Ph.D. research project

Machine learning based hybrid strategies for combinatorial optimization

Keywords: combinatorial optimization, vehicle routing problem, machine learning, structured learning, implicit layers, statistical learning.

1 Context

Rungis Market is the largest fresh produce market in the world, in particular, it supplies 63% of the food consumed in Île-de-France. The company Califrais is its official digital and logistics operator through the platform rungismarket.com. With the aim of digitalizing and decarbonizing the food supply chain, Califrais develops logistics solutions based on algorithms at the crossroads of machine learning, statistics, and optimization.

Califrais has strong connections with public research and works with academic partners such as the Laboratoire de Probabilités, Statistique et Modélisation (LPSM), Sorbonne Université and the CNRS. This privileged context has led to numerous publications in leading conferences of machine learning, such as ICML and NeurIPS.

We are offering this Ph.D. position as part of a collaboration between Califrais and the LPSM. Its aim is to study recent tools at the intersection of machine learning and combinatorial optimization that allows to jointly "learn and optimize". It is funded by the French Agency for Ecological Transition (ADEME) under the call "Logistique 4.0" won by Califrais and Sorbonne Université.

2 Description of the subject

Califrais has a history of several years of daily order and delivery data for thousands of fresh products and hundreds of customers. Extracting information from this large amount of data is critical for a variety of tasks: predictive analysis of demand or customer satisfaction, inventory optimization of perishable goods, or routing optimization. However, these data present many challenges and domain-specific characteristics; in particular, they have a lot of noise and non-stationarity. This is due to the specificity of fresh products: many external factors affect both supply and demand of these products, such as weather, shortages, political and global economic factors, etc.

In this project, we are interested in problems that fall into the domain of (discrete) optimization, or decision problems. Typically, in inventory problems, the decision consists in deciding which quantity to order on a certain day, given a state of the inventory and some costs to optimize (Nahmias, 2011). Another class of problems of interest are routing problems (Toth and Vigo, 2002). The objective is then to design an algorithm that, given a list of client positions and quantities to deliver, decides the number of vehicles required for the tour and which client each vehicle visits. This is a NP-hard combinatorial problem, with many variants depending on the constraints on the problem (typically, whether we add capacity constraints on the vehicles, time intervals for delivery...).

In both examples, it is clear that some quantities in the optimization problem are unknown. In inventory problems, to make an optimal decision on a given day, we want to estimate future demand. Similarly, in routing problems, travel times are unknown, but they are crucial quantities in the optimization, in particular in the presence of time window delivery constraints. The problem then is: how can we account for these uncertainties in the optimization?

Let us denote by $x \in \mathcal{X}$ an instance of a problem (think of a list of clients to deliver), by $y \in \mathcal{Y}$ a solution (think of a permutation of these clients), and by $c(\theta, y)$ the cost of a solution y, parametrized by some unknown θ, that we model as a random variable (for example, the cost depends on the delays which are unknown in advance). For a given problem $x \in \mathcal{X}$, the goal is then to minimize the expected cost, that is,

$$\min_{y \in \mathcal{Y}} \mathbb{E}_\theta \big[c(\theta, y) \,|\, x \big]. \tag{1}$$

To do this, the traditional procedure is to take a two-step approach: first predict the unknown quantities, and then optimize (see, e.g., Wang et al., 2012; Elmachtoub and Grigas, 2022). Typically, given a dataset of pairs $\{(x_i, \theta_i)\}_{i=1,\dots,n}$, we define a statistical model on the parameters θ. In the case of a parametric

model, let $w \in \mathbb{R}^d$ and $\ell : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^+$ be a loss function on $\mathbb{R}^d$ (for example, the quadratic loss), we then denote by h_w the function that maps x to θ: $\theta = h_w(x)$. For a new instance x, the optimization problem (1) is then replaced by the deterministic problem

$$\min_{y \in \mathcal{Y}} c(h_w(x), y). \tag{2}$$

We can summarize the "learn then optimize" pipeline as follows:

$$\text{Input } x \in \mathcal{X} \quad \to \quad \text{Parameter } \widehat{\theta} = h_{\widehat{w}}(x) \quad \to \quad \text{Solution } y \in \operatorname*{argmin}_{y \in \mathcal{Y}} c(\widehat{\theta}, y), \tag{3}$$
$$\text{where } \widehat{w} \in \operatorname*{argmin}_{w \in \mathbb{R}^d} \sum_{i=1}^{n} \ell\big(\theta_i, h_w(x_i)\big).$$

This approach is unsatisfactory for several reasons. First, the predicted quantities are treated as true quantities by the optimization algorithm. In other words, the uncertainty of $\widehat{\theta}$ is not taken into account. When dealing with noisy signals, which is especially the case in food supply chain applications, this leads to an optimization step that overfits the prediction errors. Moreover, this approach is suboptimal in the sense that it does not allow to learn some structure of the problems at hand, that is, to exploit some information from a dataset $\{x_i\}_{i=1,\dots,n}$, but always works in a "worst case" fashion. More precisely, the algorithms typically used to solve (2) are designed to work whatever the parameters $h_w(x)$, whereas we typically only want them to work on an instance x drawn from the same distribution as the set of past problems $\{x_i\}_{i=1,\dots,n}$.

Therefore, it is natural to turn to approaches that tackle both learning and optimization tasks together. This is a rich and diverse field; see, for example, Donti et al. (2017); Bengio et al. (2021); Bai et al. (2023); Mandi et al. (2024); Sadana et al. (2025); Vivier-Ardisson et al. (2024). A new and promising direction is the use of combinatorial optimization layers (Dalle et al., 2022; Baty et al., 2024). The idea is to modify the pipeline (3) to include the optimization part in the loss function used to learn $\widehat{w}$. More precisely, let f be the function that solves (2) given a parameter θ, i.e.,

$$f : \theta \mapsto \operatorname*{argmin}_{y \in \mathcal{Y}} c(\theta, y),$$

and $\mathcal{L} : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}^+$ be a loss function on the solution space. Then, the "learn and optimize" pipeline can be summarized as follows:

$$\text{Input } x \in \mathcal{X} \quad \to \quad \text{Parameters } \widehat{\theta} = h_{\widehat{w}}(x) \quad \to \quad \text{Solution } y = f(\widehat{\theta}), \tag{4}$$
$$\text{where } \widehat{w} \in \operatorname*{argmin}_{w} \sum_{i=1}^{n} \mathcal{L}\big(y_i, f(h_w(x_i))\big).$$

The main difference is that the optimal parameter $\widehat{\theta}$ is now chosen given a dataset of problem and solution pairs $\{(x_i, y_i)\}_{i=1,\dots,n}$ (instead of a dataset of problem and parameter pairs $\{(x_i, \theta_i)\}_{i=1,\dots,n}$). In particular, it is chosen taking into account the choice of the optimization algorithm f. Indeed, in practice we do not have exact solutions of (2), but we use heuristic solvers that give approximate solutions. Now the learning algorithm will depend on the choice of the heuristic f.

However, the difficulty now is that in order to solve (4) via classical gradient-based approaches, it is necessary to be able to differentiate the function f, which is usually not possible for classical combinatorial optimization solvers, since their gradient is almost always zero. Most of the efforts in the literature have focused on extending standard learning approaches to this context, via perturbation approaches or regularization, yielding differentiable combinatorial optimization functions f, known as "CO layers" (Berthet et al., 2020; Blondel et al., 2020; Dalle et al., 2022).

There are many open questions in this new but growing field. In this Ph.D. project, we propose several research perspectives, both theoretical and methodological.

1. A first methodological question is to apply the "learn and optimize" pipeline (4) to the routing problem at Califrais, which is known to be particularly complicated from a combinatorial point of view due to the presence of time window constraints and stochastic travel times. In this case, a problem $x \in \mathcal{X}$ can be modeled as a graph, i.e., $x = (\mathcal{V}, \mathcal{A})$, where $\mathcal{V}$ denotes the set of vertices ($|\mathcal{V}| = m$) and $\mathcal{A}$ the set of edges. Then the goal is to partition this graph into different routes, each

corresponding to a vehicle. One of the vertices of the graph corresponds to the depot (typically, in the Califrais application, Rungis) from which all routes must start and return, and every other vertex corresponds to a customer. Each vertex is associated with a target delivery time window $[a_i, b_i]$. The goal is to minimize the travel time under the constraint that the delivery at node i must occur within the time window. The difficulty is that the time to get from one vertex to another is unknown a priori, which is where machine learning tools come in. Formally, if we denote by $\mathcal{Y} \subset \{0,1\}^{m \times m}$ the set of feasible solutions, where a solution $y \in \mathcal{Y}$ is represented as: $y_{ij} = 1$ if a route goes from node i to node j, then, theoretically, we want to minimize

$$c(\theta, y) = \sum_{(i,j) \in \mathcal{A}} \theta_{i,j}(y) y_{i,j}, \tag{5}$$

where $\theta_{i,j}(y)$ is the travel time to go from node i to node j for the solution y. The challenge here is that the travel time depends on the time of day, that is, it depends on all the other vertices visited before node j on the route, not just the previously visited node i. In other words, we do not have a cost of the form $\theta_{i,j}$, but a cost $\theta_{i,j}(y)$. A first step will be to find an adequate model for this learning problem, together with an efficient optimization layer. A starting point will be the work by Baty et al. (2024), which explores a related dynamic vehicle problem.

2. Having familiarised with these tools and implemented an initial solution for the vehicle routing application, we will take a step back and look at the theoretical guarantees associated with pipelines of the form (4). Several directions can be undertaken, such as combining statistical learning guarantees on the ML prediction algorithm with results on the combinatorial optimization side, or investigating other learning paradigms such as reinforcement learning or active learning. A starting point will be the results of Aubin-Frankowski et al. (2024), who provide generalization bounds for a similar problem in a reinforcement learning setting. They have focused on linear optimization problems, i.e., of the form $c(\theta, y) = \langle \theta, y \rangle$ and assume that they have access to an oracle that solves (2) exactly. A first extension is to investigate how the optimization error can be included in the results, which would give practical guidance on how to balance some trade-offs between computational time for the optimization part and for the statistical part. Another direction is to extend the cost structure to include non-linearities. This is particularly interesting since the costs used for the vehicle routing problem with time windows (5) are not a priori of the form $\langle \theta, y \rangle$.
3. Another interesting direction is to investigate the use of CO layers for inventory problems, which, to our knowledge, has not been explored in the literature, despite the existence of several works on data-driven inventory optimization (Hihat et al., 2023; Qi et al., 2023). Both theoretical and practical contributions can be made in this direction.

3 Practical conditions

This Ph.D. is part of a collaboration between Califrais and the LPSM at Sorbonne Université, funded by a grant from ADEME.

- Location: the internship will take place in both locations (Paris 75005 and Paris 75010).
- Starting date: Autumn 2025.
- Duration: 3 years.
- Profile: M2 research in applied mathematics or third year of engineering school.
- Grant: 2 200 €/month before tax.

The supervising team will consist of:

- Adeline Fermanian, Califrais.
- Maxime Sangnier, Sorbonne Université.

4 Application

The application file containing:

- a resume;

- a cover letter;
- a transcript of grades (Bachelor's and Master's degrees);

should be sent by mail to adeline.fermanian@califrais.fr and maxime.sangnier@sorbonne-universite.fr.

References

P.-C. Aubin-Frankowski, Y. De Castro, A. Parmentier, and A. Rudi. Generalization bounds of surrogate policies for combinatorial optimization problems. *arXiv preprint arXiv:2407.17200*, 2024.

R. Bai, X. Chen, Z.-L. Chen, T. Cui, S. Gong, W. He, X. Jiang, H. Jin, J. Jin, G. Kendall, et al. Analytics and machine learning in vehicle routing research. *International Journal of Production Research*, 61(1): 4–30, 2023.

L. Baty, K. Jungel, P. S. Klein, A. Parmentier, and M. Schiffer. Combinatorial optimization-enriched machine learning to solve the dynamic vehicle routing problem with time windows. *Transportation Science*, 2024.

Y. Bengio, A. Lodi, and A. Prouvost. Machine learning for combinatorial optimization: A methodological tour d'horizon. *European Journal of Operational Research*, 290(2):405–421, 2021.

Q. Berthet, M. Blondel, O. Teboul, M. Cuturi, J.-P. Vert, and F. Bach. Learning with differentiable pertubed optimizers. *Advances in neural information processing systems*, 33:9508–9519, 2020.

M. Blondel, A. F. Martins, and V. Niculae. Learning with fenchel-young losses. *Journal of Machine Learning Research*, 21(35):1–69, 2020.

G. Dalle, L. Baty, L. Bouvier, and A. Parmentier. Learning with combinatorial optimization layers: a probabilistic approach. *arXiv preprint arXiv:2207.13513*, 2022.

P. L. Donti, B. Amos, and J. Z. Kolter. Task-based end-to-end model learning. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

A. N. Elmachtoub and P. Grigas. Smart "predict, then optimize". *Management Science*, 68(1):9–26, 2022.

M. Hihat, S. Gaïffas, G. Garrigos, and S. Bussy. Online inventory problems: beyond the iid setting with online convex optimization. *Advances in Neural Information Processing Systems*, 36:20421–20440, 2023.

J. Mandi, J. Kotary, S. Berden, M. Mulamba, V. Bucarey, T. Guns, and F. Fioretto. Decision-focused learning: Foundations, state of the art,benchmark and future opportunities. *Journal of Artificial Intelligence Research*, 81:1623–1701, 2024.

S. Nahmias. *Perishable inventory systems*, volume 160. Springer Science & Business Media, 2011.

M. Qi, Y. Shi, Y. Qi, C. Ma, R. Yuan, D. Wu, and Z.-J. Shen. A practical end-to-end inventory management model with deep learning. *Management Science*, 69(2):759–773, 2023.

U. Sadana, A. Chenreddy, E. Delage, A. Forel, E. Frejinger, and T. Vidal. A survey of contextual optimization methods for decision-making under uncertainty. *European Journal of Operational Research*, 320(2):271–289, 2025.

P. Toth and D. Vigo. *The vehicle routing problem*. SIAM, 2002.

G. Vivier-Ardisson, A. Forel, A. Parmentier, and T. Vidal. Cf-opt: Counterfactual explanations for structured prediction. In *Proceedings of the 41 st International Conference on Machine Learning*, volume 235. PMLR, 2024.

T. Wang, A. Atasu, and M. Kurtuluş. A multiordering newsvendor model with dynamic forecast evolution. *Manufacturing & Service Operations Management*, 14(3):472–484, 2012.