel SHAP over sampling methods is its computational speed. That is because its solution can be reformulated as a matrix operation (see Aas et al., 2021). However, the simplicity of this formulation is only achieved if feature independance is assumed, or a dependance copula is defined, which is not the case in practice. For this reason, Kernel SHAP has been progressively abandonned in favor of the traditional Permutation SHAP.

F Generalized Additive Models

To build a GAM, one must start by choosing a family of functions F (e.g splines of degree 3) and a set of indices $(I_k)_{1\leq k\leq K}$, where K is the number of additive functions. Then we fit the optimal set of functions $f_k \in F$ with respect to $(X_{\text{train}}, Y_{\text{train}})$. This is usually achieved by expressing the splines into spline basis $(s_l^k)_{1\leq l\leq L_k}$, and finding the best decompositions $\sum_p \beta_l^k s_l^k$. Our aim is to approximate $Y_{\text{train},i}$ with:

$$\hat{Y}_{\text{train},i} = \beta_0 + \sum_k^K \sum_l^{L_k} \beta_l^k s_l^k(X_{\text{train},i,I_k}), \tag{37}$$

$\hat{Y}_{\text{train}} = \tilde{X}_{\text{train}}\beta \quad$ with $\tilde{X}_{\text{train},i} = (1, s_1^1(X_{\text{train},i,I_1}), \ldots, s_{L_K}^K(X_{\text{train},i,I_K}))$ and $\beta = (\beta_0, \beta_1^1, \ldots, \beta_{L_K}^K)$.

We have expressed the problem into a linear regression, which can be solved using regularized least squares:

$$\beta_{opt} \in \underset{\beta}{\operatorname{argmin}} \, ||Y_{\text{train}} - \tilde{X}_{\text{train}}\beta||^2 + \lambda \beta^T S \beta, \tag{38}$$

where S is a symmetric semi-definite positive matrix that contains information about the second derivatives of s_l^k. This enables to enforce smoothness by constraining f''. More details can be found in Wood, 2017.

For a monovariate GAM (see section 4.1.1), we get:

$$\hat{Y}_{\text{train},i} = \sum_{j\in J} f_j(X_j) = \sum_{j\in J} \sum_l^{L_j} \beta_l^k s_l^k(X_{\text{train},i,j}). \tag{39}$$

Which allows us to derive an simple expression for the expectation of a spline:

$$\mathbb{E}(f_j(X_j)) = \sum_k b_k^j \, \mathbb{E}(s_k^j(X_j)).$$

When all the features belong to a share feature space and thus a shared spline basis, this can simplify many of the computations we will have to do. Unfortunately, this is not the case due to a mix of continuous features with various ranges, categorical features, etc.

The next page decribes in more details the more complex GAM models.

F.1 Monovariate GAM per instant

In this model, a different monovariate GAM is fitted for each instant. This can be expressed with $I_j = \{j, t\}$ and $f_j^t = 1_t(X_t)f_j(X_j)$, leading to:

$$f(X) = \sum_{t \in T} 1_t(X_t) \big(\sum_{j \in M \setminus \{t\}} f_j^t(X_j) \big) . \tag{40}$$

Now, the interventional value of the coalition S depends if t is in S or not:

- If $t \in S$: $\mathbb{E}(f(x_S, X_{\overline{S}})) = \sum_{j \in S \setminus \{t\}} f_j^{x_t}(x_s) + \sum_{\tilde{j} \in \overline{S}} \mathbb{E}(f_{\tilde{j}}^{x_t}(X_{\tilde{j}}))$.
- If $t \notin S$: $\mathbb{E}(f(x_S, X_{\overline{S}})) = \sum_{t \in T} \frac{1}{T} \big(\sum_{j \in S} f_j^{x_t}(x_s) + \sum_{\tilde{j} \in \overline{S} \setminus \{t\}} \mathbb{E}(f_{\tilde{j}}^{x_t}(X_{\tilde{j}}) | X_t = x_t) \big)$.

Some conditional expectations appear but they are easy to compute since in our setting X_t only takes $T = 48$ different values (we have data every half hour).

F.2 Adding a bivariate spline

We go back to the more general setting with K subsets I_k:

$$f(X) = \sum_{k}^{K} f_k(X_{I_k}) . \tag{41}$$

This setting is quite inconvenient for general computations, so we shall only consider monovariate splines and one bivariate spline of indices $I = (u, v)$, per instant. This leads to:

$$f(X) = \sum_{t \in T} 1_t(X_t) \big(\sum_{j \in M \setminus \{t\}} f_j^t(X_j) \quad + f_{uv}^t(X_u, X_v) \big) . \tag{42}$$

Computing the expectation is similar to the previous case, depending on if t is in S or not. The only difference is we'll have another term $\mathbb{E}(f_{uv}^t(X_u, X_v)|X_S = x_S))$, which depends on the nature of f_{uv}^t:

- Linear : $f_{uv}^t(x, y) = \lambda b_1^t(x) + \mu b_2^t(y)$.
- Marginal : $f_{uv}^t(x, y) = \sum_p b_p^t s_{u,p}^t(x) s_{v,p}^t(y)$.
- Tensor/Copula : $f_{uv}^t(x, y) = \sum_{p_1,p_2} b_{p_1 p_2}^t s_{p_1 p_2}^t(x, y) = \sum_{p_1,p_2} b_{p_1 p_2}^t K_{p1,p2}^t(x, y) s_{p_1}^t(x) s_{p_2}^t(y)$.

In practice, the linear case is excluded since b_1^t and b_2^t would actually be included in monovariates f_u^t and f_v^t. Otherwise, the cases where $u, v \in S$ or one of both is pretty straightforward. If $u, v \notin S$, and X_u and X_v are independant, we have:

$$\mathbb{E}(b_p^t s_{u,p}^t(X_u) s_{v,p}^t(X_v)) = b_p^t \mathbb{E}(s_{u,p}^t(X_u)) \mathbb{E}(s_{v,p}^t(X_v)) . \tag{43}$$

If no independance is assumed, we must integrate the bivariate spline over the whole dataset.

F.3 Formulas of our trained models

We describe here the various formulas for the different GAM models we trained. Their performances are displayed on Table 1.

- GAM 1b: Load = c(Instant) + c(DayType) + s(temp) + s(Posan)
- GAM 1: Load = offset + Instant + DayType + s(temp) + s(temp_liss_fort) + s(temp_liss_faible) + s(Posan) + s(tempMax) + s(wind) + s(sun) + s(tempMin)
- GAM 2: Load = offset + DayType + s(temp) + s(temp_liss_fort) + s(temp_liss_faible) + s(Posan) + s(tempMax) + s(wind) + sun + s(tempMin)
- GAM 3: Load = offset:DayType + s(temp) + s(temp_liss_fort) + s(temp_liss_faible) + s(Posan) + s(tempMax,wind) + sun

DayType indicates if the day is a week day, a weekend or holiday.

TempLissFaible and TempLissFort correspond to a smoothing of the temperature (weak and strong).

TempMin and TempMax correspond to the minimum and maximum temperature in a fixed window.

Wind corresponds to wind speed and sun to sun exposure.

Posan is the position in the year (between 0 and 1) and Instant is the half hour index (48 instants per day).

G Premier rapport d'avancement

Envoyé le 10/06/2024

Consignes

À envoyer 1 mois après le début du stage aux Responsables Pédagogiques (RP). Approfondir et synthétiser ma compréhension du contexte et de la mission. Reformuler la problématique, mettre en avant l'appropriation de la mission et les mise en perspective des enjeux, planning et livrables attendus.

Reformulation du contexte et du sujet

La production et la distribution d'électricité est une tâche complexe, lié au fait que cette énergie se stocke difficilement et qu'il est donc impératif de réaliser un équilibre constant entre l'offre et la demande. De ce fait, des modèles de prévisions de la demande sont développés à la R&D d'EDF. Ces modèles doivent être en mesure d'anticiper la "charge" (i.e la demande globale) avec haute fidélité, mais doivent aussi être interprétables.

En effet, la demande réelle évolue constamment et peut parfois subir des variations inattendues qui induisent des erreurs dans les modèles. Des équipes sont ainsi chargées de recalibrer régulièrement les modèles en utilisant leur expertise propre, mais ils requièrent pour cela un certain recul sur ce qui a amené tel modèle à prendre telle décision. Il se trouve qu'historiquement ce sont des modèles Generative Additive Models (GAMs) qui sont majoritairement utilisés. Or ces modèles permettent justement de décomposer la consommation totale en une somme de contributions, dépendantes chacune de variables distinctes. On peut ainsi connaître exactement la part de la météo, du calendrier, etc. Cela permet ainsi de prendre du recul sur les prédictions renvoyées par un modèle, voire de pouvoir anticiper son comportement lors de changements conjoncturels.

Plus globalement, tout le groupe EDF se positionne depuis quelques années dans une stratégie "CAP 2030", qui a pour missions d'aider les ménages à maîtriser leur facture énergétique, tripler l'activité du groupe à l'international, et de doubler la production d'énergies renouvelables. La part croissante des énergies renouvelables dans le mix énergétique induit cependant des aléas dans les stocks d'énergie disponible à chaque instant, et renforce donc le besoin de connaître précisément la demande. En plus de cela, il apparaît dans un contexte de sobriété énergétique le besoin d'avoir une compréhension fine des variables à fort impact sur la demande, afin de pouvoir activer les leviers pertinents pour une économie bas-carbone et d'être en mesure d'avoir une analyse critique des effets réels.

Au-delà de ce contexte lié à EDF sur les besoins en modèles de prévision fiables et explicables, le domaine de l'IA prédictive évolue grandement avec récemment l'entrée en scène du Deep Learning. Étant donné les performances accrues, la R&D d'EDF s'est naturellement intéressée à ces modèles basés sur les réseaux de neurones. Le problème est que ces modèles sont dits "boîte noire" : il est très difficile d'expliquer leurs prédictions. C'est dans ce contexte qu'advient le sujet de mon stage, à savoir "l'interprétabilité des modèles d'apprentissage automatique pour expliquer les prévisions de consommation électrique". L'objectif est en réalité de pouvoir retrouver une décomposition avec une somme d'effets compréhensibles, similaires à celle qu'on a avec les GAMs, mais qui fonctionne sur les prédictions données par les modèles boîtes noires.

Cadre et organisation

Dans le cadre de ce stage, je suis installé au R&D Lab d'EDF à Palaiseau. Ce site est le plus grand parmi les 9 existants du groupe. Je suis plus particulièrement intégré au département OSIRIS (Optimisation, SImulation, RIsques et Statistiques), qui a pour mission de développer des logiciels d'aide à la décision pour les entités métier. Au sein de ce département, différentes équipes coexistent. Certaines se focalisent sur les risques financiers et le trading sur le marché de l'énergie, d'autres sur l'optimisation des moyens de production ; le groupe R39 dans lequel je me trouve a pour mission de prédire la consommation moyen et court terme des entreprises et particuliers. Plus précisément, R39 fournit des modèles de prévision à trois entités distinctes : ENEDIS (filiale EDF chargée de la gestion et de l'aménagement du réseau), la DOAAT (Direction Optimisation Amont Aval Trading d'EDF), et la DCO (Direction Commerciale Opérationnelle d'EDF). Il y a aussi l'Energy Data Lab qui est entièrement dédié à la recherche. Mon stage est sous la direction de Margaux Brégère, membre de R39 ainsi que qu'enseignante-chercheuse associée à la Sorbonne. Le projet s'inscrit dans la volonté de R39 d'approfondir ses connaissances et outils en interprétabilité. Cela s'avère de plus en plus critique à mesure que les modèles à base de réseaux de neurones se déploient. Actuellement, les prédictions finales sont un mélange entre les prédictions des GAMs et des modèles plus complexes dont ceux de deep.

Ma mission peut ainsi se décomposer en plusieurs étapes:

- Réaliser une étude bibliographique des méthodes d'interprétabilité en IA.
- Appliquer les méthodes pertinentes sur les modèles de prédiction de consommation électrique.
- Estimer la fiabilité de ces décompositions en comparant à celles intrinsèques des GAMs par exemple ou bien à des données de contributions réelles.
- Développer une boite outil qui puisse être associée aux modèles de R39.

Évidemment, il est difficile voire impossible d'estimer la contribution réelle de chaque variable dans la demande globale. Pour valider nos méthodes, il a donc été décidé de générer des données synthétiques de consommation, avec la plateforme SMACH. Ce simulateur a été développé par SEQUOIA (Services, Économie, QUestions humaines, Outils innovants et IA, département de la R&D distinct de OSIRIS) et permet de réaliser des simulations multi-agents, qui agissent selon des scénarios aléatoires de consommation, suivant des statistiques calculées sur des échantillons de la population française (données récoltées avec l'INSEE). Nous avons alors accès aux détails de la consommation de chaque agent, et pouvons ainsi remonter aux vraies parts climat / hors climat par exemple. Cela permet alors de confronter ces données aux contributions estimées par nos méthodes.

Je travaille actuellement avec un PC de travail dans un bureau que je partage avec une autre stagiaire. J'échange régulièrement avec les autres membres de l'équipe au sujet du contexte global de la prévision, et j'ai eu une première réunion avec Mathieu Schuman de SEQUOIA pour établir le cahier des charges pour les données SMACH. Nous faisons par ailleurs des points hebdomadaires avec Margaux pour discuter des progrès et des pistes à suivre.

Avancées

Cela fait pratiquement un mois que j'ai commencé le stage. Pour l'instant, j'ai parcouru la bibliographie tant sur le contexte de la problématique que sur les aspects techniques liés à l'interprétabilité des modèles. J'ai réalisé des premières simulations sur des données RTE, puis sur des données internes à EDF plus complètes. Je me suis particulièrement concentré sur les Approximated Local Effects (ALE) et les valeurs de Shapley, et commencé à coder une librairie interne pour spécialiser celle déjà existante en python. L'intérêt est de pouvoir l'utiliser sur les modèles codés en python ou en R de la R&D, et d'avoir un meilleur contrôle sur les paramètres d'approximation. Nous avons par ailleurs lancé récemment la simulation des données SMACH et devrons donc bientôt être en mesure d'analyser la pertinence de nos décompositions.

À terme, Margaux aimerait avoir un outil indépendant du modèle à expliquer, d'où le choix de se concentrer sur les variables de Shapley. Celles-ci sont en effet calculées à base de théorie des jeux et d'hypothèses statistiques sur les données, et permet de décomposer localement (e.g à chaque instant) la contribution de chaque variable. Mais il y a alors toute une problématique autour du choix de la méthode pour approximer les coefficients, car il s'agit d'évaluer des espérances sur des coalitions de variables en très grand nombre, ce qui implique en pratique de devoir échantillonner. Il apparaît aussi divers enjeux liés à la dépendance des variables entre elles, voire d'estimer les phénomènes de causalité, etc.

Pour tout ce qui est planning il n'y a pas d'échéances particulières puisqu'il s'agit d'un sujet de recherche. L'idée est d'explorer les pistes que nous avons et voir jusqu'où nous pouvons aller. Il y a néanmoins l'objectif d'avoir un modèle qui fonctionne "bien" d'ici la fin du stage. L'évaluation de ses performances n'est pas évidente puisque la vrai influence de chaque variable dépend des hypothèses et définitions qu'on se donne, néanmoins un calcul des écarts entre notre décomposition et celle des GAMs ou des contributions avec SMACH est une première idée. J'ai par ailleurs proposé qu'on réalise une étude à la fois théorique et empirique de l'effet des différentes approximations dans le calcul des valeurs de Shapley. Selon l'avancement, cela pourra éventuellement amener à la publication d'un article sur ce sujet. Il y aura de toute manière le rapport final qui décrira plus en détail les expériences menées et les résultats obtenus.

H Rapport Spécifique d'Observation Managérial

Ce rapport a pour vocation d'établir une analyse de l'entreprise d'accueil de mon stage de fin d'étude, à savoir EDF, sur les aspects managériaux ainsi que les enjeux humains. Mon stage a débuté le 29 avril 2024 (fin prévue le 31 octobre 2024) et s'intitule : "Interprétabilité des modèles d'apprentissage automatique pour expliquer les prévisions de consommation électrique".

H.1 Fiche Identité de l'entreprise et de l'unité d'accueil

EDF (Electricité de France) est une entreprise française, détenue à 100% par l'État, dont la mission est de produire et fournir le territoire en électricité. Elle est le leader mondial en énergie bas carbone, et premier producteur et fournisseur d'électricité en France ainsi qu'en Europe. Elle est notamment réputée pour son nucléaire, qui représente environ 80% de sa production et 70% du mix énergétique français. Son chiffre d'affaires était de 140 milliards d'euros en 2023 et sa masse salariale d'environ 180 000 employés. Cela inclut ses différentes filiales telles que Enedis et Framatome. La R&D d'EDF comporte quant à elle 2000 employés, et représente environ 1% des dépenses du groupe, c'est-à-dire un budget de 700 millions d'euros. Elle a pour but de réaliser des études et des outils pour les différentes unités opérationnelles et décisionnelles, afin de contribuer à la performance de l'entreprise ainsi qu'à éclairer sur l'avenir, tant du point de vue de la croissance que de celui de la décarbonation. Elle est notamment réputée pour ses plateformes de tests et de simulations, parmi les plus modernes et efficaces dans le secteur de l'énergie. La R&D est répartie dans 9 centres de recherche dont 3 en France : Chatou, les Renardières et Paris-Saclay. Mon stage lui se déroulait sur le plateau de Saclay, à Palaiseau.

Durant ce stage, j'étais rattaché au département OSIRIS (Optimisation, SImulation, RIsques, et Statistiques). Sa mission principale est de développer des logiciels d'aide à la décision, pour prévoir les besoins énergétiques et optimiser la production électrique, sur des horizons de l'infra-journalier au long terme (20-30 ans). Le département regroupe environ 150 salariés. Au sein de ce département, différentes équipes coexistent. Certaines se focalisent sur les risques financiers et le trading sur le marché de l'énergie, d'autres sur l'optimisation des moyens de production ; le groupe R39 dans lequel je me trouve a pour mission de prédire la consommation moyen et court terme des entreprises et des particuliers. C'est un enjeu crucial pour le groupe, étant donné que l'électricité ne se stocke pas et que la production doit donc être finement calibrée sur la demande. J'étais plus particulièrement sous la tutelle de Margaux Brégère, membre de R39 ainsi qu'enseignante chercheuse associée à la Sorbonne. Le projet du stage s'inscrit dans la volonté de R39 d'approfondir ses connaissances et outils en interprétabilité, compétence fondamentale pour avoir du recul sur les prédictions fournies par les modèles.

Description détaillée de l'équipe R39:

- Manager principal : BENARBIA Ikrame (ikrame.benarbia@edf.fr)
- Taille : 20 permanents + un nombre variable de stagiaires et thésards (5 actuellement)
- Clients : principalement des filiales et entités opérationnelles du groupe, e.g ENEDIS (filiale EDF chargée de la gestion et de l'aménagement du réseau), DOAAT (Direction Optimisation Amont Aval Trading d'EDF), DCO (Direction Commerciale et Opérationnelle d'EDF)

L'équipe est divisée en deux, avec d'un côté ceux réalisent les études qui correspondent aux demandes des différents clients (différents horizons temporels, granularités spatiales et types de modèles), et de l'autre les chefs de projets, qui encadrent ces études. Chacun peut être amené à travailler sur plusieurs projets. Par ailleurs, certains membres encadrent des stagiaires ou des thésards (e.g. Margaux). Les thésards sont toujours co-encadrés par EDF et par un laboratoire public tel que l'INRIA. Un stagiaire est quant à lui rattaché à l'un des projets de l'équipe, pour ma part c'est l'Energy Data Lab, qui est un projet "corporate", c'est-à-dire dont le client est la R&D elle-même.

H.2 Diagnostic managérial de l'unité opérationnelle

La culture managériale au sein du groupe dépend grandement de l'échelle d'observation. Dans le cadre de mon stage, les relations hiérarchiques auxquelles je suis réellement confronté sont l'encadrement de mon stage par ma tutrice, ainsi que le rôle de chef d'équipe exercé par Madame Benarbia. À noter que nous appelons tous cette dernière par son prénom, Ikrame. Tout le monde se tutoie à la R&D, sauf éventuellement avec le chef de département et lors des entretiens RH. Cette proximité est en fait une des valeurs clés de l'équipe ; Ikrame me l'a fait comprendre dès l'entretien. Elle souhaite une ambiance où tout le monde se sent bien, et où les gens communiquent en face à face plutôt que de rester cloîtrés dans leurs bureaux. Ce sont vraiment les valeurs humaines sur lesquelles elle a insisté, en dehors des compétences techniques requises.

De fait, j'ai en effet observé une forte cohésion de groupe. Il y a régulièrement des événements d'équipe organisés ; par exemple les JOsiris (jeux olympiques à l'échelle du département), ou bien la sortie d'équipe où nous avons enregistré une chanson dans un studio réputé à Paris. De façon plus régulière, nous nous retrouvons en général aux alentours de 9h pour prendre le café tous ensemble. Ces moments informels sont une composante cruciale au bon fonctionnement du groupe. C'est en effet durant ces échanges qu'on apprend à mieux se connaître, qu'on peut se mettre à jour sur les différentes actualités ; c'est aussi l'occasion pour moi de me renseigner plus précisément sur ce que font les autres, sur certaines questions administratives, etc.

L'équipe se retrouve aussi régulièrement en dehors du lieu de travail, pour aller faire du sport par exemple, ou en afterwork pour célébrer la promotion d'un collègue. Y participent en général des membres d'autres départements aussi. C'est le genre d'événements qui participent à la motivation générale.

Niveau organisation hiérarchique, il s'agit d'un système dit matriciel, ou "par projet". Les chefs de projets ont la seule autorité sur les études qu'ils gèrent, pour lesquelles ils décident des jours alloués, en fonction de l'étendue du sujet, des compétences à mettre en jeu, et des attentes du client. Ils doivent ensuite piloter l'étude et rendre compte des avancées (à base de livrables sous forme de slides, de packages de code, etc). Nous avons un rendez-vous "actualités" hebdomadaire où tout le monde remonte les points qu'il souhaite mettre en avant, ainsi que des "learnings" de façon plus sporadique où il est question de présenter ses travaux aux restes de l'équipe. Ikrame se charge de suivre le planning général du groupe et de faire le lien avec le reste du département. J'ai tout de suite compris que son rôle était de soutenir l'équipe avant tout. En cas de soucis quelconque (éventuel retard sur un projet, relations compliquées avec un interlocuteur, etc), elle prend tout de suite une posture rassurante et se met par défaut de notre côté.

La seule friction que j'ai pu observer durant mon stage a eu lieu entre Ikrame et un thésard, après que celui-ci l'ai corrigée un peu trop directement lors d'une réunion et ne s'est jamais proprement excusé par la suite. De ce que j'ai compris, les deux considèrent l'autre en faute et la situation ne s'est donc jamais vraiment résolue. Je pense que ce genre de situation est inévitable car on ne peut pas s'entendre avec tout le monde. Fort heureusement, cela n'a pas impacté le groupe ; j'ai remarqué cependant que le thésard vient moins souvent sur site. De mon point de vue, c'est lui qui est en tort car même si l'ambiance se veut la plus amicale possible, Ikrame reste sa supérieure hiérarchique. Je pense aussi néanmoins que celle-ci devrait avoir le recul pour lâcher prise.

Quoi qu'il en soit, il n'y a pas au jour le jour de relations hiérarchiques à proprement parler. Chacun est par exemple autonome dans sa gestion du travail. En effet, en dehors des stagiaires et des thésards, les membres de l'équipe doivent imputer eux-mêmes leurs jours à tel ou tel projet (même s'ils avancent en réalité sur différents projets à la fois). De même, la gestion de ses horaires et du télétravail est très flexible. Sur le papier, il s'agit d'un forfait jour et il est attendu de travailler 40 heures par semaine, c'est-à-dire 8 heures par jour. Le télétravail est autorisé depuis le COVID avec un minimum d'un jour sur site par semaine et 50% sur le mois. Il existe une plateforme sur laquelle on doit alors déclarer ses journées, et l'on badge par ailleurs pour entrer et sortir des bâtiments. Mais dans les faits, Ikrame fait confiance à l'équipe, et il est tout à fait acceptable de rentrer plus tôt sur les journées moins productives, quitte à rester plus tard certains soirs.

Nous ne sommes absolument pas fliqués sur le temps de travail réellement effectué, et c'est même communément admis que certaines semaines - typiquement pendant l'été - seront moins productives que d'autres. Cette flexibilité me plaît beaucoup, et je pense qu'elle est nécessaire dans la recherche. J'ai pour ma part l'habitude depuis longtemps de travailler dans les transports, pour lire de la bibliographie par exemple, ou bien tard le soir pour faire tourner des modèles pendant la nuit. Ce n'est pas inhabituel dans ce milieu où les chercheurs sont avant tout des passionnés, et par ailleurs en IA où les modèles doivent tourner même en dehors des horaires de travail habituels. Le télétravail et les horaires flexibles permettent ainsi de compenser ces écarts. C'est aussi un moyen de s'assurer un bon équilibre entre vie au travail et vie en dehors, surtout lorsque l'on a des contraintes de sommeil, de transports, de vie de famille, etc.

Pour autant, il faut évidemment s'assurer que le travail est tout de même bien fait. Chaque groupe se réunit en fait de manière régulière pour discuter de l'avancée des différents projets. Pour ma part, nous avons une réunion hebdomadaire avec Margaux, durant laquelle je lui montre les différentes expériences que j'ai réalisées ou je lui résume les différents articles que j'ai lus. Elle m'indique alors les pistes à suivre, me challenge sur certains points théoriques, etc. Néanmoins, elle me laisse aussi beaucoup d'autonomie, car le sujet est vaste et il y a une infinité de manières de l'aborder. C'est sous mon impulsion qu'on a décidé par exemple d'expérimenter avec différentes approximations des "valeurs de Shapley" (l'une des méthode d'explicabilité que l'on étudie). Il s'agit-là d'une composante peut-être unique à la recherche, à savoir le fait de travailler aussi pour soi plutôt qu'uniquement pour l'entreprise. Les agents permanents de la R&D ont certes des clients avec des deadlines imposées, mais le but est aussi de produire des avancées scientifiques de qualité (ça l'est encore plus pour les thésards). Cela se traduit par la publication d'articles, par la participation à diverses conférences, etc. Il n'y a pas en général d'objectifs à court-terme de rentabilité (même s'il est évident qu'EDF investit aussi dans la recherche avec une vision stratégique en tête).

Un point qu'il peut être intéressant de creuser d'un point de vue managérial est la gestion de la motivation. Comme dans toute entreprise, celle-ci est en partie liée à la rémunération. Contrairement au milieu académique où celle-ci est généralement assez basse, EDF paye plutôt bien ses chercheurs. Il y a de nombreux avantages (notamment la réduction de la facture électrique), et on peut recevoir des primes en fonction des performances de l'entreprise et de l'équipe. Il existe une grille de salaire où chaque ligne correspond à un niveau de rémunération (NR) et les colonnes à un échelon. Le NR est établi au préalable selon sa formation et ses expériences. À noter que la plupart des employés de la R&D proviennent d'écoles d'ingénieurs (e.g. ENSAI, Télécom, mais aussi ENS), c'est alors le rang de l'école qui impacte le NR. Je trouve ce système assez archaïque, car l'école n'est pour moi pas représentative des connaissances réelles.

Le NR dépend aussi bien sûr du rôle dans l'entreprise. Un manager aura donc par exemple un NR plus élevé qu'un agent permanent, lui-même ayant un NR plus élevé qu'un thésard. Enfin, les échelons évoluent avec l'ancienneté dans l'entreprise et sous l'éventuelle impulsion du manager, comme moyen de récompense. De ce que j'ai compris, les salaires de départ sont plutôt élevés pour le secteur, mais leur augmentation au cours du temps (même maximale) est considérée assez faible. En général, un employé va plutôt essayer d'évoluer dans l'une des trois orientations possibles : l'expertise où l'on se spécialise dans un domaine en particulier et on devient alors référent sur ce sujet, la gestion de projets où l'on est amené à prendre en charge des études sur des domaines plus variés, et le management pour monter dans la hiérarchie et devenir par exemple chef de département.

Bien sûr, la rémunération ne peut pas expliquer à elle-seule la motivation des chercheurs à EDF, car il serait tout à fait possible de l'augmenter drastiquement en passant dans le privé, chez Total par exemple, voire dans une boite de finance pour ceux qui font du trading. Indéniablement, il y a un confort à EDF qu'on ne retrouve pas ailleurs. Celui-ci peut être ressenti à différentes occasions : les horaires flexibles, l'autonomie, les sujets captivants, et peut-être aussi la sécurité de l'emploi. Pour ce dernier point, je pense qu'il est en partie dû à la position particulière d'EDF. C'est tout d'abord un grand groupe privé, grâce auquel la R&D jouit d'un grand site moderne, muni de nombreux équipements, qui propose de nombreux avantages et une certaine pérennité dans le temps.

C'est aussi une entité entièrement dépendante de l'État (celui-ci détenant 100% des parts de l'entreprise), qui ne subit donc pas de grands programmes de licenciements dans l'unique but de satisfaire des actionnaires. De fait, on s'y sent donc important mais aussi à l'abri. Tout cela rend possible des carrières longues et variées.

Enfin, EDF est avant tout une entreprise utile, qui œuvre pour le bien commun : à la fois du pays et de la planète. La production d'électricité et la décarbonation font partie des enjeux cruciaux tant pour la souveraineté nationale que pour la lutte contre le réchauffement climatique. Ces raisons écologiques et patriotiques sont justement parmi celles qui m'ont poussées à postuler, et je pense que cela donne du sens au travail réalisé tous les jours par la plupart des employés de l'entreprise. Ces thématiques sociétales sont d'ailleurs assez présentes au jour le jour : nous recevons par mail des revues hebdomadaires qui nous mettent à jour sur les réalités économiques et géo-stratégies du groupe, de nombreuses pratiques ont été mises en place pour l'écologie telles que utilisation de mugs individuels pour le café, le tri des déchets, les incitations pour venir en transports en commun ou vélo, les aides à l'achat de véhicules électriques, etc. J'ai aussi observé un bel effort de parité femmes-hommes (en particulier dans mon équipe), ainsi que la présence de profils d'origines variées. L'inclusivité n'est donc pas négligée. Nous avons par ailleurs des syndicats qui se battent avec ténacité pour nos droits et qui nous tiennent au courant en continu des débats actuels (ce qui à tendance d'ailleurs à remplir assez vite la boite mail...).

Le point sur lequel je serais peut-être le moins positif est la conduite du changement, qui a lieu à différentes échelles dans l'entreprise. À l'échelle la plus globale, l'entreprise s'est donné une vision stratégique "cap 2030", dans laquelle il est question de bien-être des clients, de stratégie internationale et de neutralité carbone. Chaque projet se doit d'être plus au moins rattaché à ces grandes lignes. Néanmoins, il est souvent difficile de voir exactement où se rattache chaque étude, et nous n'avons pas de retours sur la progression réelle de ces objectifs. Il serait intéressant de détailler des plans de route et de mettre en avant des métriques, afin que chacun puisse se sentir réellement concerné.

À l'échelle du site à Saclay, il existe par ailleurs une plateforme sur laquelle on peut faire remonter des dysfonctionnements (e.g un parking vélo non sécurisé, un store dysfonctionnel, etc). De ce que j'ai observé, les délais pour les réparations sont assez élevés et les agents se plaignent d'un manque de retour quant au traitement des demandes. Il existe aussi une plateforme pour déposer des avis, tant positifs que négatifs, sur la qualité de vie au travail et les actions menées pour l'environnement. Il y a entre autres un atelier annuel du changement encadré par des consultants externes, durant lesquels les agents sont invités à discuter des points à améliorer à la R&D, après lequel un livrable est produit sous forme d'un cahier des charges décrivant les changements prioritaires à mettre en place. Il s'avère cependant que ces points ne sont rarement vraiment adressés. Le chef de département a qualifié certaines exigences de "demandes au père noël", ce qui a été très mal reçu par certains employés. D'autres demandes, telles que l'envie d'être formé sur les innovations de demain sont probablement à l'origine des formations obligatoires "data & IA" que nous avons subies, produites par des prestataires externes et en réalité assez "pipeau" (il existe cependant des formations optionnelles plus pointues si l'on cherche bien). Je constate donc qu'EDF a la bonne idée de vouloir impliquer les salariés dans l'amélioration continue de l'entreprise, mais il faudrait aller au bout du processus et mettre en pratique de façon durable les améliorations identifiées.

Enfin à l'échelle de l'équipe, la plupart des changements sont conduits par Ikrame. Ceux-ci sont souvent menés de manière informelle. Par exemple, il y a eu un récent changement de bureau imposé aux thésards par le département, annoncé par Ikrame en réunion, après lequel j'ai senti qu'il y a eu quelques frustrations. En effet, les thésards commençaient à s'habituer à leur précédent bureau, et seraient maintenant bien plus loin des autres. Ils ont fini par faire remonter leur mécontentement autour du café puis lors de la réunion suivante, ce à quoi Ikrame a répondu en détaillant les raisons du changement (favoriser les échanges entre les différents thésards du département), et a proposé qu'ils reviennent dans leurs anciens bureaux d'ici une semaine si les nouveaux ne convenaient toujours pas. J'ai trouvé ce retour tout à fait honnête, même si le changement aurait sûrement été mieux accueilli s'ils avaient été mis au courant plus tôt, voire impliqués dans la décision dès le départ.

H.3 Rapport d'étonnement

Il y a plusieurs points sur lesquels j'ai été étonné durant ce stage. Premièrement, j'ai été surpris par l'innovation à la R&D, tant positivement que négativement. Un exemple est Microsoft Copilot, qui a été installé pour tous les utilisateurs. C'est une sorte de version plus sécurisée de ChatGPT, c'est-à-dire un grand modèle de langage que l'on peut questionner dans tous les domaines, faire rédiger des textes, etc. Vu la réticence de certains face à cette nouvelle technologie, il n'était pas évident qu'un grand groupe (réputés pour leur inertie face au changement) l'adopte aussi vite. Je pense personnellement qu'il est en fait essentiel pour les entreprises d'être à jour sur les dernières évolutions afin de rester compétitives. Ici, il est nullement question de remplacer des employés pour réduire les coûts ; il s'agit simplement d'un outil pour faciliter - et donc améliorer - le travail réalisé. Pour ma part, je l'ai par exemple utilisé pour comprendre certains scripts codés en R, langage de programmation que je ne maîtrisais pas encore.

Pour autant, les technologies au cœur de la R&D, utilisées par exemple pour prédire la consommation électrique, sont quant-à-elles un peu veillottes à mes yeux. En effet, ce n'est que récemment que les réseaux de neurones ont commencé à être étudiés, et ils ne sont clairement pas priorisés vis à vis des méthodes linéaires plus anciennes. Je pense que ce choix s'explique en partie par la spécialisation plutôt statistique de la majorité des agents (plutôt qu'en machine learning, qui est en fait un domaine assez récent). La plupart codent par exemple en R plutôt qu'en python (langage aujourd'hui majoritairement utilisé en IA). Mais c'est aussi justifié par le fait que les gains de performances des réseaux de neurones pour la prédiction de séries temporelles ne sont pas encore avérés. Les méthodes linéaires étant plus simples à entraîner, et davantage explicables, il n'y a pas eu de réelle incitation à se ré-orienter entièrement vers le deep learning.

Il faut aussi ne pas oublier que la R&D est l'entité la plus à la pointe de l'innovation du groupe. Un collègue m'a raconté son parcours dans les équipes finances, où les notions élémentaires d'informatique n'étaient pas toujours maîtrisées. Le nombre de paiements en doubles qui passaient inaperçus car les factures étaient traitées manuellement est bluffant. J'ai en fait été assez surpris par le faible partage d'information entre les différentes entités du groupe. En effet, chaque entité d'EDF travaille indépendamment, avec ses propres contraintes, ses propres clients, ses propres comptes à rendre. Parfois, certaines filiales n'acceptent pas de partager leurs données pour des raisons de confidentialité ou d'enjeux compétitifs. Le partage de connaissance globalement m'a paru peu répandu. Certaines équipes du département découvraient par exemple les modèles que nous utilisions alors que leurs sujets se recoupent avec les nôtres, et inversement.

Quoi qu'il en soit, j'ai été agréablement surpris par la bonne ambiance au jour le jour dans l'équipe, et ce même avec le distanciel très répandu. En effet, il y a des jours (le vendredi notamment) où beaucoup d'agents choisissent de télétravailler. C'était un peu surprenant au début, notamment durant la période initiale où je n'y étais pas encore autorisé. Néanmoins, l'équipe est assez grande et il y a au final toujours au moins une personne présente sur site. En fait, j'ai finalement beaucoup apprécié cette variabilité, cela permettait de manger et de discuter avec des personnes différentes chaque jour. Certains agents s'absentent aussi parfois pour se rendre en conférence ou bien donner des cours dans les écoles. Le quotidien du chercheur ne se résume donc absolument pas à travailler seul dans son bureau et on est en fait souvent amené à bouger. Même au sein du site, des séminaires facultatifs ont lieu pour en apprendre plus sur des domaines annexes. J'ai par exemple participé à des conférences sur la théorie des jeux avec Orange, et à la journée de IA pour l'environnement dans les locaux de Centrale. C'était très enrichissant.

À la suite de cette expérience de stage en R&D, j'ai décidé de poursuivre en thèse avec EDF. J'ai en effet trouvé un sujet qui me plaisait (toujours en lien avec la prévision de consommation) et les conditions de travail m'ont convaincues. Ce sera en partenariat avec l'INRIA de Sophia Antipolis ; cela me poussera à régulièrement me déplacer ce qui me convient bien. Le stage m'aura par ailleurs conforté dans le fait que je veux continuer à travailler sur des sujets théoriques encore un certain temps, sur les séries temporelles en particulier, mais j'ai aussi pris goût à l'étude des enjeux géostratégiques liés à l'énergie. La culture à EDF m'a plu et je ne serais pas contre l'idée de continuer à travailler chez eux après.

I Auto-évalution des compétences

Compétence	Points forts	Points faibles	Pass / Faible
Complexité	Analyse de la chaîne de production et de distribution de l'électricité, en particulier les parties prenantes liées à la prévision de consommation. Modélisé un modèle d'interprétabilité pour répondre aux besoins de l'équipe et pouvoir l'intégrer dans leur pipeline. Contraintes de temps de calcul, de robustesse, de clarté, adressées au travers de mon package étudié et documenté.	Pas encore de retours des unités opérationnelles sur leurs attentes pour un tel package. Difficile de savoir si le package sera réellement utilisé.	Pass
Métier ingénieur	Développé mon expertise dans le domaine de la prédiction de série temporelles (en particulier courbes de charge) ainsi que l'interprétabilité pour le machine learning. Auto-formation sur les bonnes pratiques pour avoir un code structuré, qui s'exécute en un temps optimal, et adapté à la pipeline du groupe R&D. Capacité à rédiger un papier technique clair et bien structuré.	Environnements compliqués à mettre en place, j'ai eu beaucoup de soucis par moments.	Pass
Innover entreprendre	Propositions d'améliorations pour les algorithmes naïfs d'approximation des valeurs de Shapley. Propositions de pistes de recherche pour les expériences sur certains paramètres, notamment l'aspect conditionnel pour éviter les écueils des variables corrélées.	Re-découverte d'éléments théoriques probablement déjà étudiés en partie, mais parsemés et pas forcément appliqués au contexte de la prévision.	Pass

Compétence	Points forts	Points faibles	Pass / Faible
Création de valeur	Étude élargie sur le sujet ainsi qu'un package réutilisable par l'équipe et améliorable par la suite, pour analyser les sorties de leurs modèles et prendre du recul sur les prédictions.	Pas de création de valeur monétaire directe, ce qui est en fait assez courant en recherche. L'investissement est plus sur le long terme.	Pass
Interculturel	Lecture et rédaction de bibliographie en anglais, échanges avec certains intervenants étrangers.	Pas pu participer à des conférences internationales, mais cela viendra en thèse.	Pass
Digital	Conception de mon package selon les bonnes pratiques du domaine, avec une documentation claire et efficace. Utilisation d'outils avancés (machines à distance, copilot, …).	Pas eu d'accès aux clusters de calculs. Cela viendra en thèse.	Pass
Convaincre	Présentation de mes travaux à de nombreuses reprises devant différents interlocuteurs. Slides et articles clairs, aisance à l'oral pour vulgariser et démontrer.	Pas eu besoin de convaincre de la nécessité de mon étude pour me faire financer ou autre.	Pass
Equipe projet	Intégré au sein d'une équipe, présentation de mes travaux, échanges et entraide.	Seul à travailler sur mon étude.	Pass
Ethique soutenable	Intérêt des valeurs de Shapley pour estimer les parts climats et hors-climat de la consommation.	Pas de calcul de l'emprunte carbone de mon étude.	Pass