Expert aggregation and metrics for short-term electricity price forecasting

David OBST - david.obst@edf.fr

3rd of October 2024

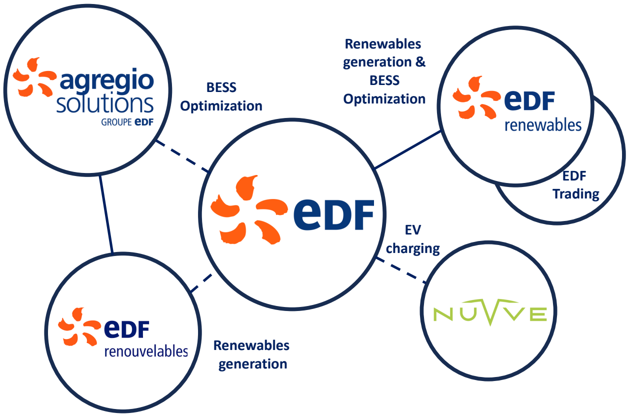
EDF R & D Saclay

Table of Contents

Industrial Context

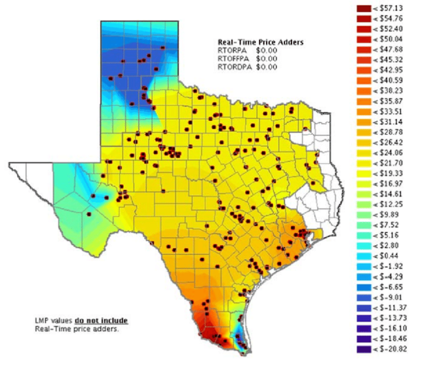
Activities of Group EDF

- Group EDF has a number of subsidiaries with different activities (renewables generation, BESS operation, EV charging...).
- Nonetheless electricity price forecasting (EPF) is relatively new topic for R&D (circa 2021).



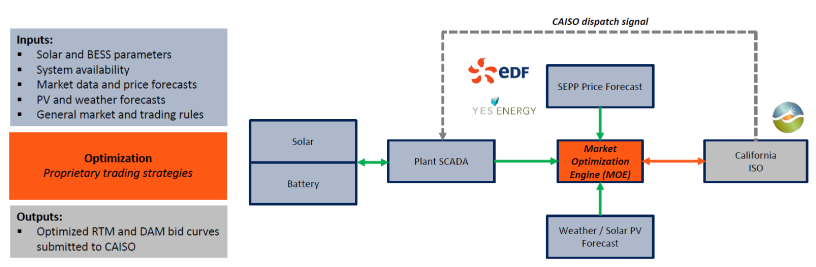
Price forecasting for EDF Renewables North America

- EDF Renewables operates 3 battery energy storage systems (BESS) in California.
- US prices are **nodal**.
- They required an **operational** tool for EPF on a day-to-day basis to help them to decide on how to optimize their battery.
- We developped a `R` library called `SEPP` (Short-term Electricity Price Prediction) for that purpose.
- `SEPP` allows for predictions on different markets: day-ahead (spot), real-time (intraday) and anciliary services.



Battery optimization at EDF RNA

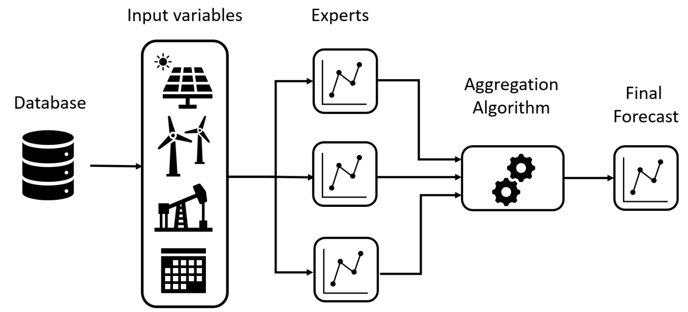
SEPP is one of the 3 main inputs of Renewables' **Market Optimization Engine (MOE)**.



In 2025: produce bid-offer curves directly from SEPP's quantile forecasts factoring revenue potential vs risk.

SEPP framework

- SEPP is a pipeline that goes from scrapping the data with the right API to performing prediction:



- SEPP uses an **ensemble of models to yield both point forecasts and quantile forecasts**.
- SEPP is used operationally by EDF RNA: hence **must remain light computationally.**
- SEPP being an operational tool makes it unconvenient for research purposes.

SEPP models

Models inputs

California data is provided by **YesEnergy**.

For day d hour h the following features are used:

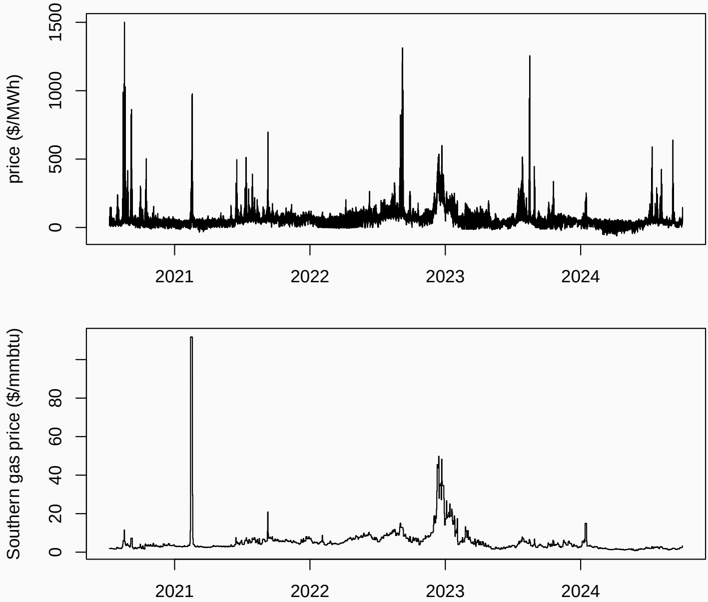
- Day-ahead prices of $d-1$ and $d-7$ for every hour.
- Citygate and SoCal gas prices of the previous day p^G_{d-1}.
- Forecasted load $\hat{L}_{d,h}$ of day d for every hour h.
- Forecasted wind generation $\hat{W}_{d,h}$ of day d for every hour h.
- Forecasted solar generation $\hat{S}_{d,h}$ of day d for every hour h.
- Calendar features (day of the week, time of the year, holidays, ...).

Continuous features are **standardized** by substracting the training-set mean and divided by the standard-deviation.

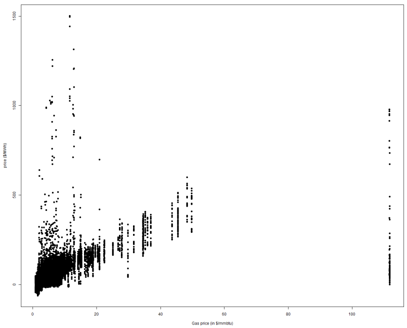
Price preprocessing

- A plethora of **variance stabilizing transforms (VST)** have been implemented in `SEPP` [Uniejewski et al., 2017]:
  - `mlog`
  - `polynomial`
  - `PIT`
- Experiments showed that depending on which VST is used, the results in terms of MAE and RMSE vary a lot.
  - PIT was better overall, but worse in highly volatile situations.
  - No transform was worse overall, but better predicted spikes.
- With the frequency of negative prices (almost daily) **are the standard VST suitable** ?
- Expert aggregation of experts using different VST has showed to be beneficial both for MAE and RMSE. **However benefits in terms of battery PnL are less clear.**

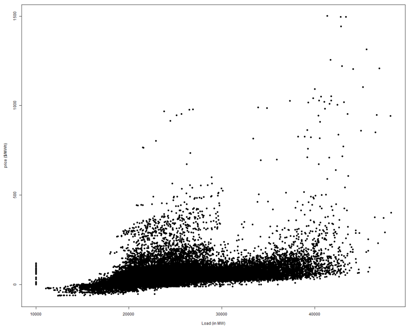
Visualization of the data



Visualization of the data (II)

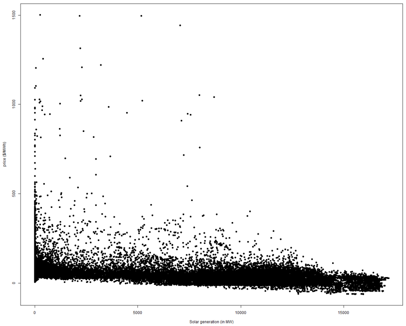


(a) Gas price.

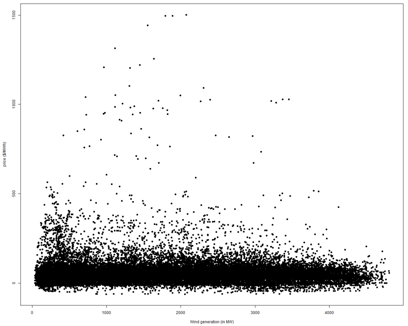


(b) Load.

Visualization of the data (III)



(c) Solar.



(d) Wind.

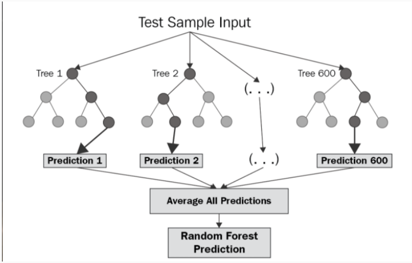
LEAR model

- One model is fitted for every instant $h \in \{1, 2, \ldots, 24\}$:

$$
\begin{aligned}
p_{d,h} = \gamma + \sum_{j=1}^{24} \beta_j^{d-1} p_{d-1,j} + \sum_{j=1}^{24} \beta_j^{d-7} p_{d-1,j} + \beta^G p_{d-1}^G + \overline{\beta}\, \overline{p}_{d-1} + \\
\underline{\beta}\, \underline{p}_{d-1} + \sum_{j=1}^{24} \beta_j^L \hat{L}_{d-1,j} + \sum_{j=1}^{24} \beta_j^S \hat{S}_{d-1,j} + \sum_{j=1}^{24} \beta_j^W \hat{W}_{d-1,j} + \\
\sum_{k=1}^{7} \beta_k^{DoW} 1(DoW_d = k) + \beta^{Hol} IsHoliday_d + \varepsilon_{d,h} \quad (1)
\end{aligned}
$$

- Model (1) is learned with the `R` library `Glmnet`.
- We consider the standard LASSO model (elastic net $\alpha = 1$). Penalty λ is obtained through cross-validation.
- Variant of (1) using GAM (Generalized Additive Models) were implemented with `mgcv` [Wood, 2017], but without improvement so far.

XGBoost and Random Forest



- Single model fitted with `instant` as variable
- Learned with the `xgboost` and `ranger` librairies.
- Hyperparameters were fixed a while ago on a hold-out set. However probably outdated.
- Interestingly enough, "vectorizing" the data (e.g. with previous day $p_{d-1,h}$ for all instant) does not significantly enhance the predictions.

Configuration of models within SEPP

The user can define as many experts as they want in a JSON file. These models can be different for each node, market, etc...

Model name	RF Long Term	RF Short Term	LASSO
Model type	RF	RF	Glmnet
Days of data	730	100	120
VST	Log	None	None
Pivot ?	No	Yes	Yes
Quantiles	$[0.01, 0.79, 0.99]$	None	-
Thresholding	$[-10, 600]$	$[-1000, 9999]$	$[0, 500]$
Hyperparams	`mtry`=3 `max.depth`=8	`mtry`=5 `max.depth`=5	- -
Features	-	-	load, gas˙price

Expert Aggregation

Motivation

- Individual predictors may have specific strengths and weaknesses, be good at certain moments and bad at others.
- The goal of expert aggregation is to achieve a more robust prediction by combining the individual predictors.
- Two types of aggregation methods are implemented in SEPP:
  - **OPERA** (Online Prediction by ExpeRt Aggregation): weights $w_t^{(i)}$ are computed each instant sequentially:

$$\hat{p}_t = w_t^{(1)}\,\hat{p}_t^{(1)} + w_t^{(2)}\,\hat{p}_t^{(2)} + \cdots + w_t^{(K)}\,\hat{p}_t^{(K)}$$

  - **Stacking**: a meta-model combine the experts according to feature values:

$$\hat{p}_t = f\big(\hat{p}_t^{(1)}, \ldots, \hat{p}_t^{(K)}, x_{t,1}, \ldots, x_{t,d}\big).$$

OPERA algorithm

- *OPERA (Online Prediction by ExpeRt Aggregation)* [Gaillard and Goude, 2016] computes weights $w_t^{(k)}$ according to the errors (ML-poly algorithm):

$$\frac{\sum_t \ell(y_t, \hat{y}_t^{(k)})}{\sum_{j=1}^{K} \sum_t \ell(y_t, \hat{y}_t^{(j)})}$$

- OPERA works both for pointwise forecast (the mean) and quantile forecast.

Stacking

- Difference in expert performance can also be explained by sensibility to covariates.
- OPERA is unable to directly take advantage of those $\Rightarrow$ model that includes the covariates in the aggregation.
- The idea of stacking is to have a meta model trained jointly on the experts and the features:

$$\hat{p}_t = f\big(p_t^{(1)}, \ldots, p_t^{(K)}, x_{t,1}, \ldots, x_{t,d}\big)$$

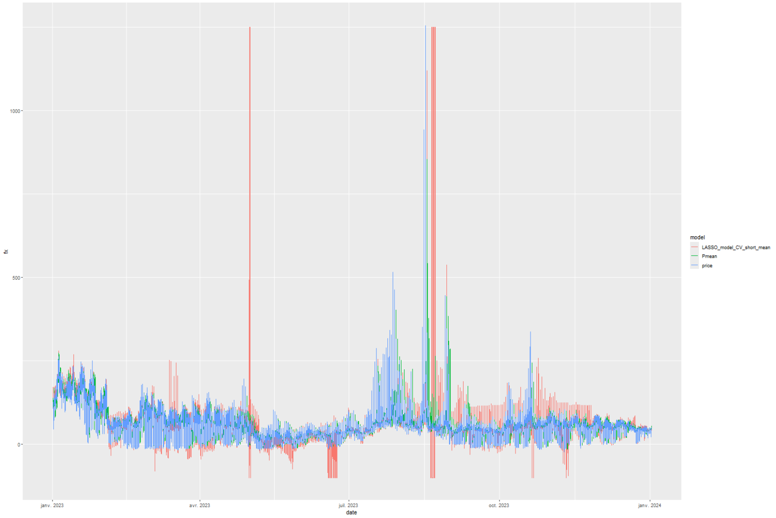
- `SEPP` has a simple random forest acting a stacking model.

Results for California

Backtesting procedure

- We applied the SEPP framework for day-ahead price prediction for the year 2023.
- Training period might differ for each model (between 60 days and 2 years).
- Models are re-trained weekly every Sunday morning.
- Evaluation metrics will be MAE (mean absolute error), RMSE (root mean squared error) and battery PnL.
- Previous experiments as well as literature [Serafin and Weron, 2024] show that high performance in MAE/RMSE might not translate into best PnL.

Prediction visualization



Results for regular metrics

Main evaluation metrics are mean absolute error and root mean squared error:

$$\mathrm{MAE} = \frac{1}{24d} \sum_{d} \sum_{h=1}^{24} |p_{d,h} - \hat{p}_{d,h}| \tag{2}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{24d} \sum_{d} \sum_{h=1}^{24} (p_{d,h} - \hat{p}_{d,h})^2} \tag{3}$$

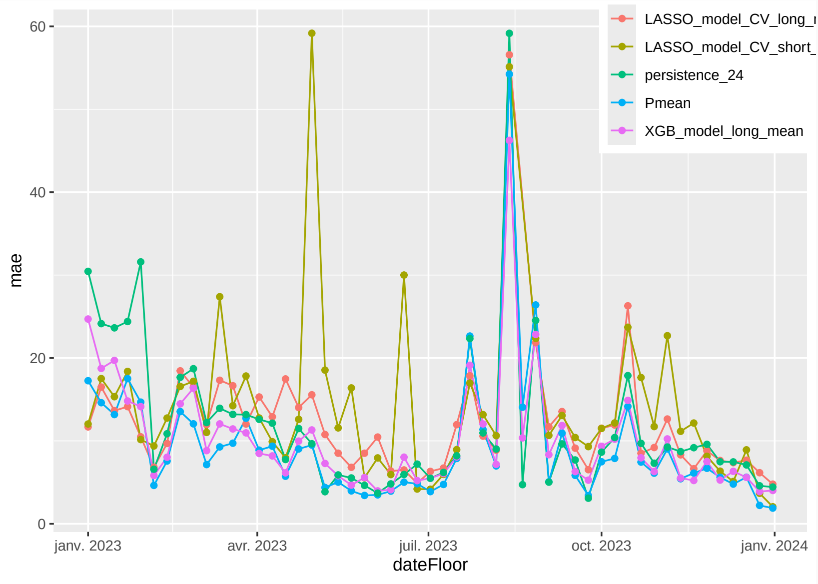
Model	MAE	RMSE
Persistence	12.1	28.8
LEAR short	18.7	97.6
LEAR long	37.9	471
XGBoost	10.3	24.5
RF	11.7	26.8
Aggreg	9.51	24.6

(e) Whole 2023.

Model	MAE	RMSE
Persistence	18.4	26.2
LEAR short	15.2	20.5
LEAR long	13.5	17.7
XGBoost	13.8	20.3
RF	16.1	23.9
Aggreg	11.8	16.3

(f) Jan to Mar 2023.

Weekly mean absolute error



BigBeau BESS characteristics

BigBeau BESS specs:

- Capacity of 140MWh
- Charge and discharge rate of 35MW
- Efficiency of 88%
- Daily mileage: 280MWh (one round-trip)
- Cycle cost of **4000$**.

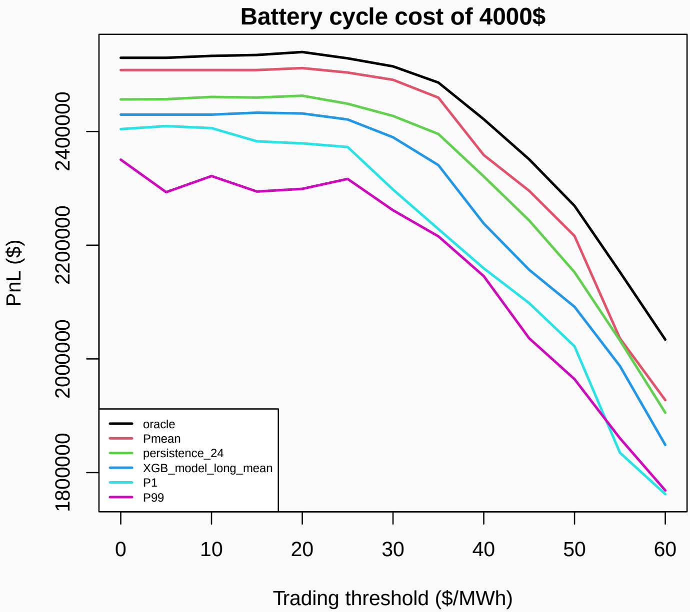
Hypothesis: bids are always accepted.

PnL for a battery on the day-ahead market

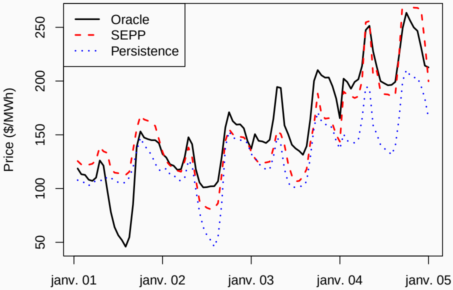
PnL is computed similarly to [Serafin and Weron, 2024]. For each day d:

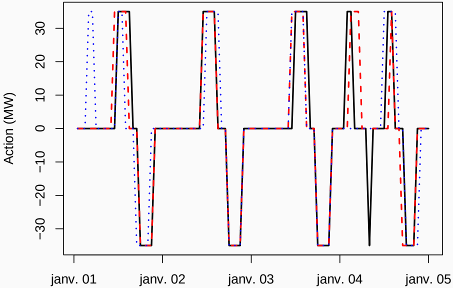
1. A matrix of spreads is computed as:
$$\Delta\hat{p}^d_{i,j} = \hat{p}_{d,j} - \hat{p}_{d,i}, \quad \text{for} \quad i < j$$
2. For a given trade threshold τ, if there exists a pair (i, j) such that:
$$\Delta\hat{p}^d_{i,j} > \tau$$
select the charge-discharge pair (i, j) that maximizes the spread:
$$(i, j) = \arg\max_{i<j} \Delta\hat{p}^d_{i,j}$$
3. Update the mileage and set $\Delta\hat{p}^d_{i,j} \leftarrow -\infty$.
4. Repeat the process while the mileage limit is not reached and there exists a pair (i, j) such that:
$$\Delta\hat{p}^d_{i,j} > \tau.$$
5. The PnL over the period is then simply $\sum p_{sell} \times V_{sell} - p_{buy} \times V_{buy}$.

PnL of the different models



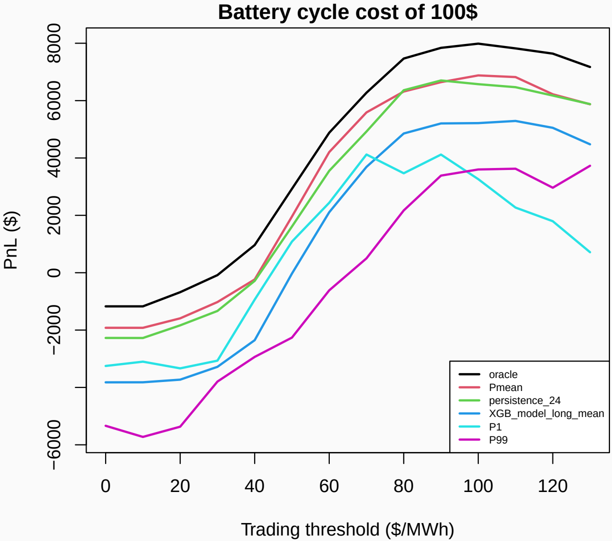
Example of actions (threshold = 0)





PnL for the Worms2403 battery

$C = 1\text{MWh}$, $P = 1\text{MW}$, 1 daily cycle, cycle cost of 100\$.



We can analyze the coefficients of model (1) to interpret the impact on the prediction.

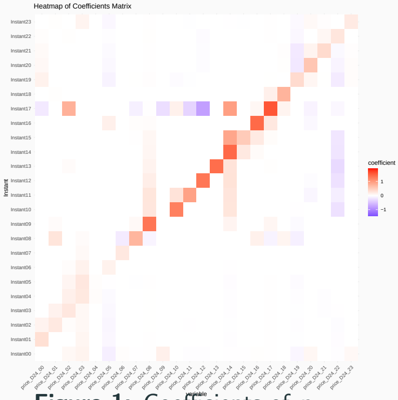


Figure 1: Coefficients of $p_{d-1,h}$.

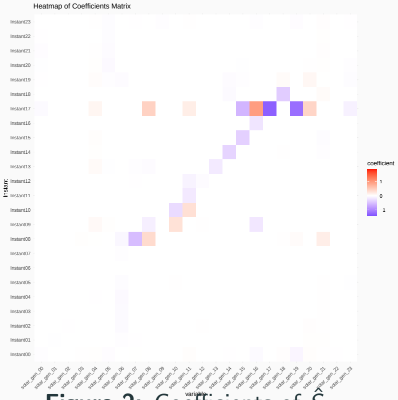


Figure 2: Coefficients of $\hat{S}_{d,h}$.

Conclusion & Future research

Conclusion

- The R library SEPP is used operationally by EDF Renewables North America as an input of their Market Optimization Engine to derive bid-offer curves.
- SEPP is capable of producing predictions for every node in California and Texas, but also for France.
- SEPP produces an ensemble of experts using standard statistics / ML models.
- Expert aggregation enhances the precision in terms of MAE, RMSE and PnL to a certain extent. However models with lower MAE/RMSE aren't necessarily better in PnL.

Future work

- Online optimization of hyperparameters.
- Direct optimization of models to maximize battery PnL (neural net optimized on genetic algorithms for instance ?).
- Inclusion of weather, load or generation scenarios for both point and probabilistic forecasts.
- Prediction spread matrix instead of simple price prediction [Serafin and Weron, 2024].
- Day-ahead / Real-time (DART) spread for market arbitrage.

Thank you for your attention !

Capezza, C., Palumbo, B., Goude, Y., Wood, S. N., and Fasiolo, M. (2021).
Additive stacking for disaggregate electricity demand forecasting.
The Annals of Applied Statistics, 15(2):727–746.

Enticott, E. and Fasiolo, M. (2025).
Probabilistic additive stacking for electricity load and price forecasting.
TBP.

Gaillard, P. and Goude, Y. (2016).
opera: Online prediction by expert aggregation.
URL: https://CRAN. R-project. org/package= opera. r package version, 1.

Serafin, T. and Weron, R. (2024).
Loss functions in regression models: Impact on profits and risk in day-ahead electricity trading.
Technical report, Department of Operations Research and Business Intelligence, Wroclaw ....

Uniejewski, B., Weron, R., and Ziel, F. (2017).
Variance stabilizing transformations for electricity spot price forecasting.
IEEE Transactions on Power Systems, 33(2):2219–2229.

Wood, S. N. (2017).
Generalized additive models: an introduction with R.
chapman and hall/CRC.

Probabilistic Additive Stacking

- Work with the University of Bristol [Capezza et al., 2021, Enticott and Fasiolo, 2025].
- Stacking meta-model is taken as:

$$p(y_i|x_i) = \sum_{k=1}^{K} \alpha_k(x_i) p_k(y_i|x_i) \tag{4}$$

- In (4) the most common form chosen for the $\alpha_k(x)$ is:

$$\alpha_k(x) = \frac{\exp\Big(\eta_k(x)\Big)}{\sum_{j=1}^{K} \exp\Big(\eta_j(x)\Big)}$$

  where the $\exp\Big(\eta_k(x)\Big)$ are based on a GAM [Wood, 2017].
- Hence the idea is to have models with heigher weight for certain market conditions (high gas prices, high / low penetration of renewables, etc...).
- The model is then learned by MLE.
- `R` library `Gamstacker` currently in development.