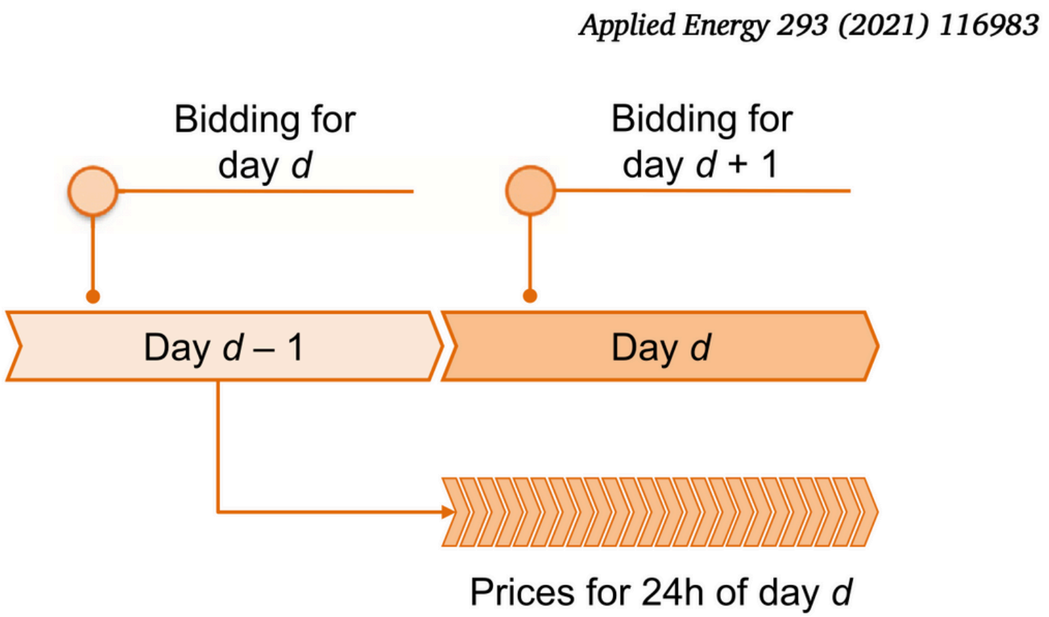
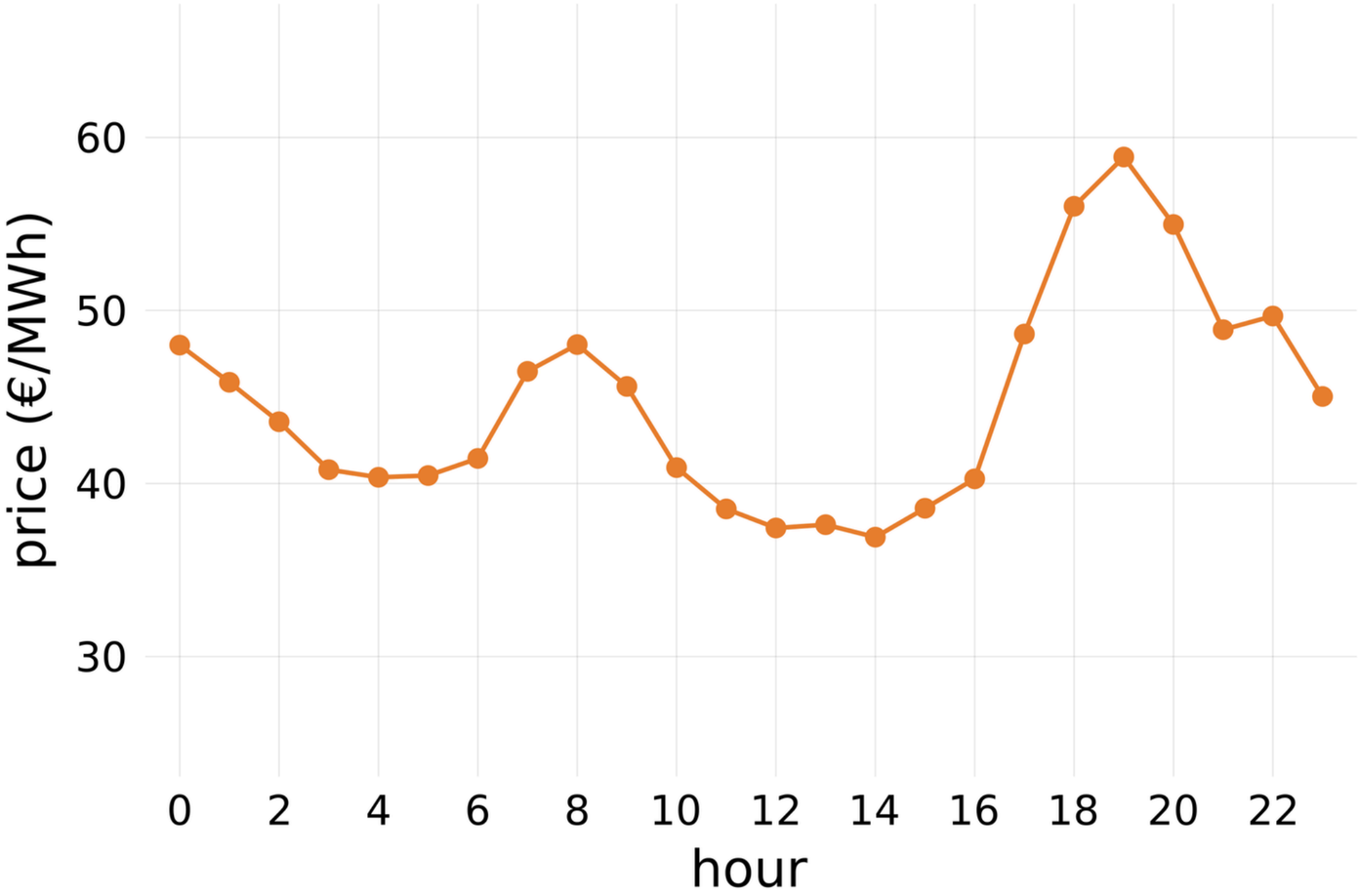
From stealing accuracy to stealing profits : Predicting Day-ahead Electricity Prices with Temporal Hierarchy Forecasting (THieF)

Arkadiusz Lipiecki, Kaja Bilińska, Nikolaos Kourentzes, Rafał Weron

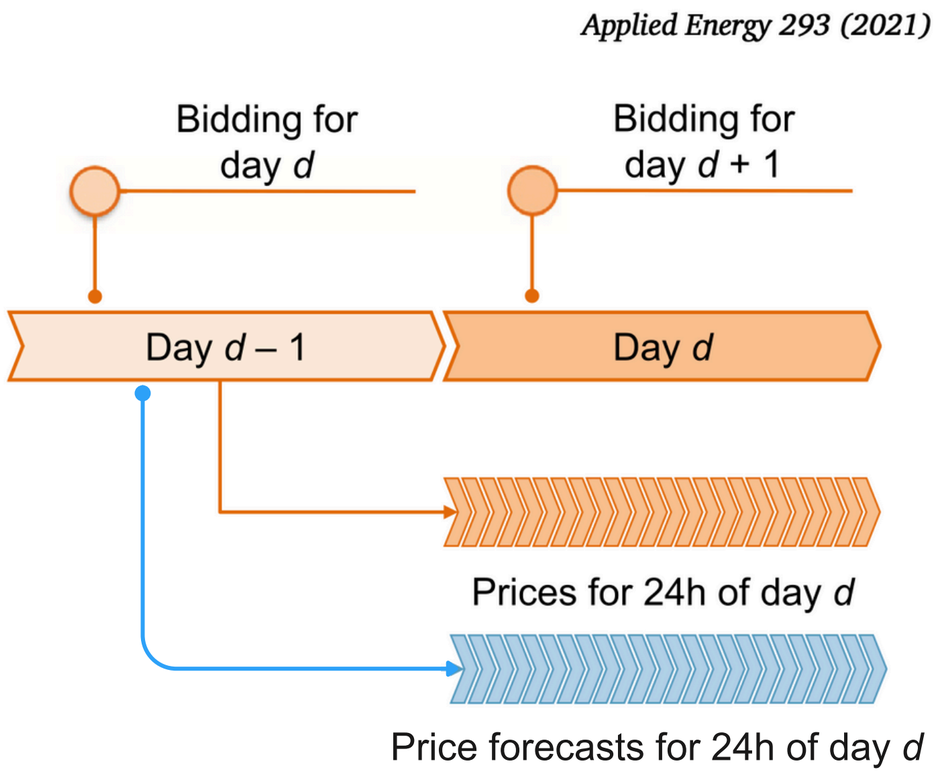
Wrocław University of Science and Technology

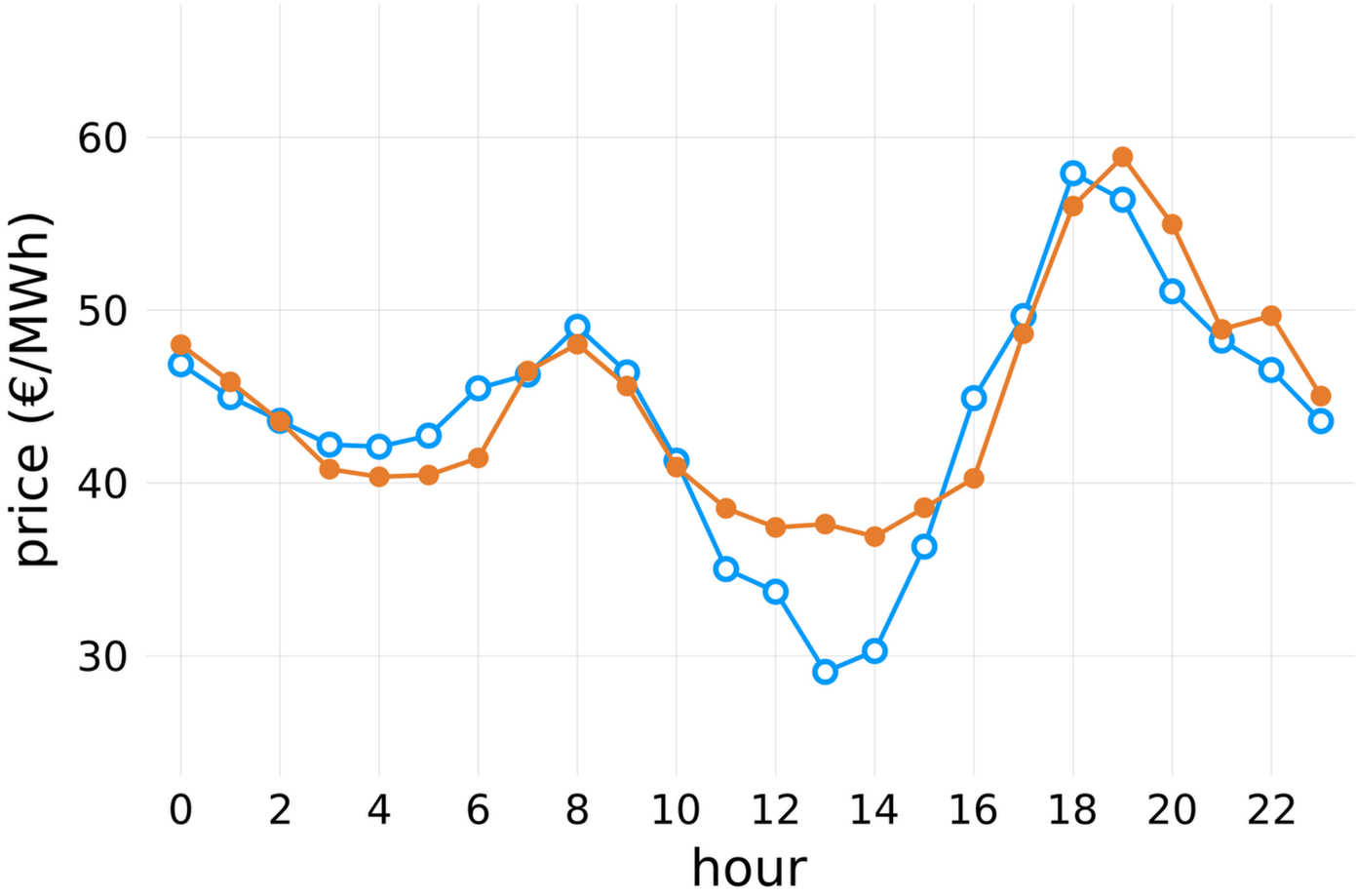
Day-ahead markets



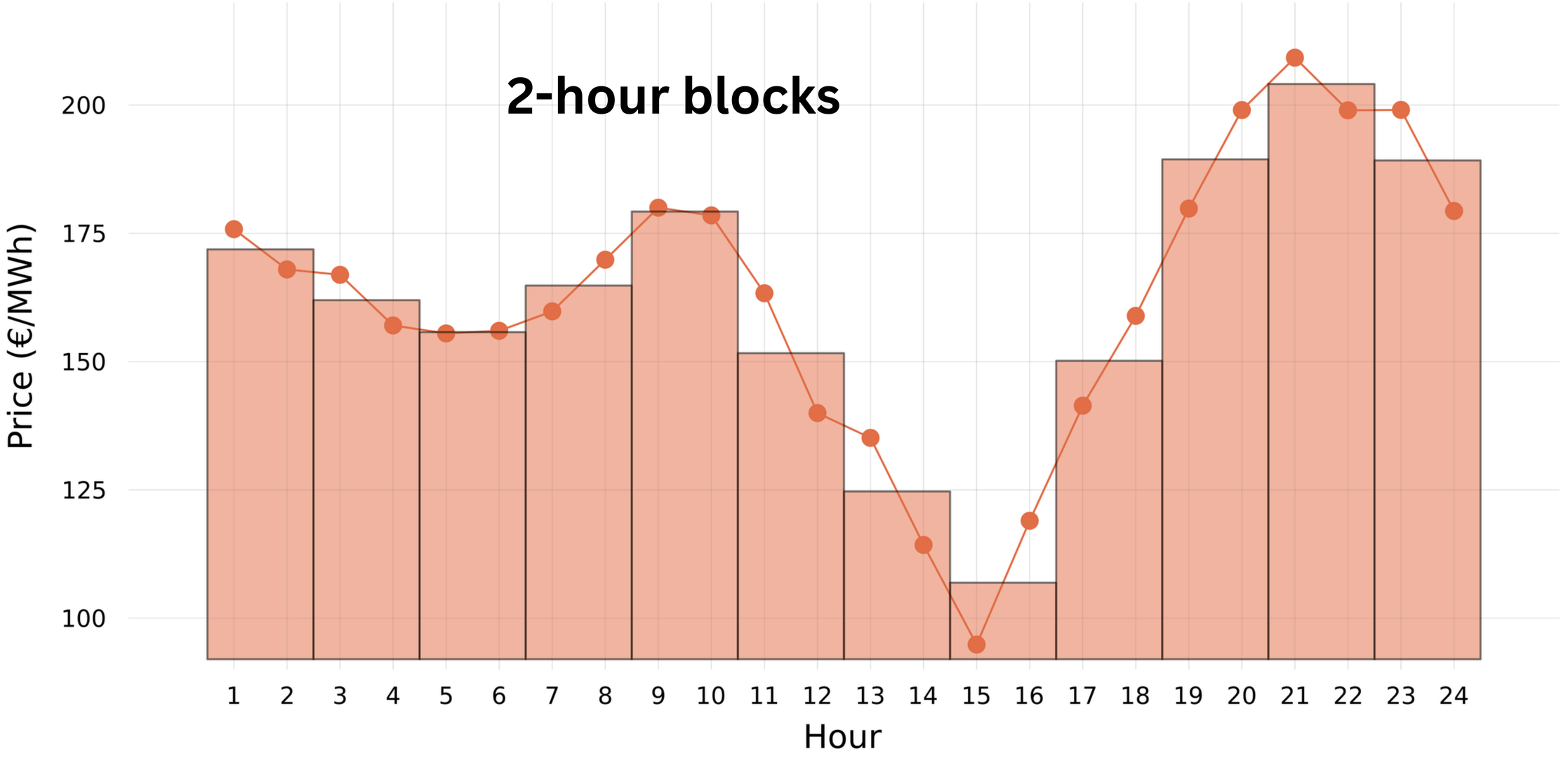


Hourly Forecasts

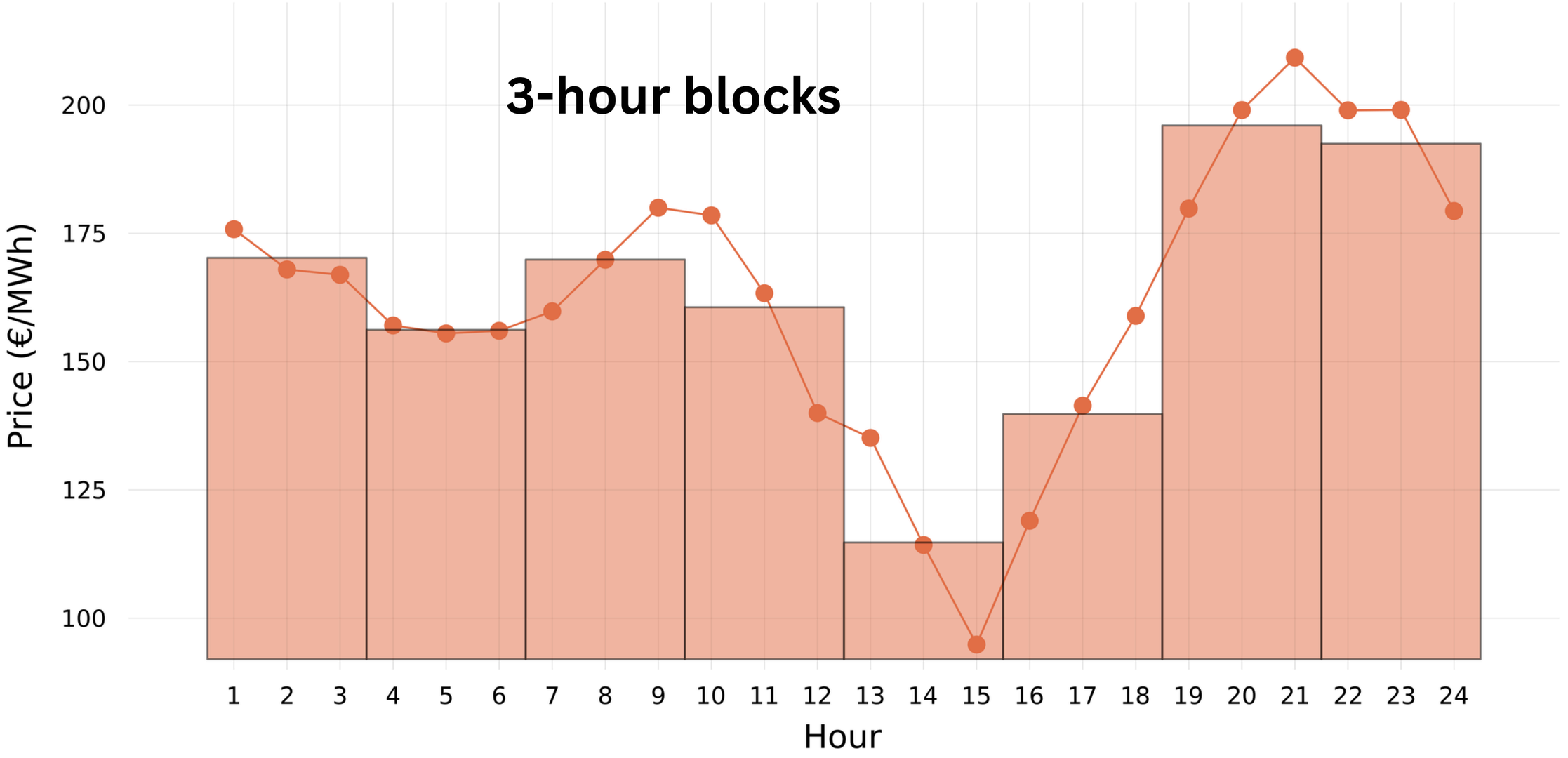




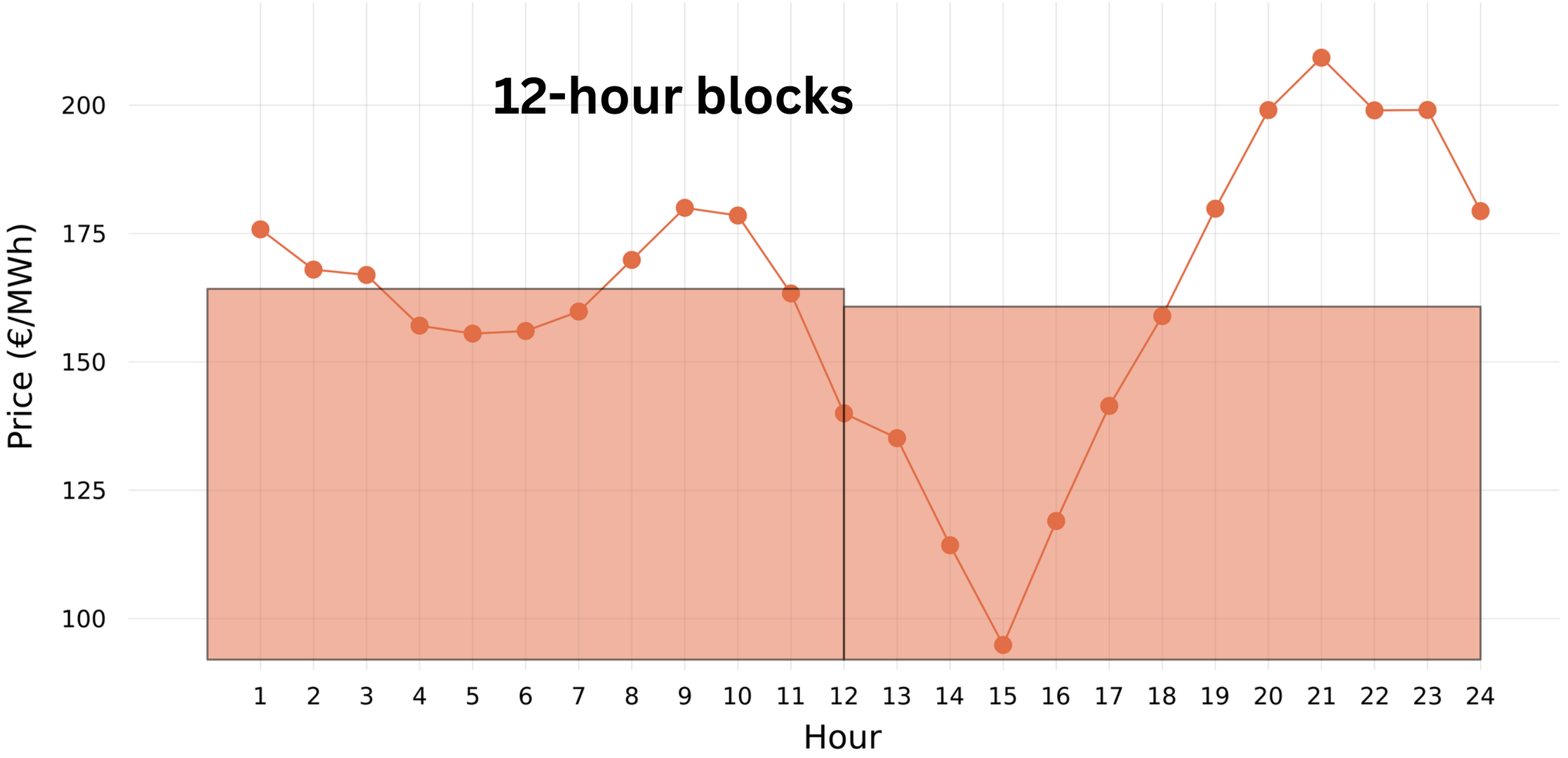
Blocks


2-hour blocks
Price (€/MWh)
200
175
150
125
100
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
Hour

Blocks


3-hour blocks
Price (€/MWh)
200
175
150
125
100
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
Hour

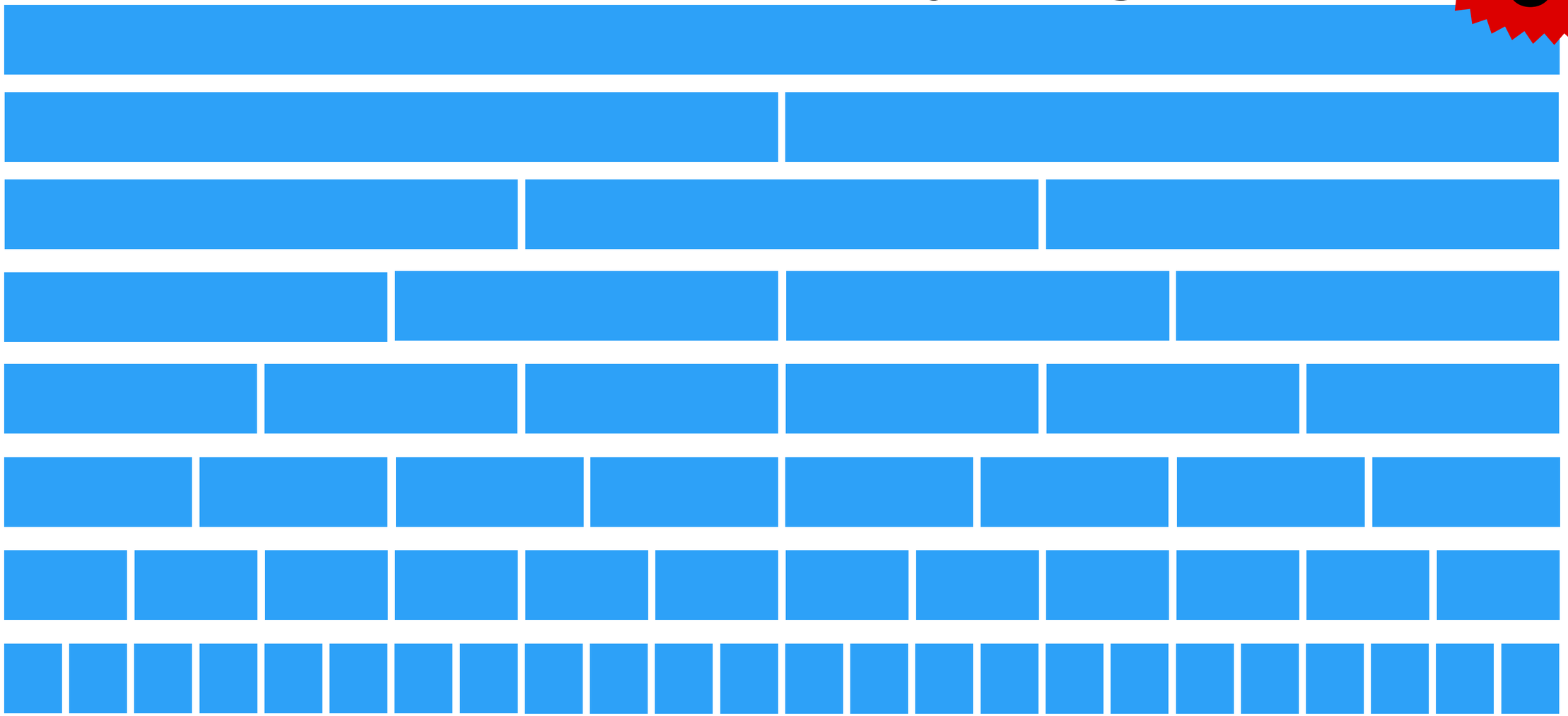
Blocks


12-hour blocks
Price (€/MWh)
200
175
150
125
100
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
Hour

Temporal hierarchies

one 24-hour block (daily average)

60

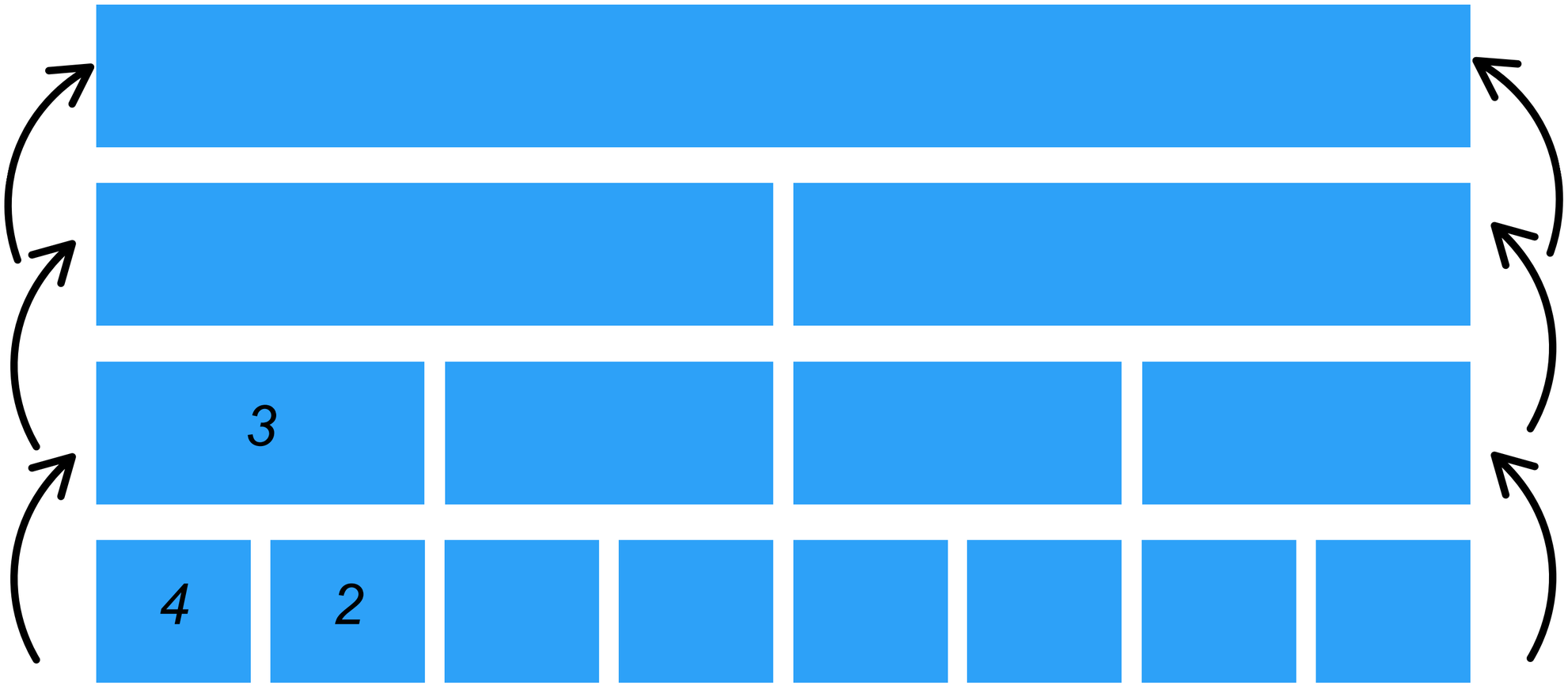


24 hourly blocks

Temporal hierarchies

Bottom up

Temporal hierarchies



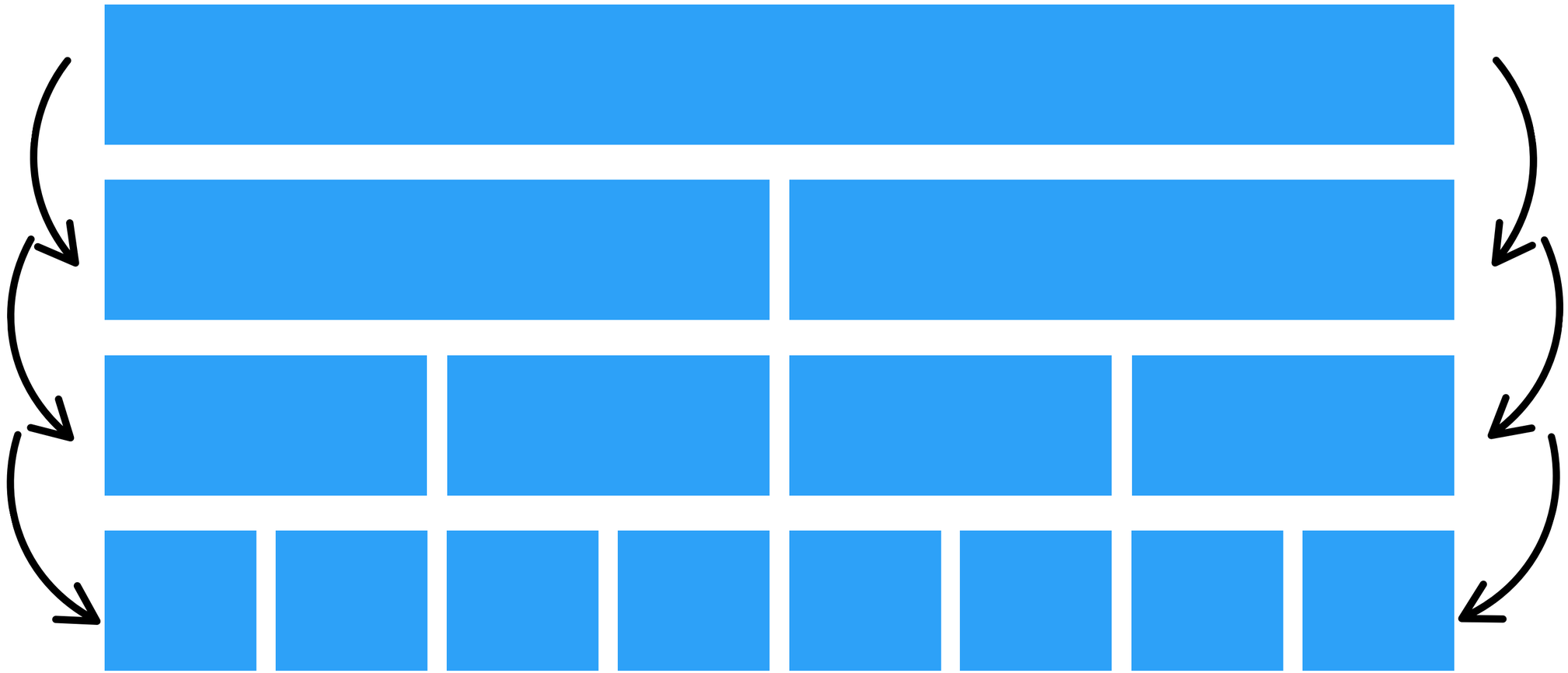
Bottom up

Temporal hierarchies

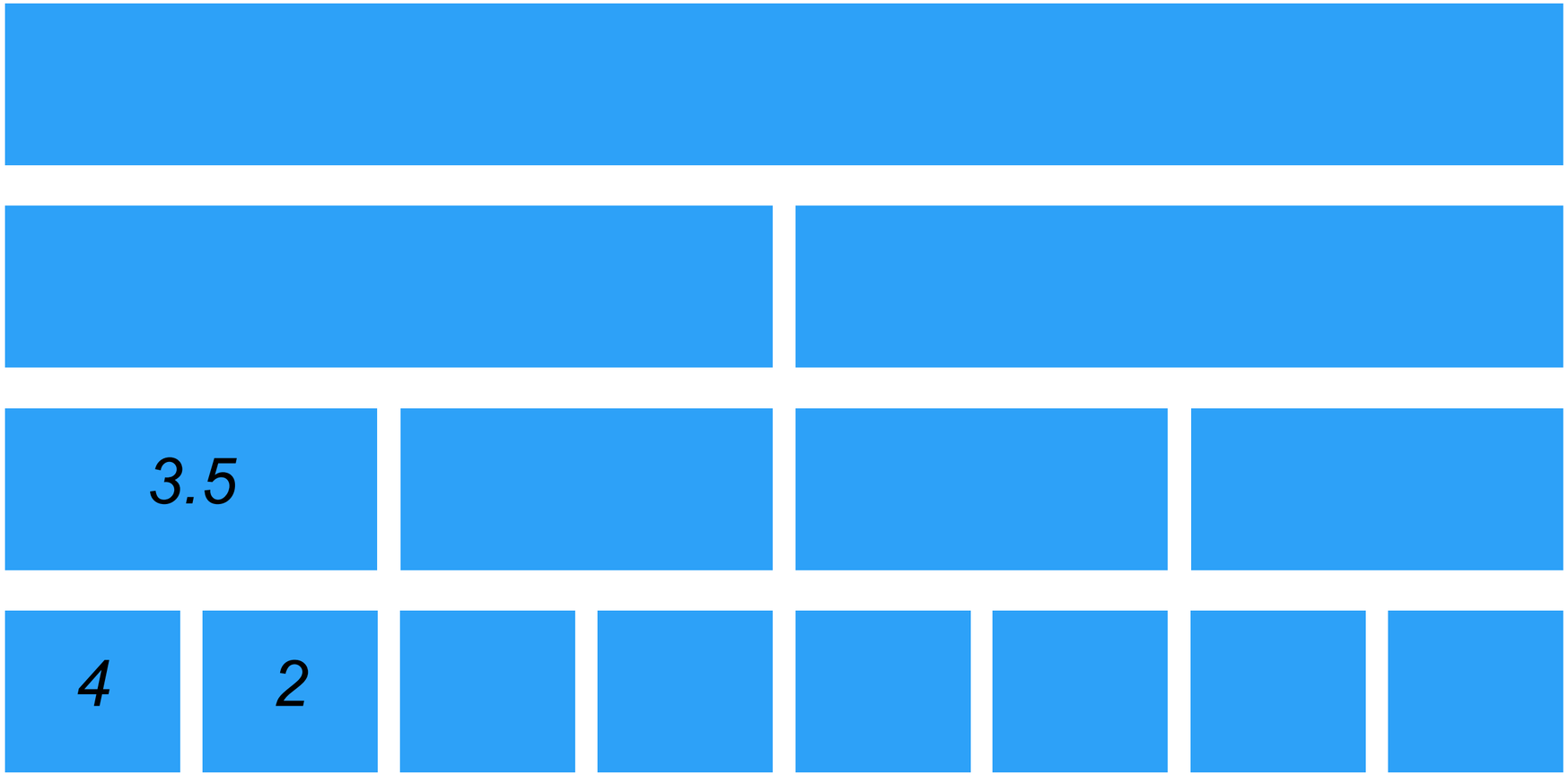
Top down

Temporal hierarchies

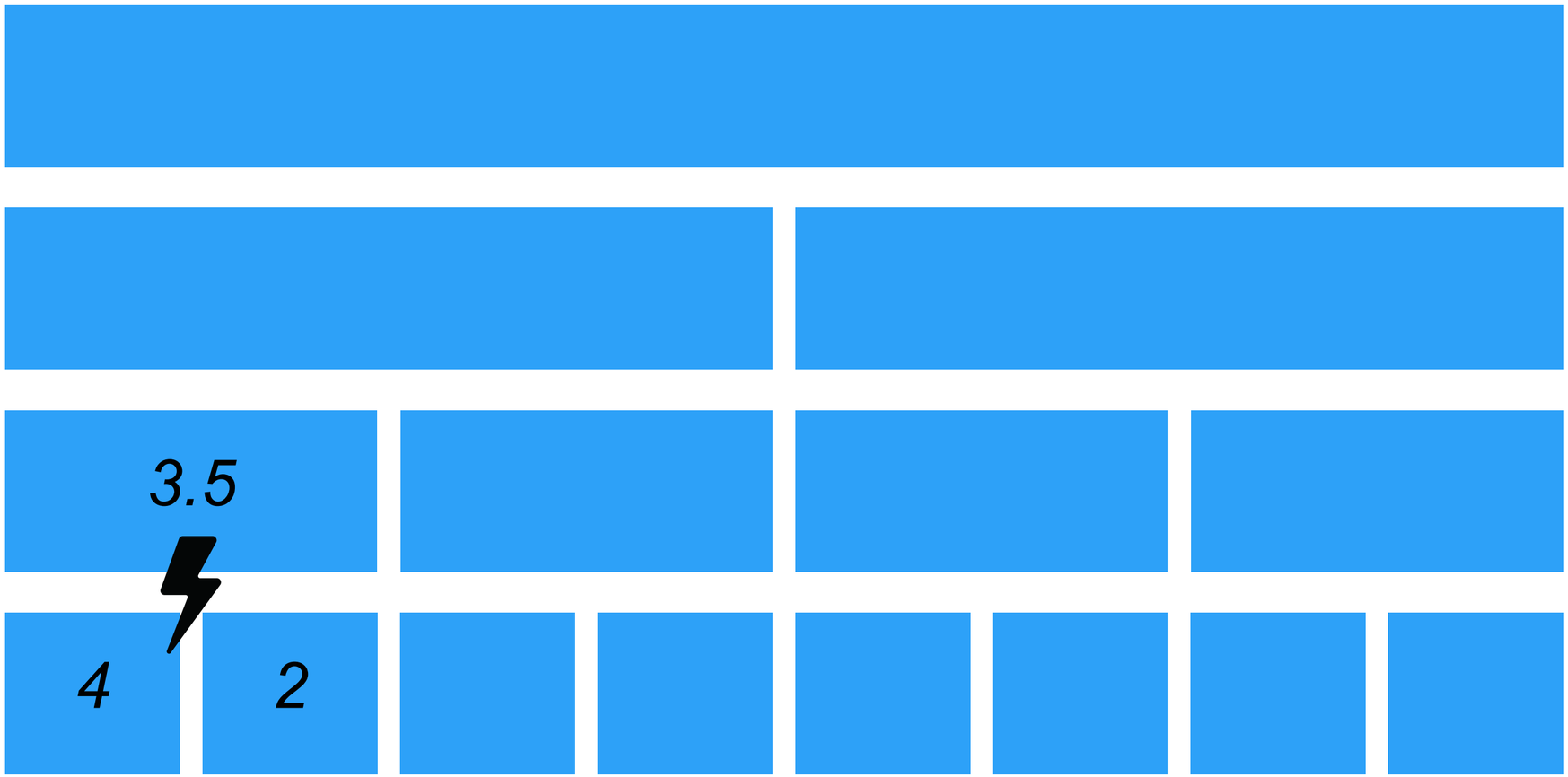
Top down



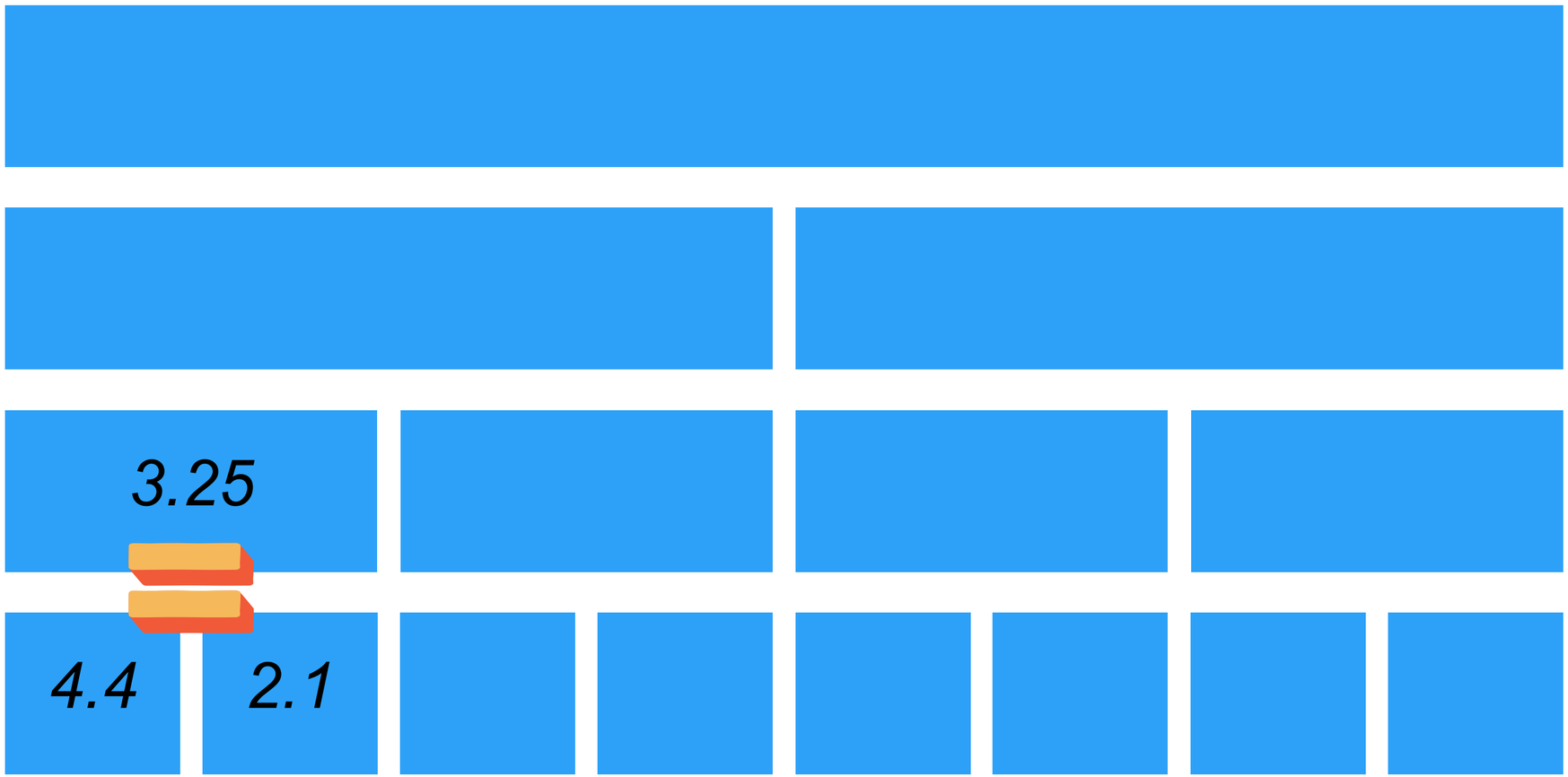
Forecast reconciliation



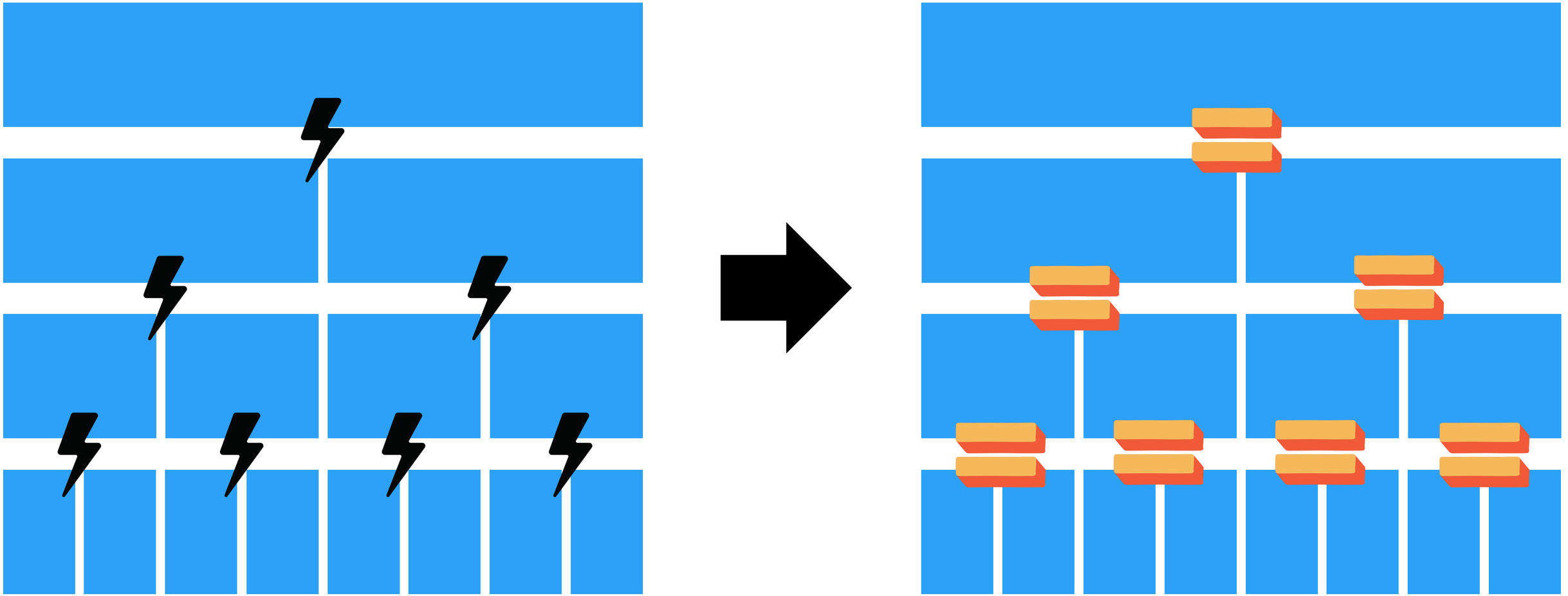
Forecast reconciliation



Forecast reconciliation

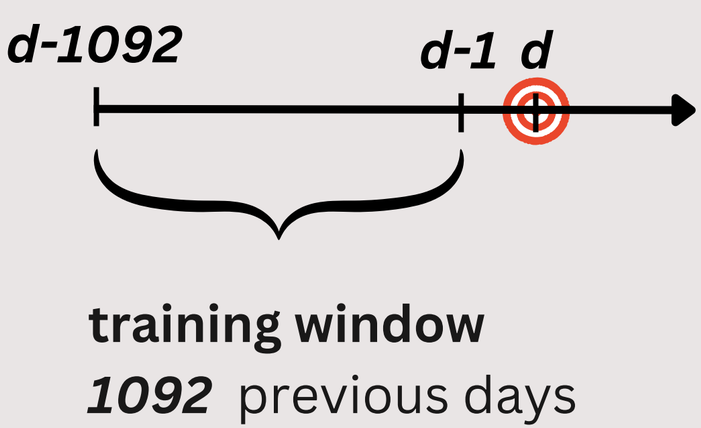


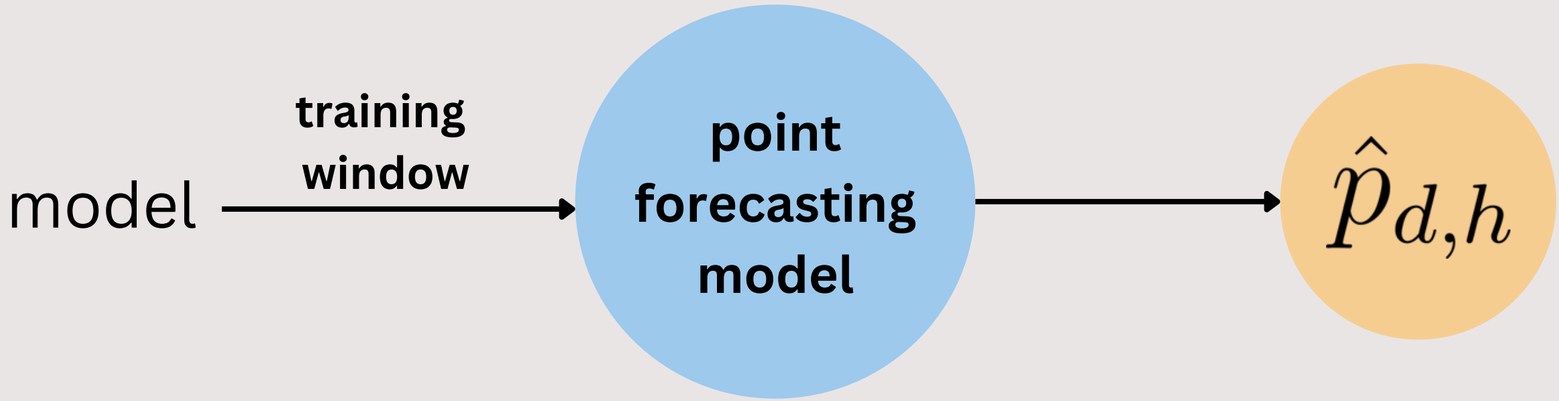
Forecast reconciliation



Step I: Compute base forecasts

Estimate base models for block ***h*** of day ***d***

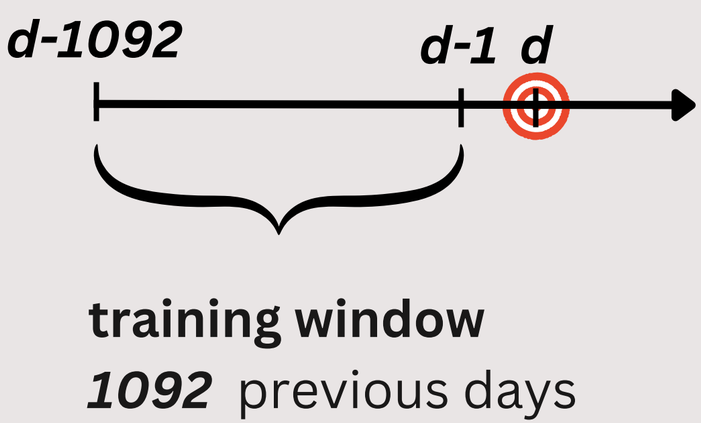


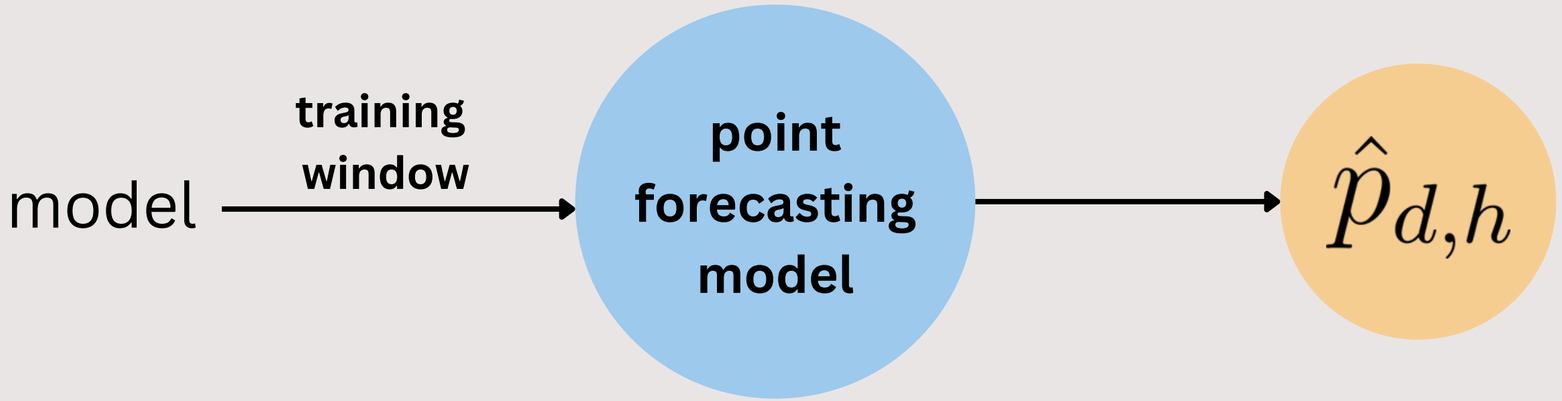


$$\hat{p}_{d,h} = f\left(p_{d-1,h}, \ldots, p_{d-7,h}, p_{d-1}^{min}, p_{d-1}^{max}, \hat{L}_{d,h}, \hat{W}_{d,h}, \mathrm{API}_{d-2}, \mathrm{TTF}_{d-2}, \mathrm{D}_d^{(1)}, \ldots, \mathrm{D}_d^{(7)}\right)$$

Step I: Compute base forecasts

Estimate base models for block ***h*** of day ***d***



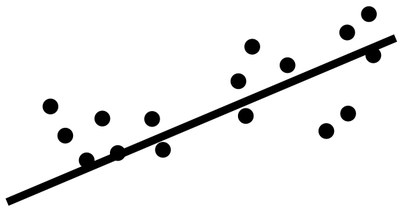


$$\hat{p}_{d,h} = f\left(p_{d-1,h}, \ldots, p_{d-7,h}, p^{min}_{d-1}, p^{max}_{d-1}, \hat{L}_{d,h}, \hat{W}_{d,h}, \mathrm{API}_{d-2}, \mathrm{TTF}_{d-2}, \mathrm{D}^{(1)}_{d}, \ldots, \mathrm{D}^{(7)}_{d}\right)$$

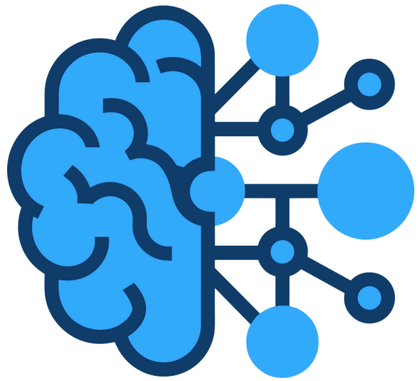
Base forecasts

$$\hat{p}_{d,h} = f\left(p_{d-1,h}, \ldots, \; p_{d-7,h}, \; p_{d-1}^{min}, \; p_{d-1}^{max}, \; \hat{L}_{d,h}, \; \hat{W}_{d,h}, \mathrm{API}_{d-2}, \; \mathrm{TTF}_{d-2}, \; \mathrm{D}_{d}^{(1)}, \ldots, \; \mathrm{D}_{d}^{(7)}\right)$$

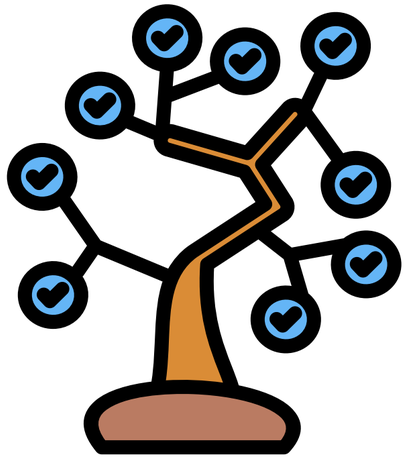
ARX



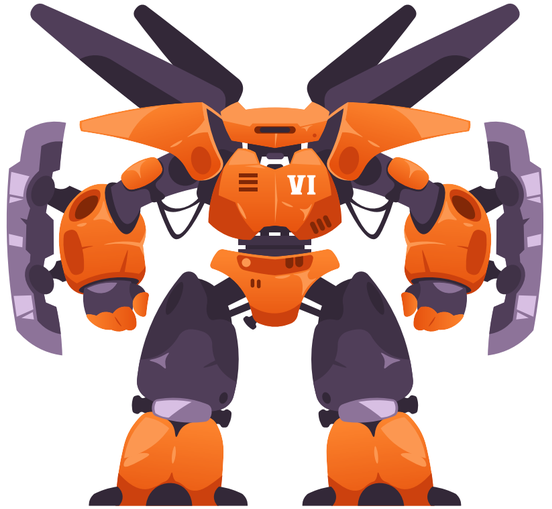
NARX



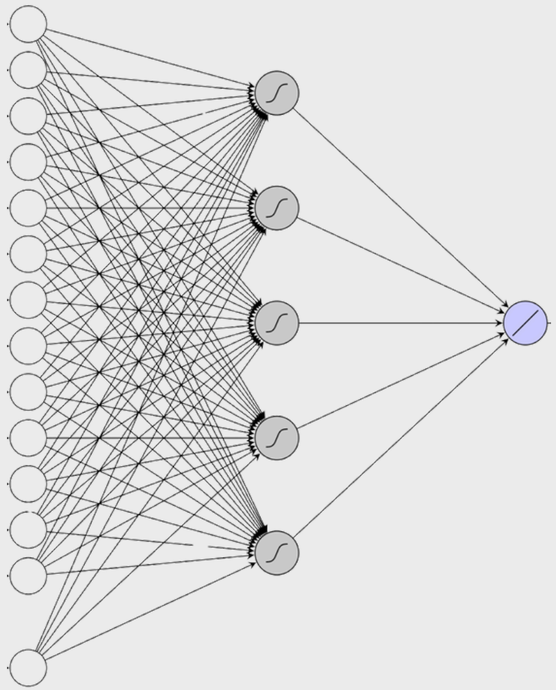
XGB



Mitra

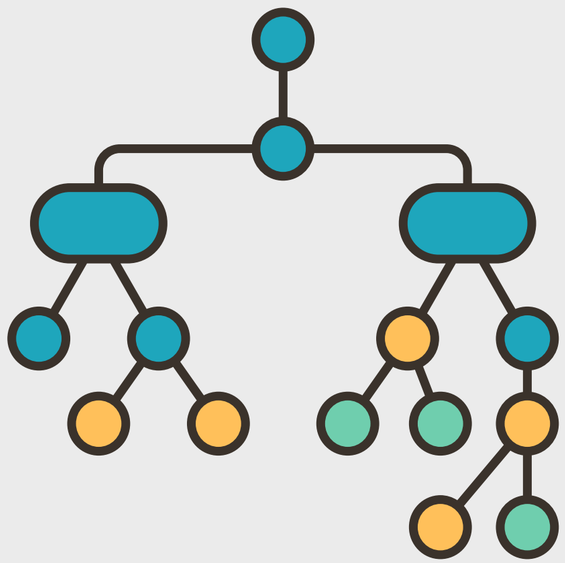


NARX



- 5 neurons in the hidden layer
- Hyperbolic tangent activation functions
- Early stopping with 10% validation set
- Weights estimated with Levenberg-Marquardt in Matlab
- Final forecast is an average of 10 independently trained networks

XGB

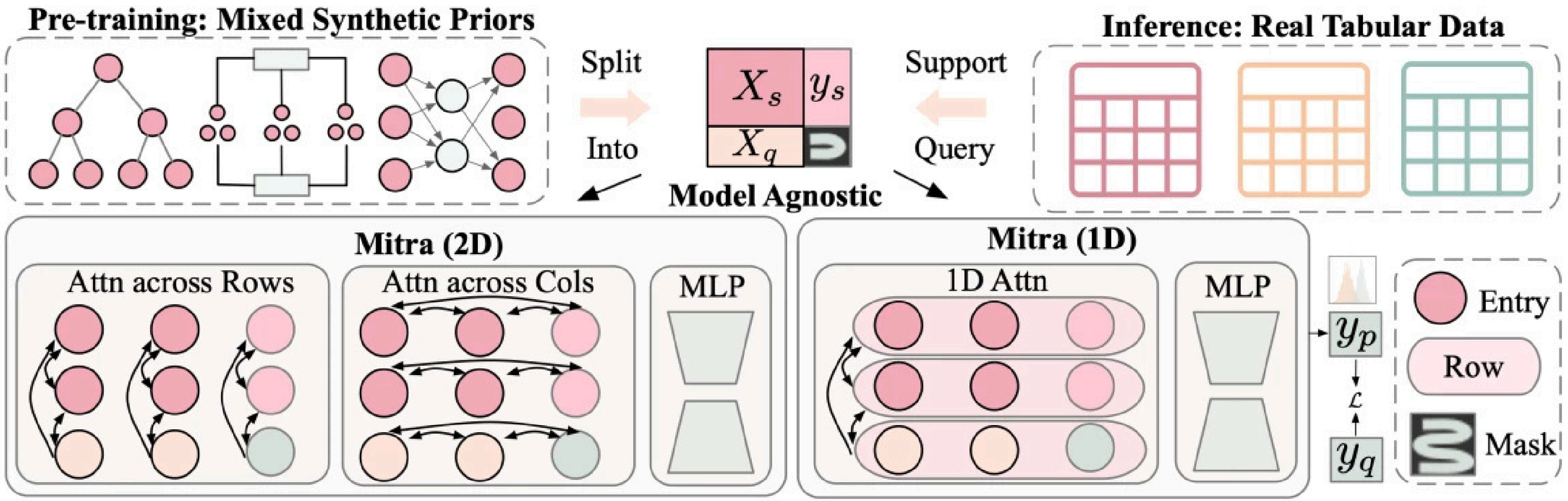


- Python **XGBoost** package
- Mean squared error loss function
- Optuna-based hyperparameter optimization
- 10 independent runs once a year, separately for each hour/block
- Final forecast is an average of 10 XGB decision trees with different hyperparameters

Mitra

AutoGluon

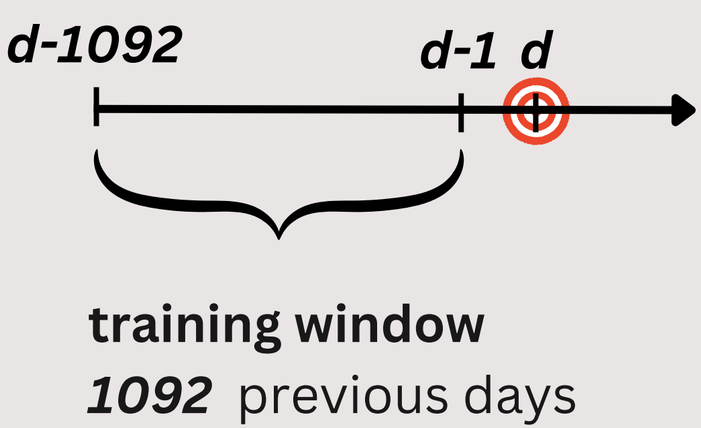
- 12-layer Transformer of 72 M parameters
- Pretrained on synthetic data
- Zero-shot forecasting with in-context learning



amazon.science/blog/mitra-mixed-synthetic-priors-for-enhancing-tabular-foundation-models

Step I: Compute base forecasts

Estimate base models for block ***h*** of day ***d***



training window
1092 previous days

model → **training window** → **point forecasting model** → $\hat{p}_{d,h}$

x60

$$\bar{\mathbf{P}}_{\mathbf{d}} = [\hat{p}_{d,1}, \cdots, \hat{p}_{d,60}]^T$$

Unreconciled forecasts

Minimum trace (MinT) reconciliation

Reconciled forecasts

$$\tilde{\mathbf{P}}_{\mathbf{d}} = W\bar{\mathbf{P}}_{\mathbf{d}}$$

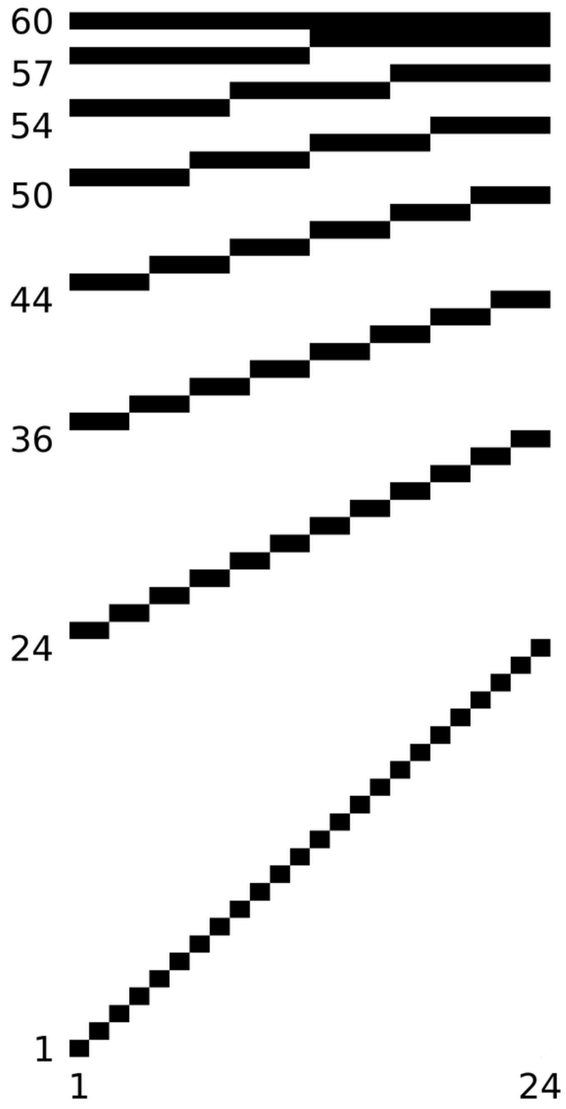
Base forecasts

Error covariance matrix

$$\tilde{\mathbf{P}}_{\mathbf{d}} = \mathbf{S}(\mathbf{S}^T\boldsymbol{\Lambda}^{-1}\mathbf{S})^{-1}\mathbf{S}^T\boldsymbol{\Lambda}^{-1}\bar{\mathbf{P}}_{\mathbf{d}}$$

Summing matrix

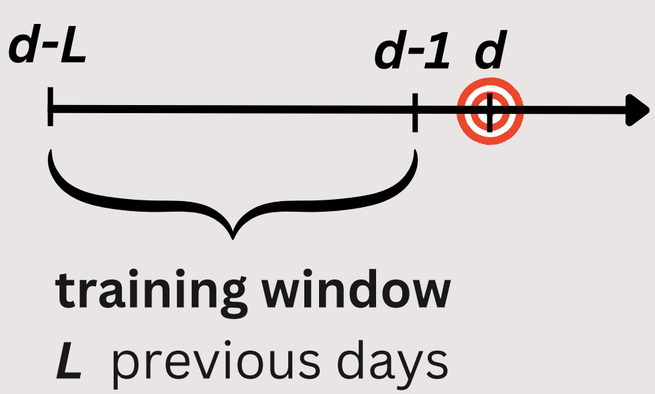
Wickramasuriya et al. (2017)



Summing matrix

Step II: Compute errors

Having estimated base models for block ***h*** of day ***d***



model —training window→ **point forecasting model** → $\hat{p}_{d,h}$

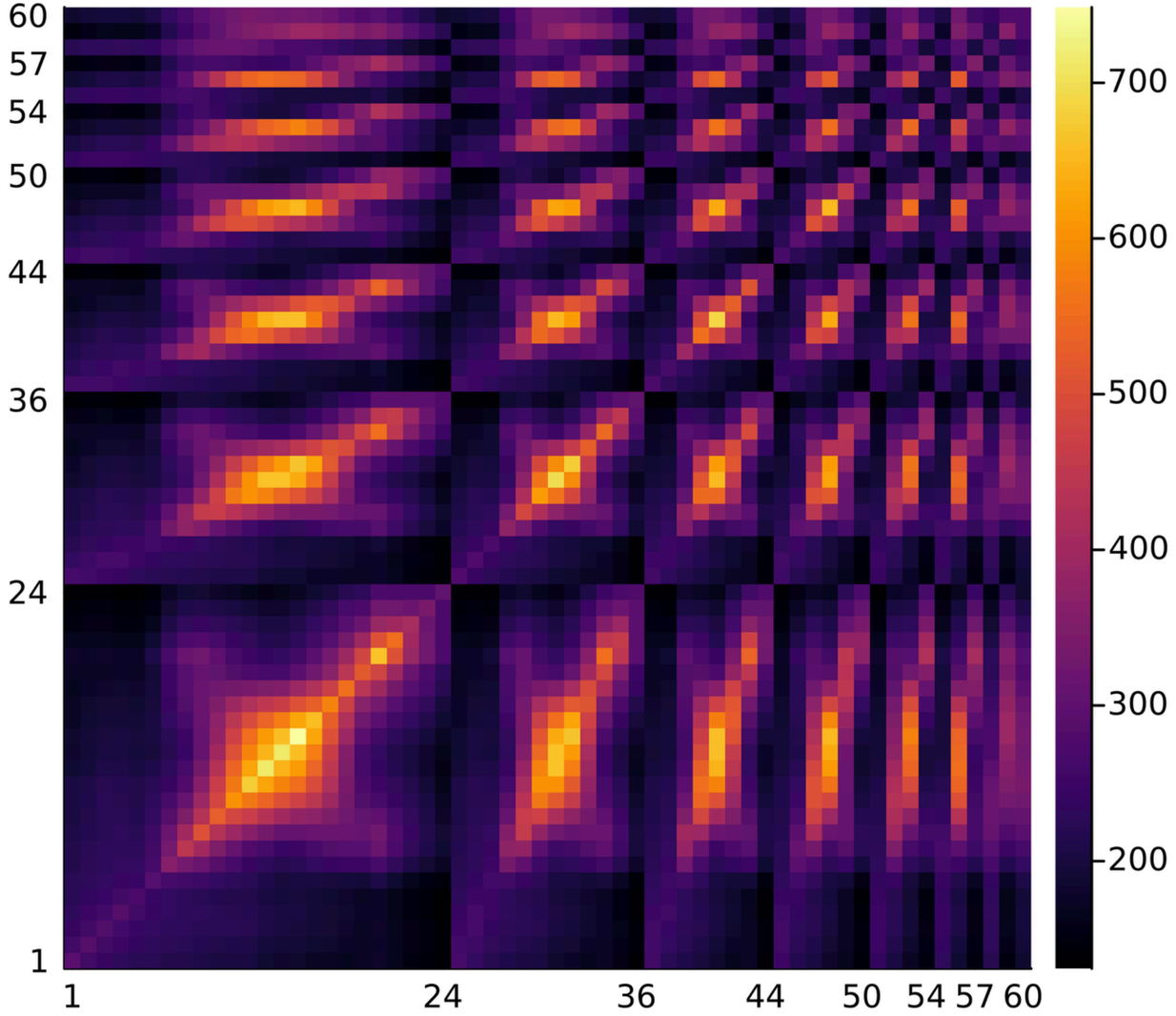
$$\mathbf{E_d} = \mathbf{P_d} - \bar{\mathbf{P}}_\mathbf{d}$$

shrinkage → Ledoit & Wolf (2004)

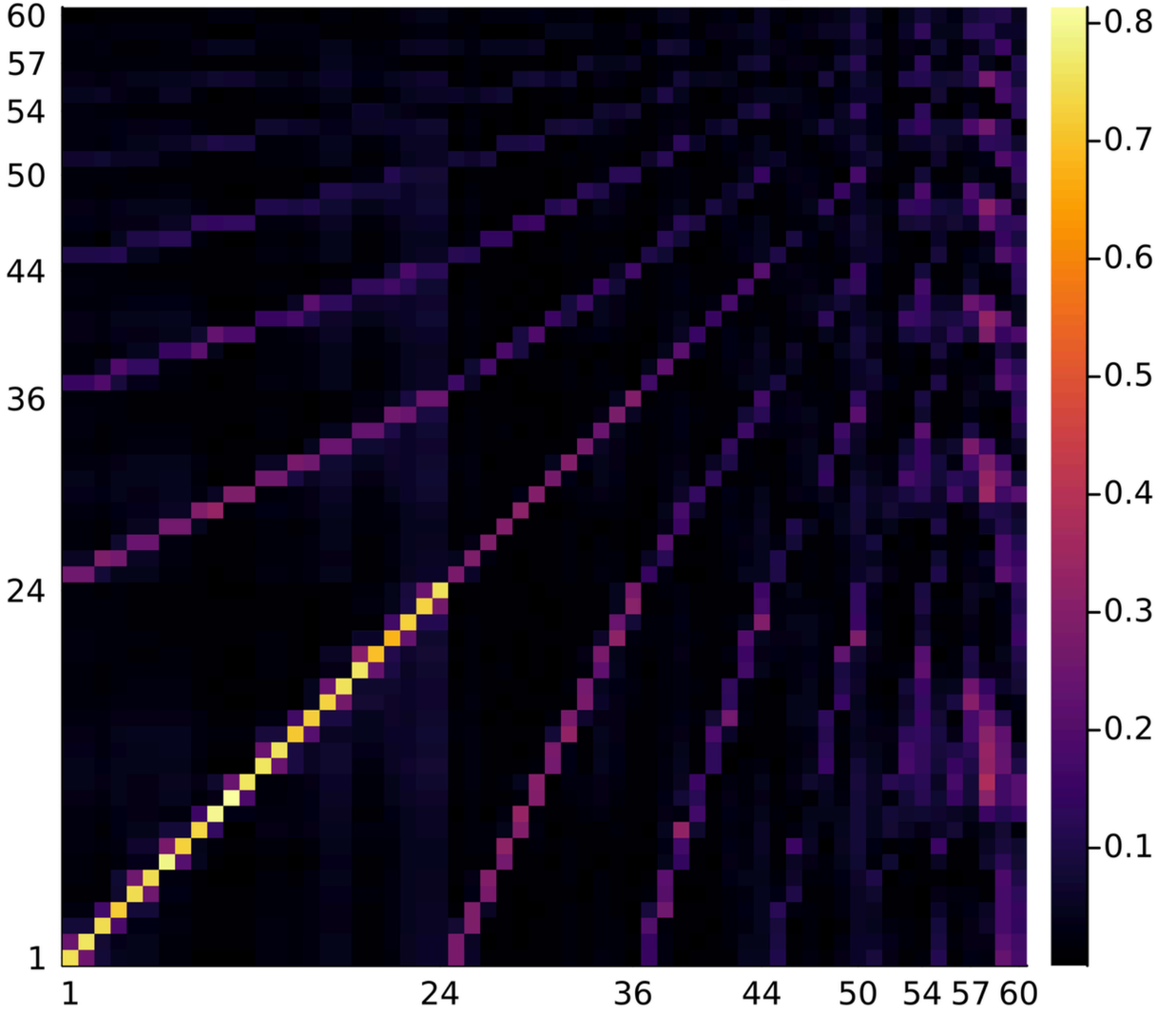
Forecast error covariance matrix

Step III: Compute weights through MinT

Error covariance

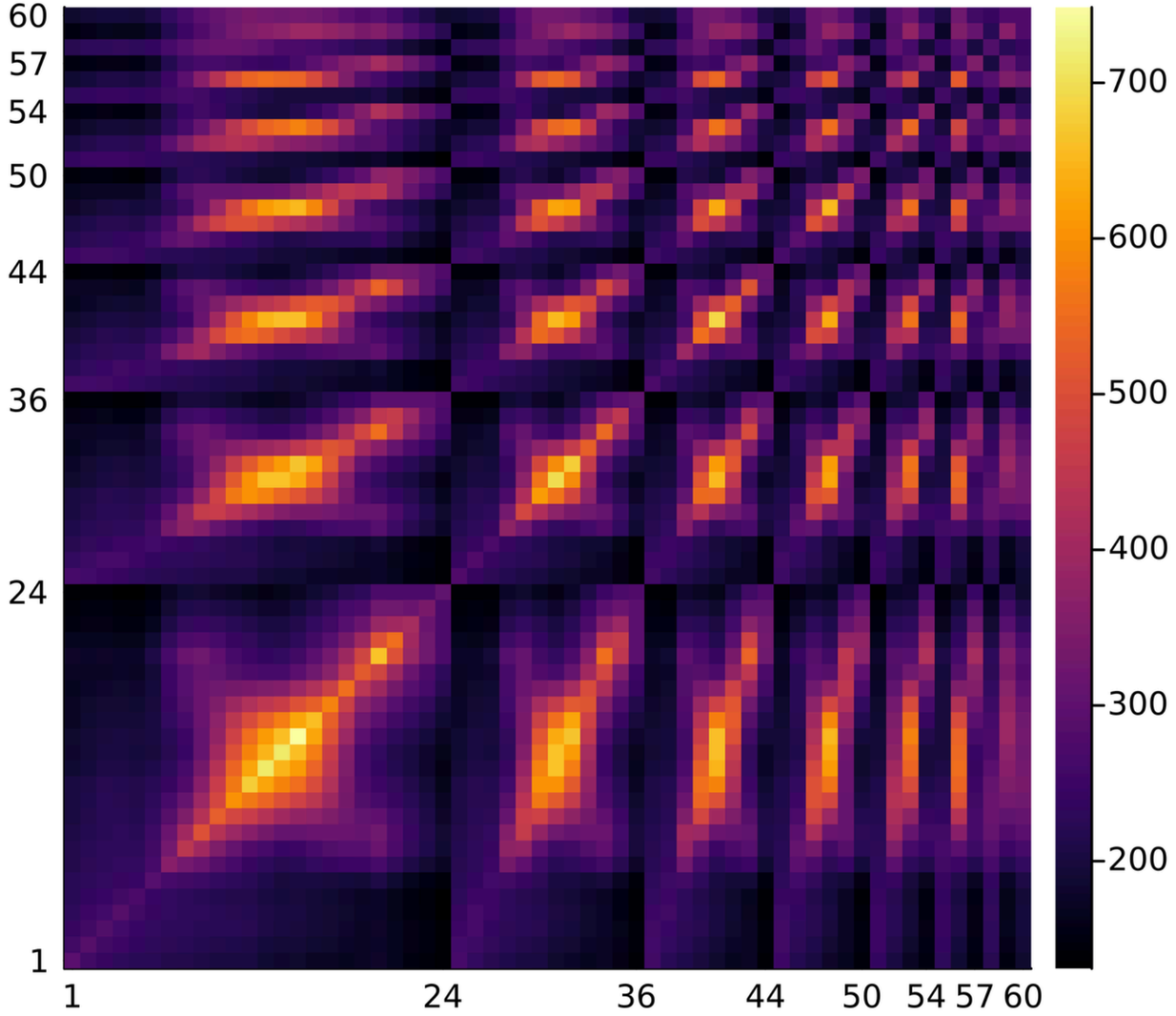


Estimated weights

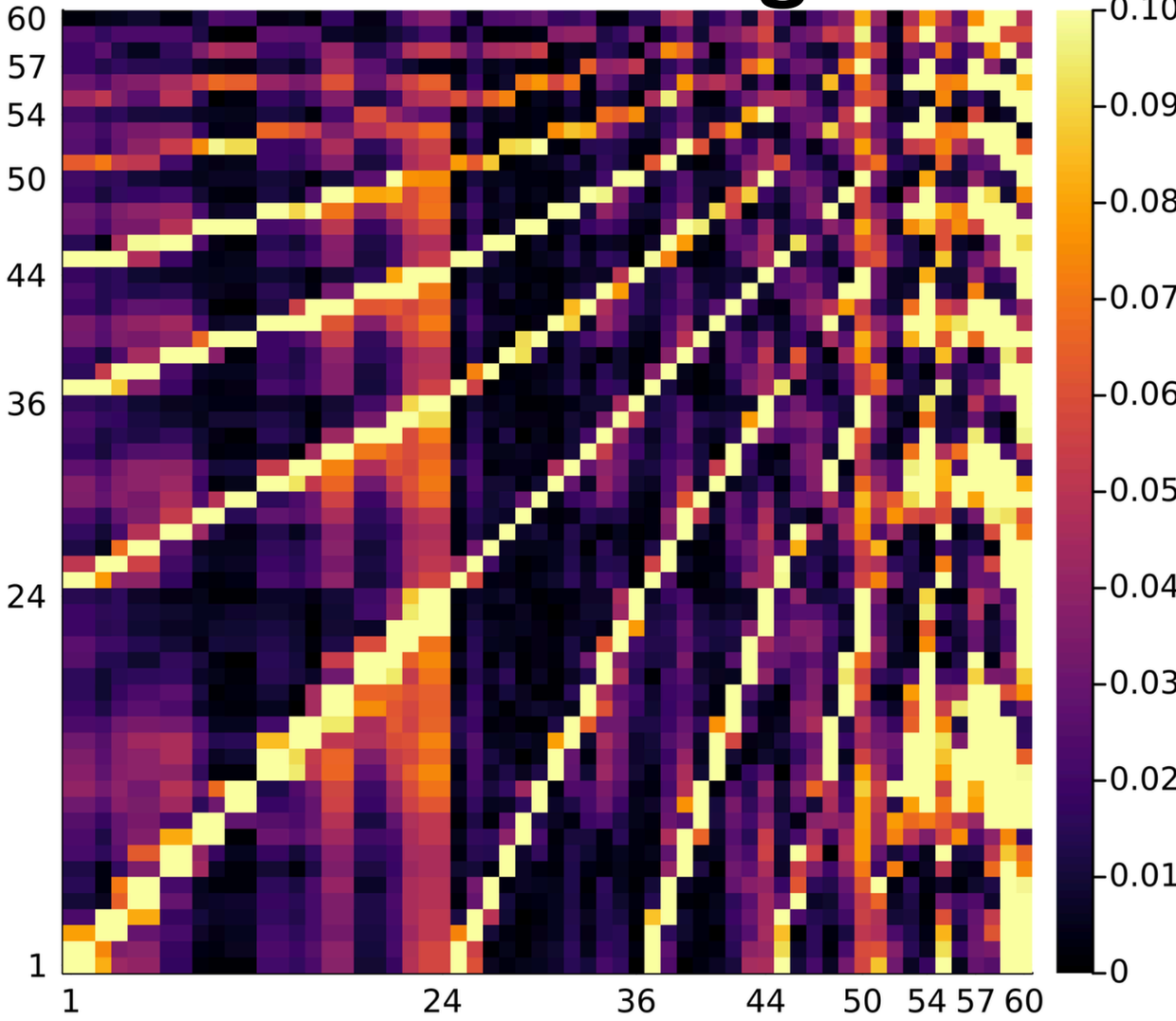


Step III: Compute weights through MinT

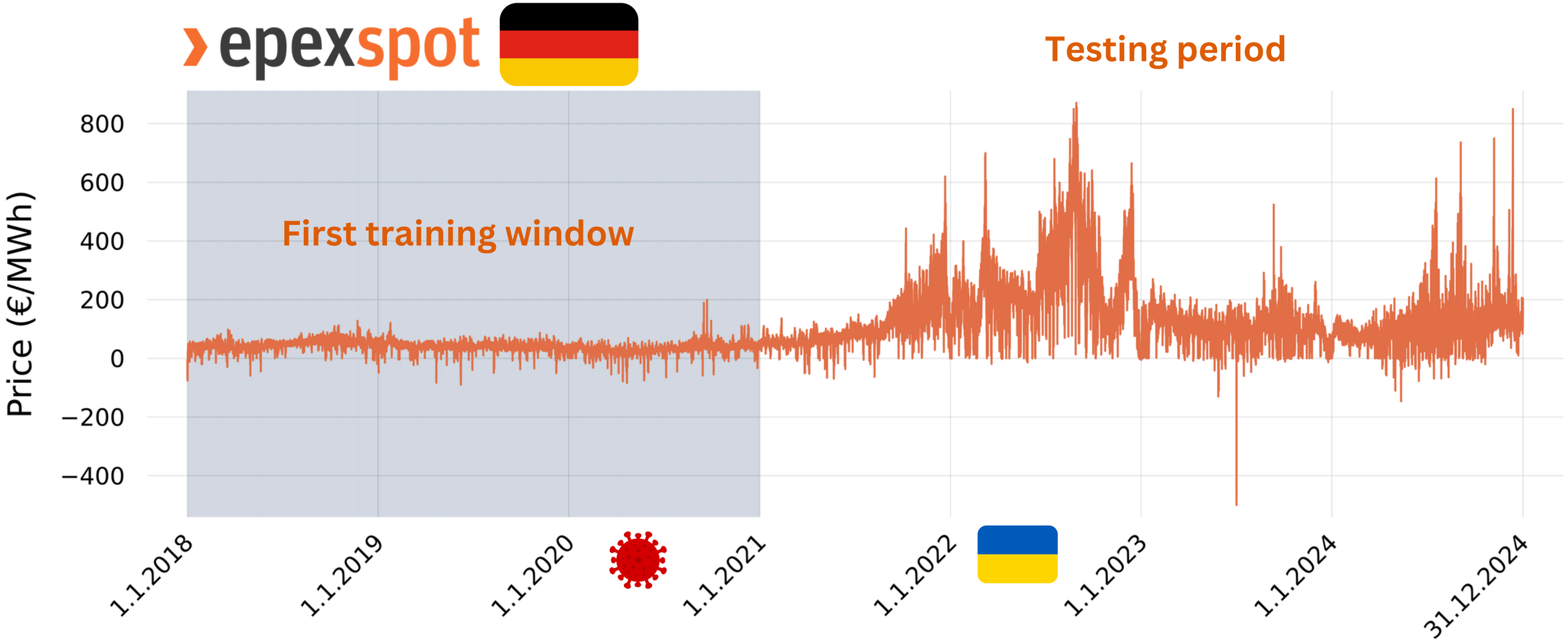
Error covariance



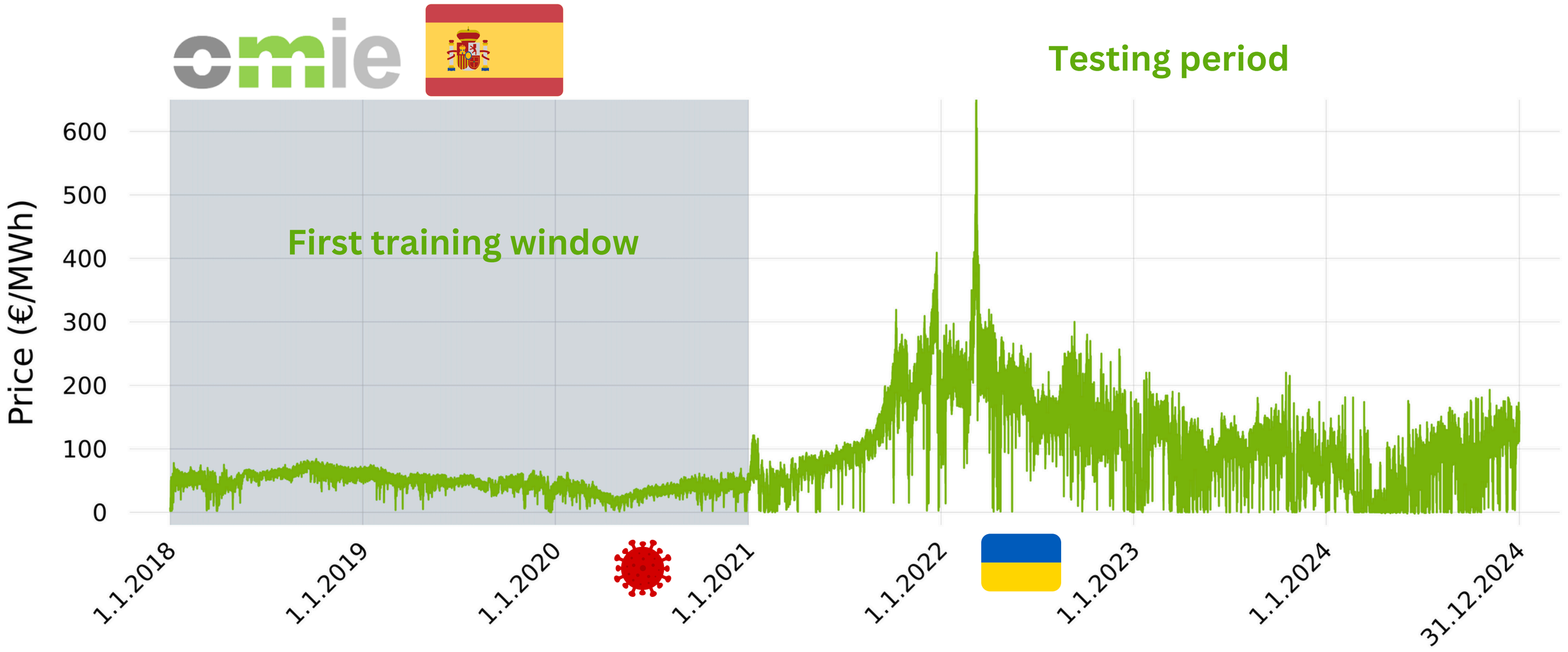
Estimated weights



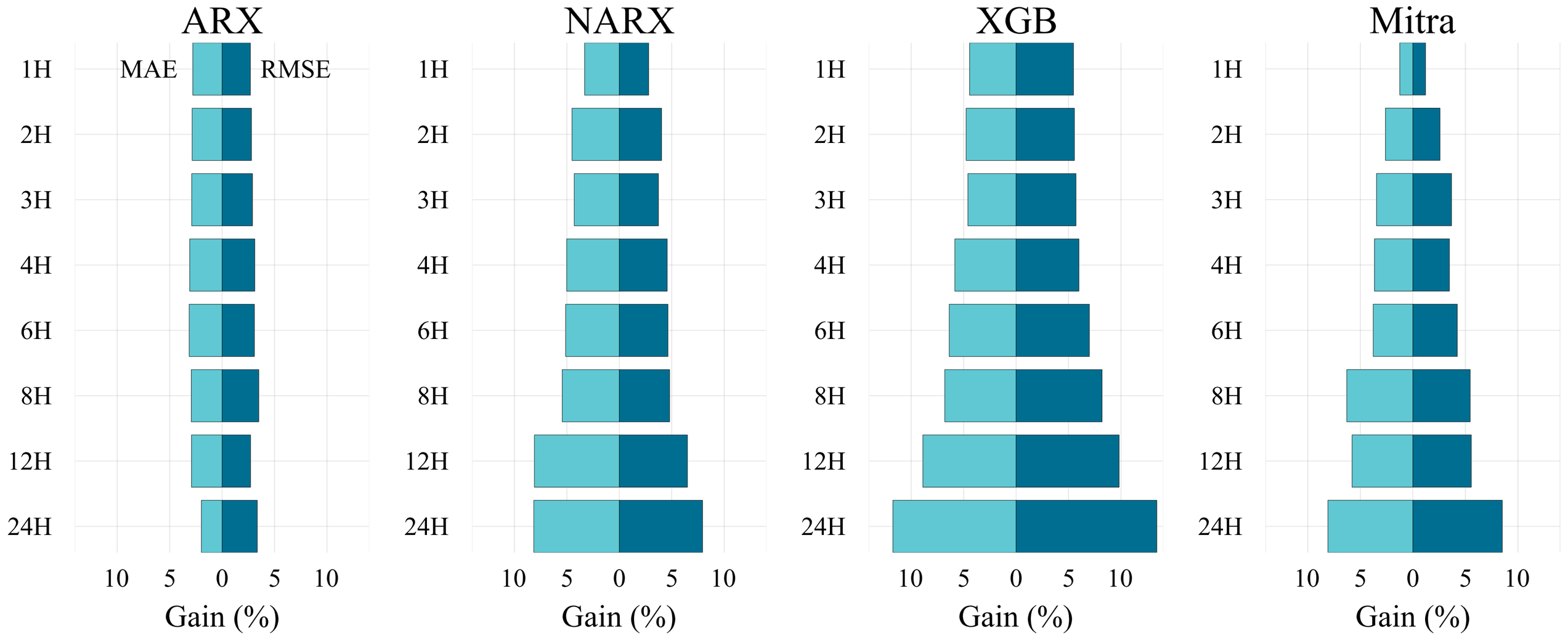
Data


epexspot
Testing period
First training window
Price (€/MWh)
800
600
400
200
0
−200
−400
1.1.2018
1.1.2019
1.1.2020
1.1.2021
1.1.2022
1.1.2023
1.1.2024
31.12.2024

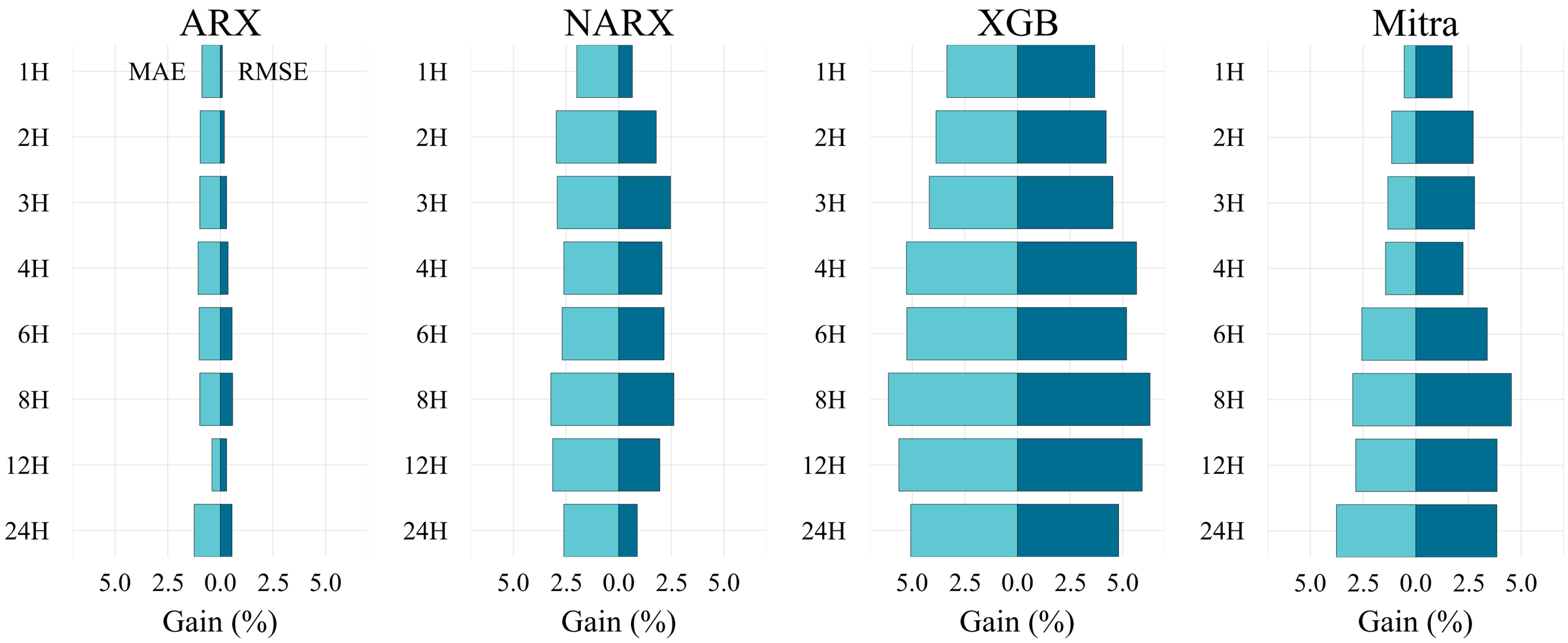
Data


omie
Testing period
First training window
Price (€/MWh)
600
500
400
300
200
100
0
1.1.2018
1.1.2019
1.1.2020
1.1.2021
1.1.2022
1.1.2023
1.1.2024
31.12.2024

Results


ARX
NARX
XGB
Mitra
MAE
RMSE
1H
2H
3H
4H
6H
8H
12H
24H
10
5
0
5
10
Gain (%)

Results


ARX
NARX
XGB
Mitra
MAE
RMSE
1H
2H
3H
4H
6H
8H
12H
24H
5.0
2.5
0.0
2.5
5.0
Gain (%)

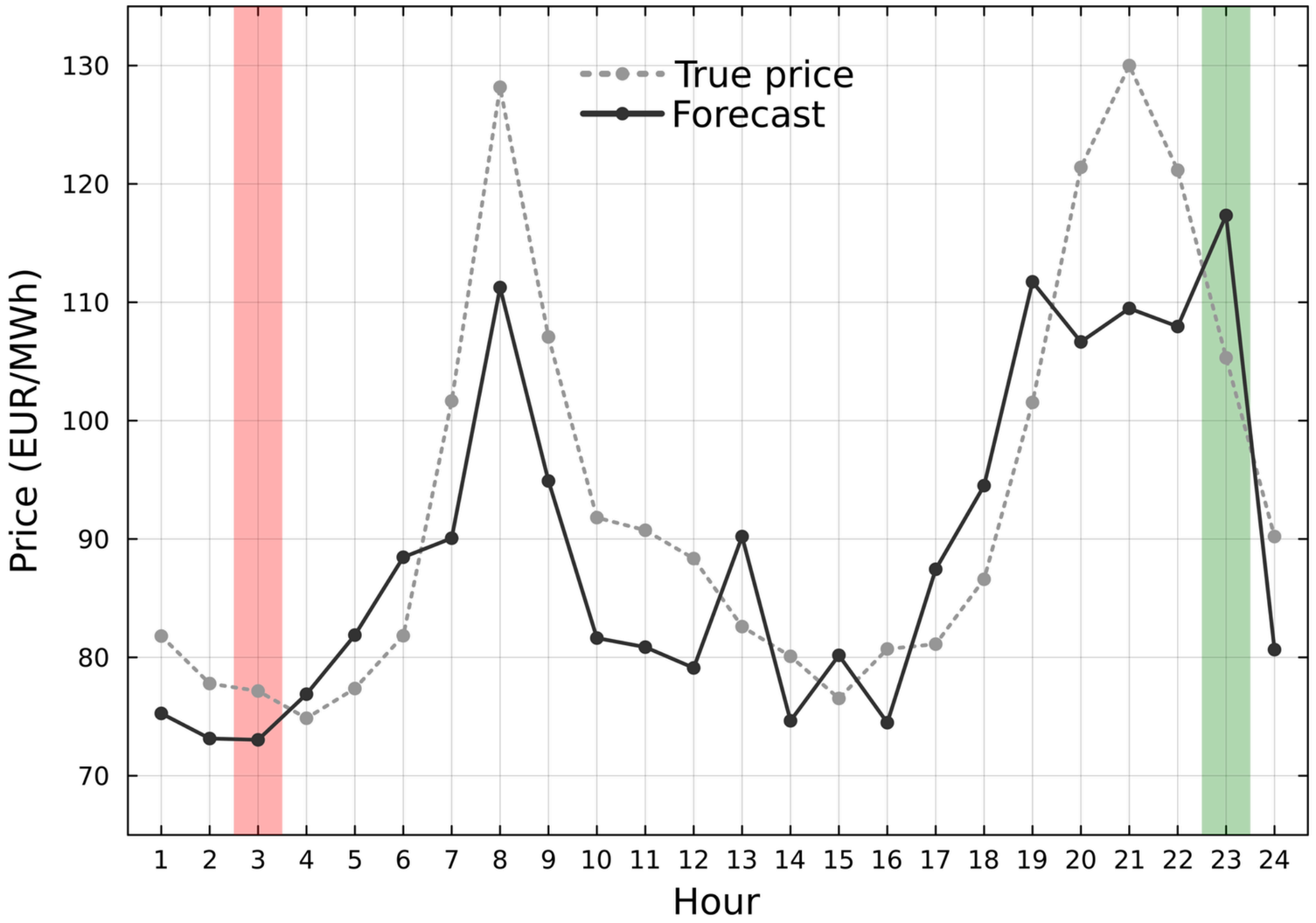
Quantitative Finance > Statistical Finance

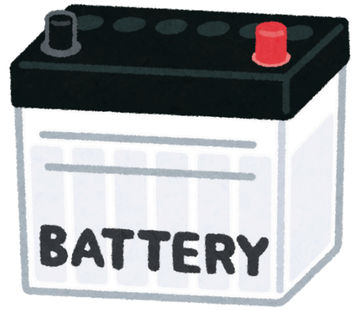
[Submitted on 15 Aug 2025]

Stealing Accuracy: Predicting Day-ahead Electricity Prices with Temporal Hierarchy Forecasting (THieF)

Arkadiusz Lipiecki, Kaja Bilinska, Nicolaos Kourentzes, Rafal Weron

Part II - Stealing profits

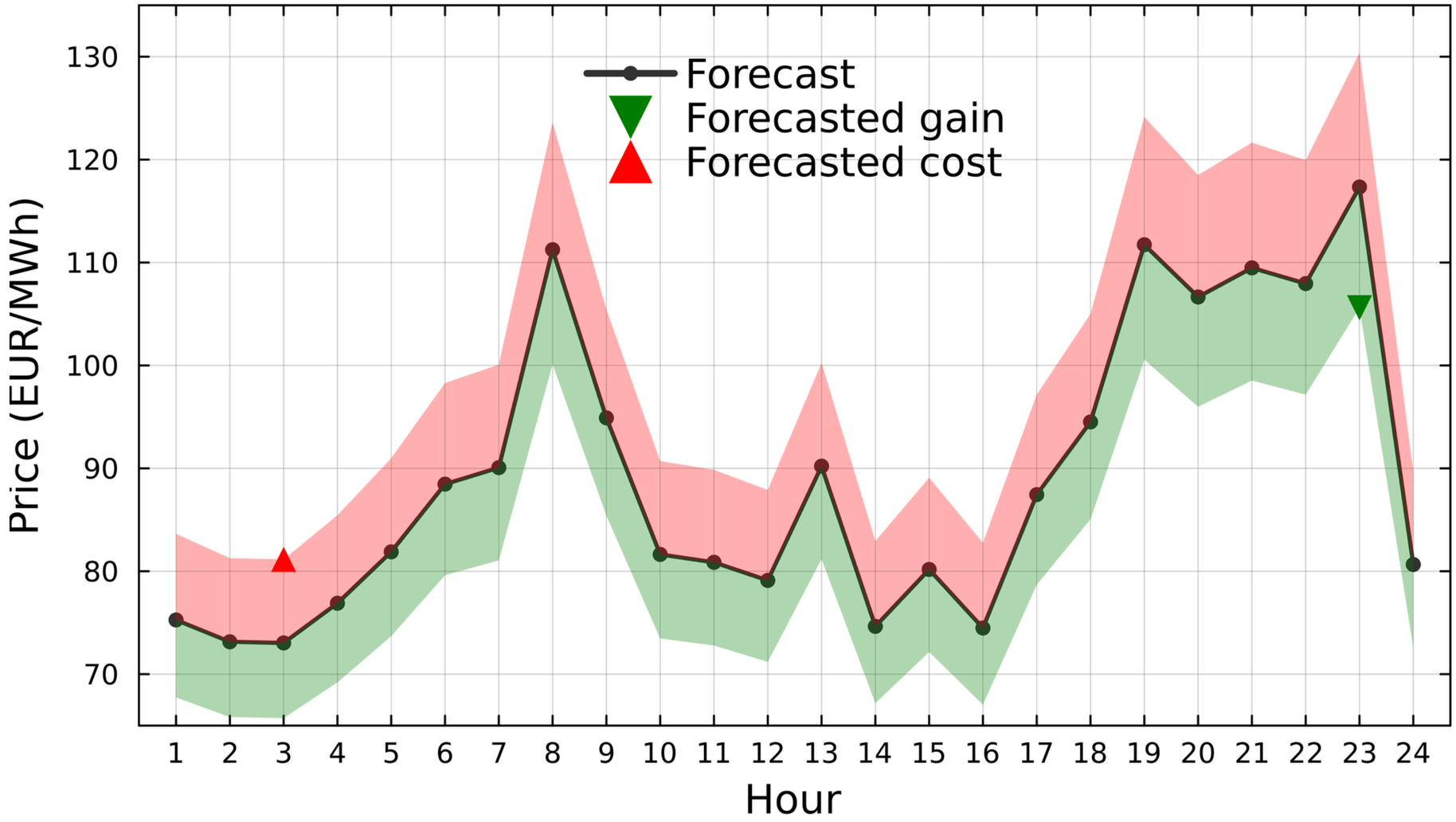




Capacity	**1 MWh**
C-rating	**1**
Efficiency	**0.9**

Profit = Efficiency × Sell price × Capacity − (1/Efficiency) × Buy price × Capacity − Other costs

Part II - Stealing profits



Profit = Efficiency × Sell price × Capacity − (1/Efficiency) × Buy price × Capacity − Other costs

New temporal hierarchy - effective price spreads

Profit = Efficiency × Sell price × Capacity – (1/Efficiency) × Buy price × Capacity – Other costs

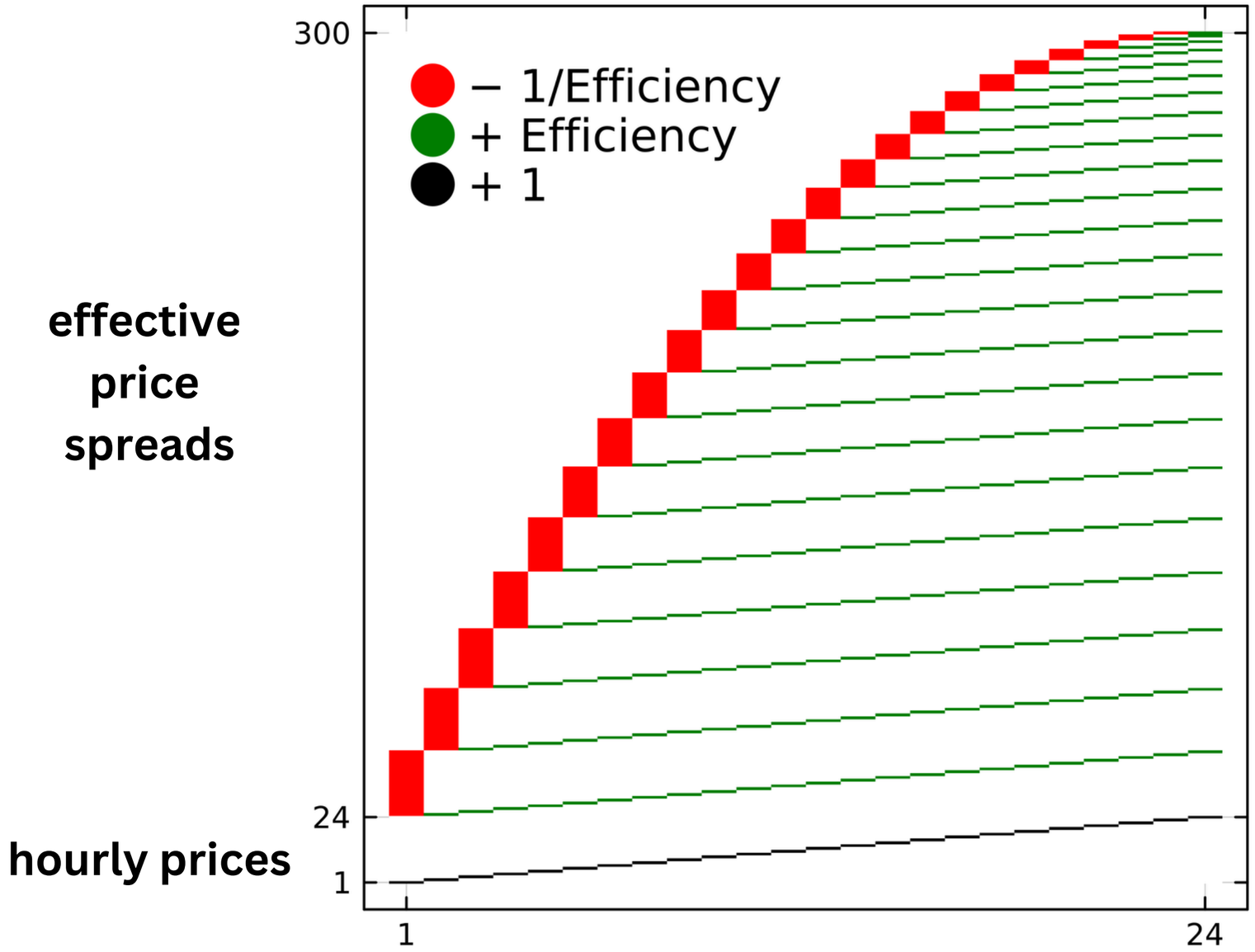
Spread(H1, H2) = Efficiency × Price(H2) - (1/Efficiency) × Price(H1)
such that H2 > H1

Serafin & Weron (2025)

300 base forecasts: 24 hourly prices and 276 effective price spreads

$$\hat{p}_{d,h} = f\left(p_{d-1,h}, \ldots, p_{d-7,h}, p^{min}_{d-1}, p^{max}_{d-1}, \hat{L}_{d,h}, \hat{W}_{d,h}, \mathrm{API}_{d-2}, \mathrm{TTF}_{d-2}, \mathrm{D}^{(1)}_{d}, \ldots, \mathrm{D}^{(7)}_{d}\right)$$

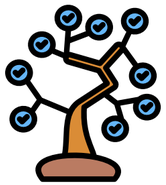
Summing matrix

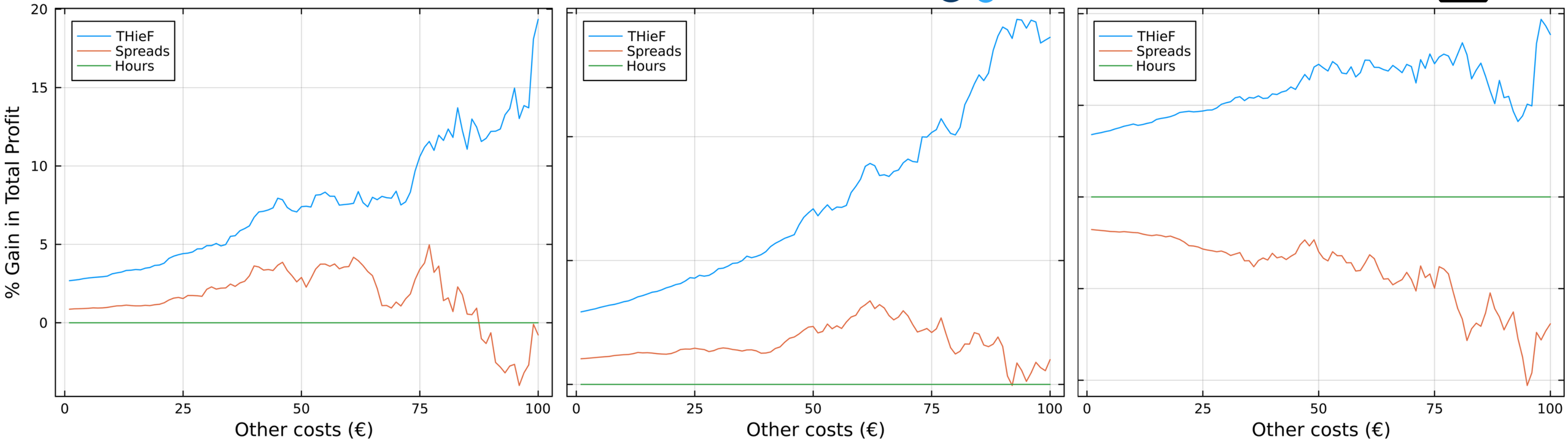


Stealing Profits

ARX 

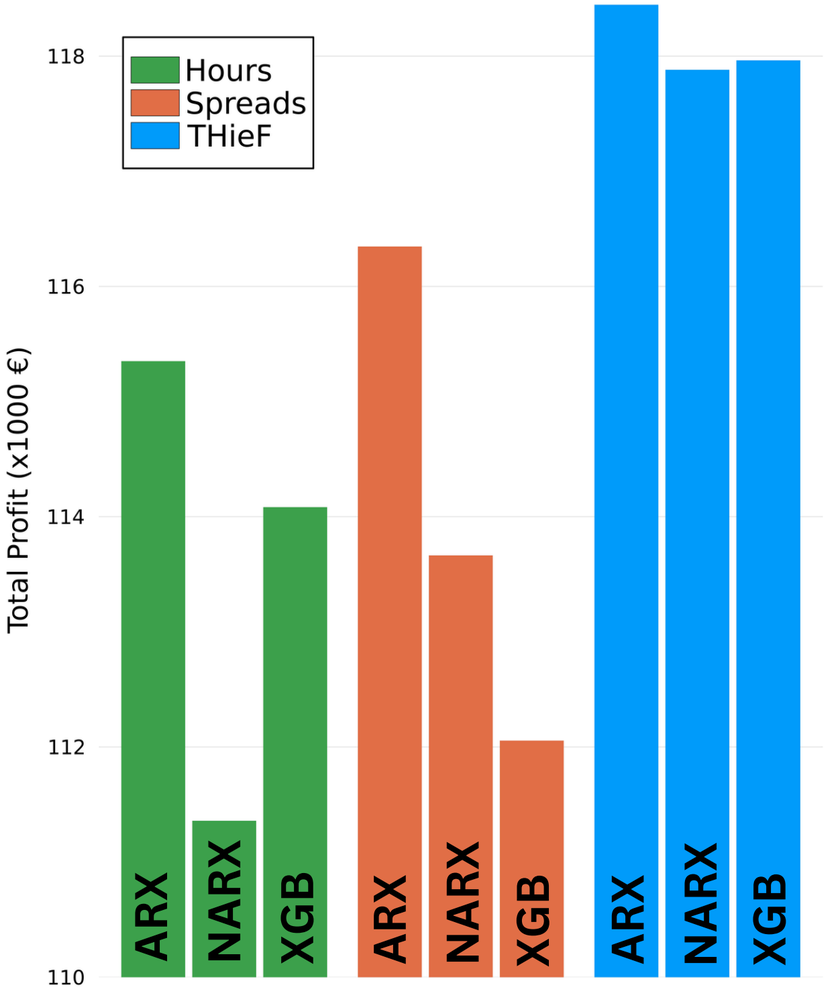
NARX 

XGB 

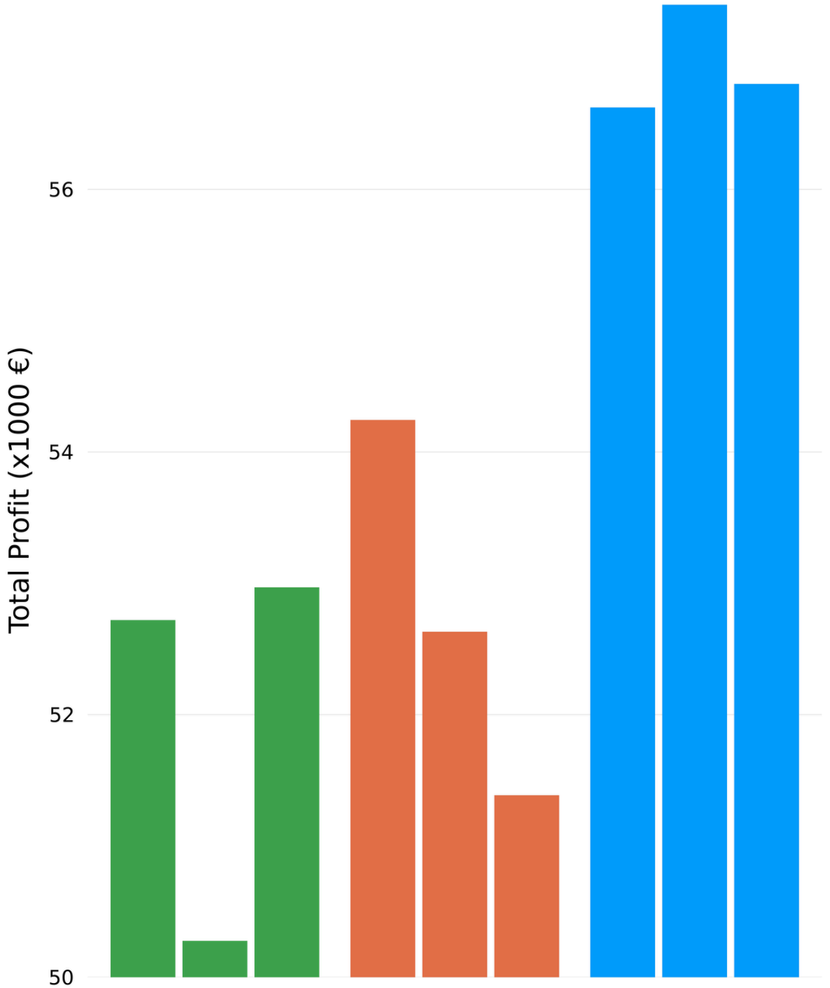


Profit = Efficiency × Sell price × Capacity − (1/Efficiency) × Buy price × Capacity − Other costs

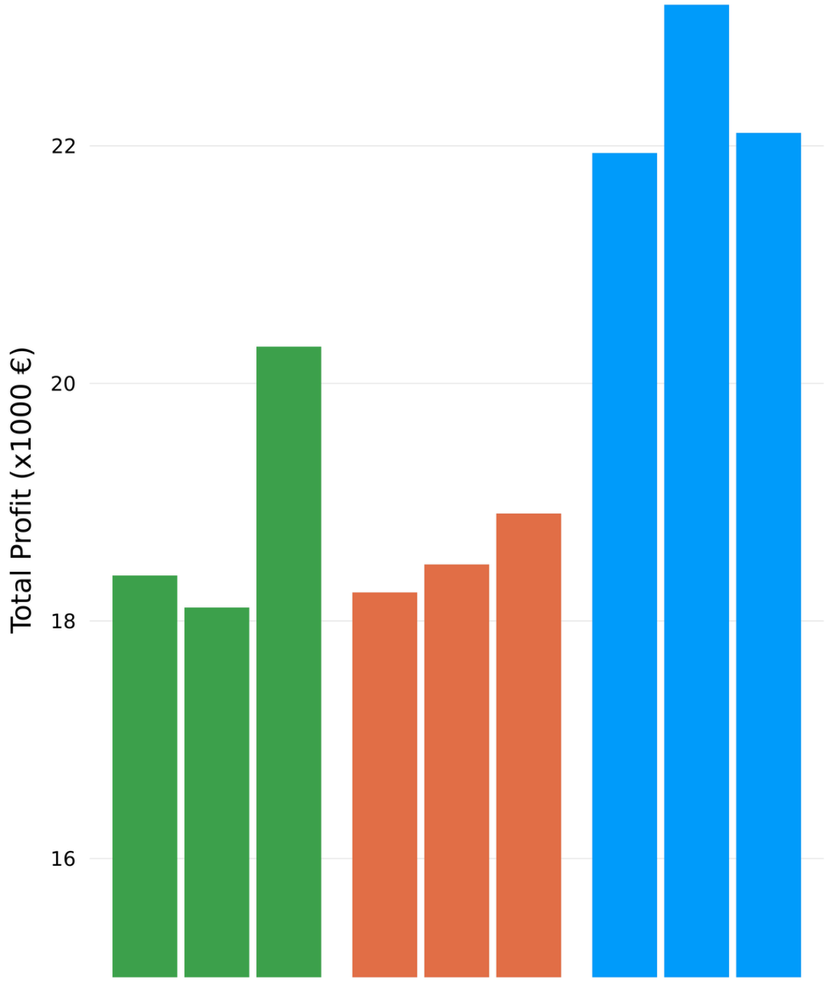
Stealing Profits



Other costs = 0 €



Other costs = 50 €

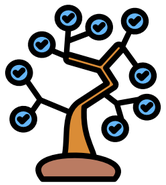


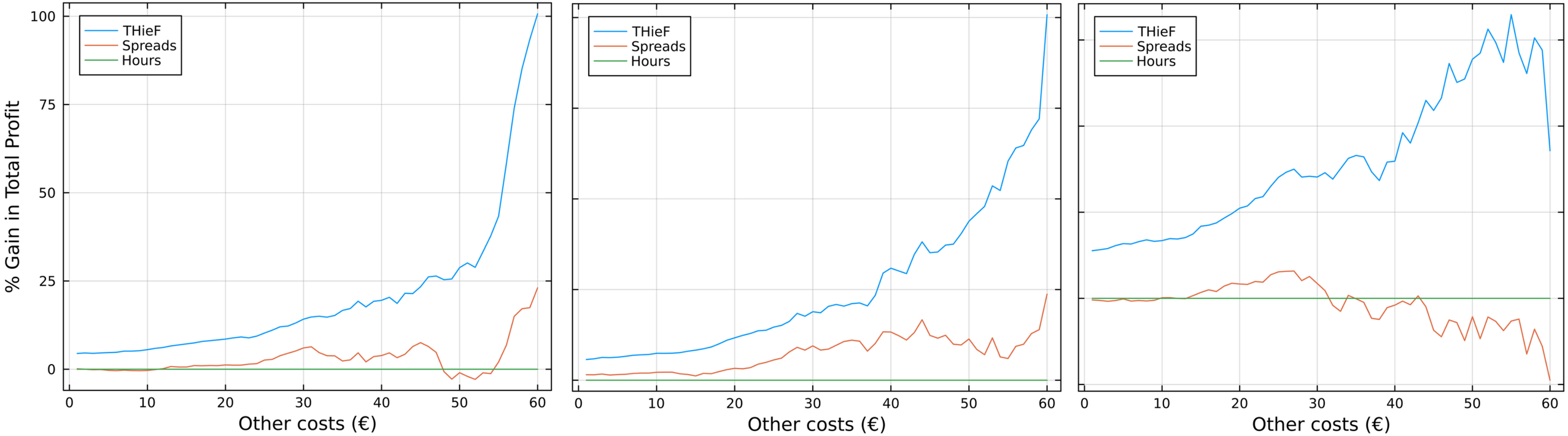
Other costs = 100 €

Stealing Profits

ARX 

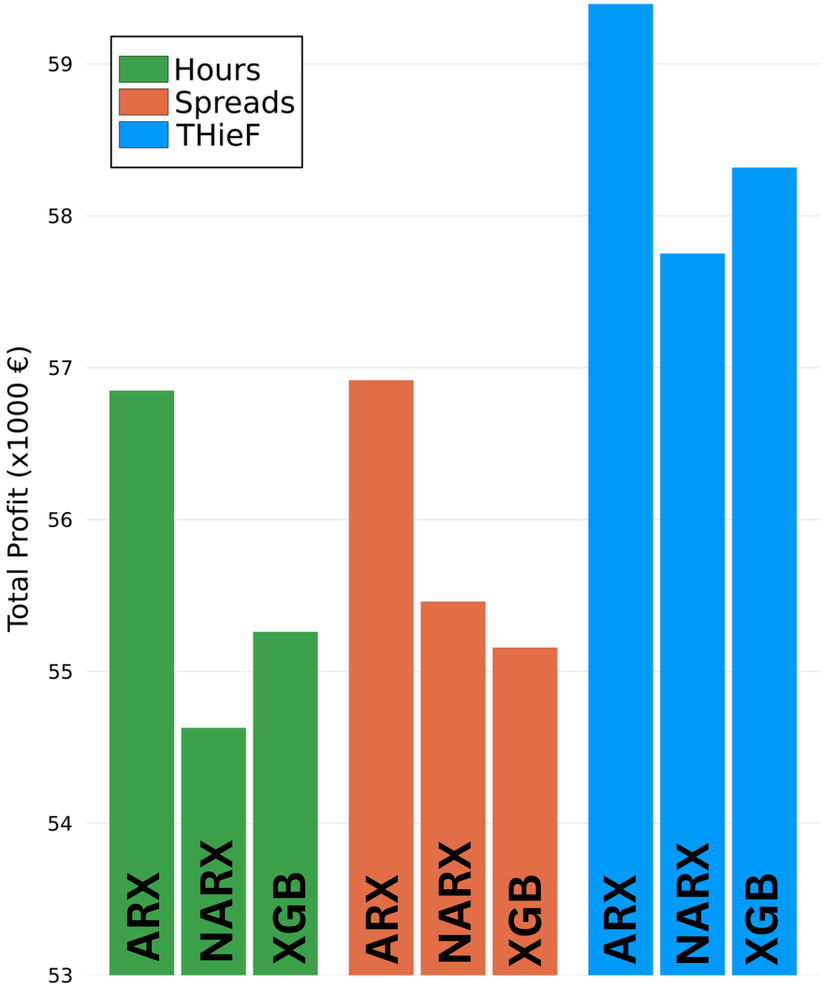
NARX 

XGB 

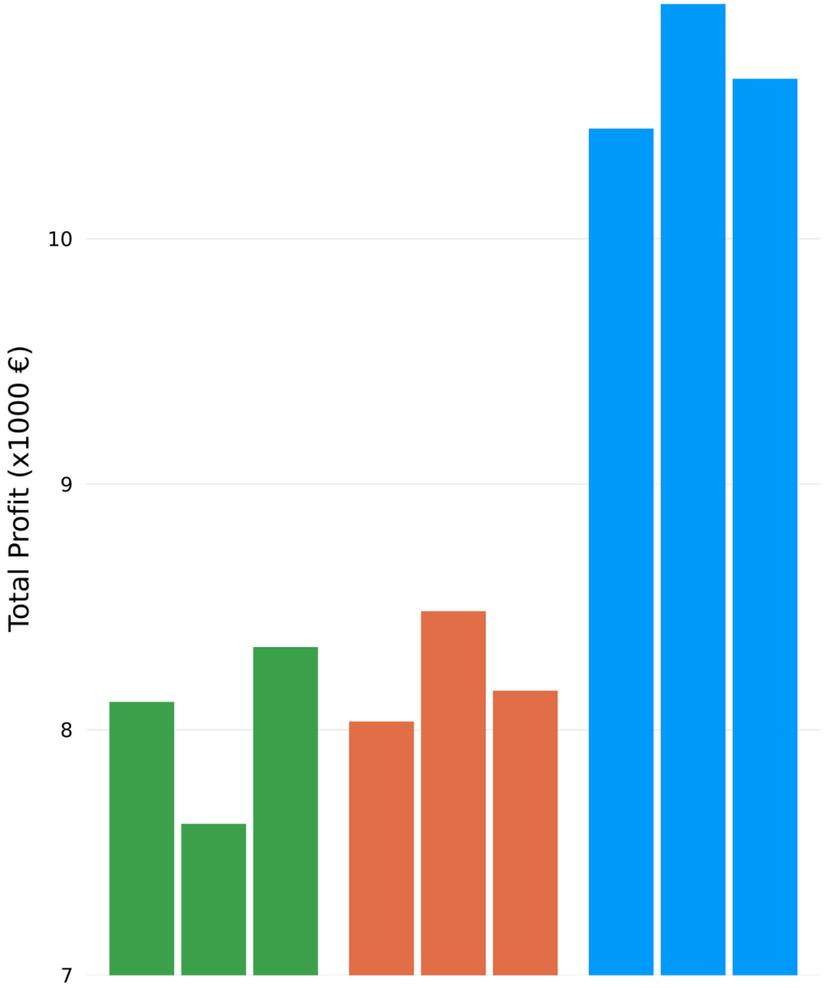


Profit = Efficiency × Sell price × Capacity − (1/Efficiency) × Buy price × Capacity − Other costs

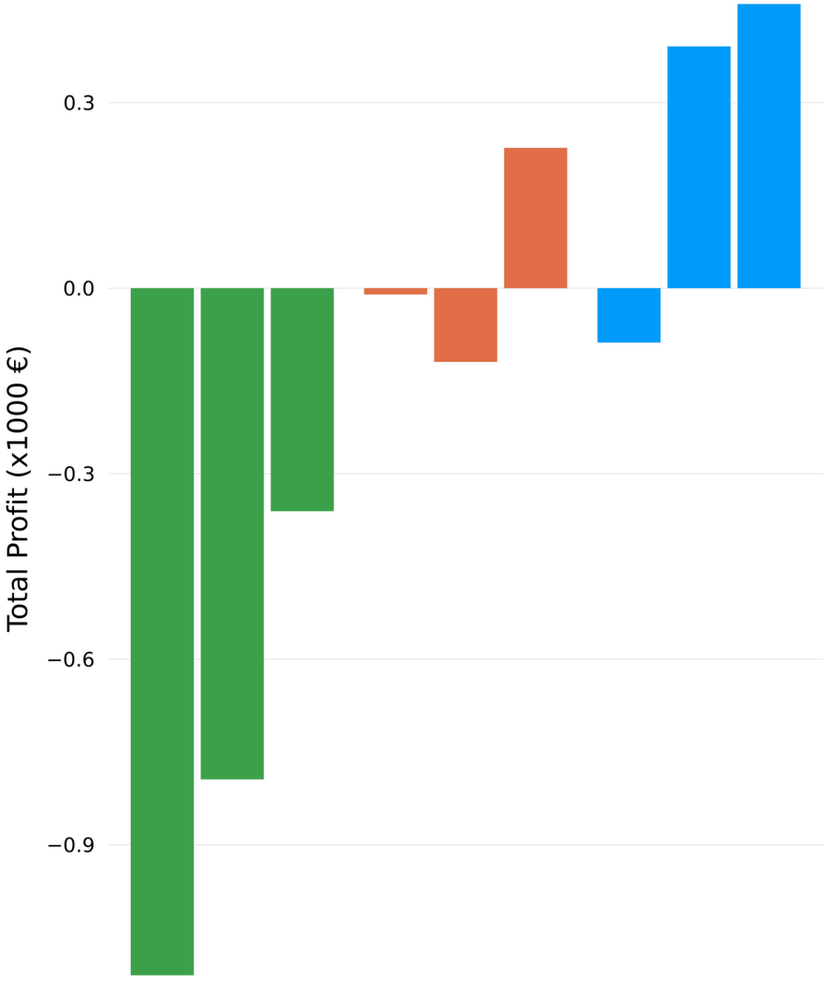
Stealing Profits



Other costs = 0 €



Other costs = 50 €



Other costs = 100 €