A Data-driven Forecasting Model for an Aggregator of Electric Vehicles via Inverse Optimization

Ricardo Fernández-Blanco, Juan Miguel Morales, Salvador Pineda, Álvaro Porras

Postdoctoral researcher
OASYS group, University of Malaga

Ricardo.FCarramolino@gmail.com

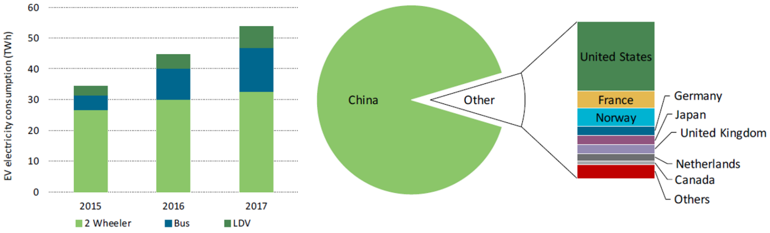
October 20th, 2019

European Research Council
Established by the European Commission

Optimization and Analytics for Sustainable energY Systems

Motivation

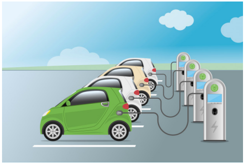
- Growing electrification of the road transport: Greener cities and transport system
- Estimated global electricity demand from EVs was around 54 TWh in 2017

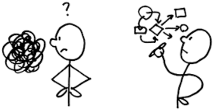


Source: *International Energy Agency.*

Motivation

- Electromobility will impact on the power system operation and planning
- Aggregators play a key role to manage a fleet of EVs and suitable tools are still to be developed





Context and Challenges

Within a market-based context, the aggregator agents face several challenges:

Forecast	How can they forecast the EV-fleet power in the short-term?
Bidding	How to come up with a bidding curve to participate in the electricity market?
Data	How can they make use of past data about electricity prices and driving patterns?

Forecasting Model

- We want to forecast the EV-fleet (aggregate) power p_t.
  Past observations: driving patterns, prices λ'_{t-l} or p'_{t-l}, $\forall l = 1, 2, \ldots$.
- We maximize the aggregate welfare of the EV fleet

Forecasting model

$$\max_{p_b} \quad \sum_{b \in \mathcal{B}} p_b \left(m_b - \lambda \right)$$

subject to:

$$\underline{P} \leq \textstyle\sum_{b \in \mathcal{B}} p_b \leq \overline{P} \quad \Longrightarrow \quad (\underline{\beta}, \overline{\beta})$$

$$0 \leq p_b \leq \overline{E}_b, \forall b \in \mathcal{B}^c \quad \Longrightarrow \quad (\underline{\phi}_b^c, \overline{\phi}_b^c)$$

$$\underline{E}_b \leq p_b \leq 0, \forall b \in \mathcal{B}^d \quad \Longrightarrow \quad (\underline{\phi}_b^d, \overline{\phi}_b^d)$$

Forecasting Model

- We want to forecast the EV-fleet (aggregate) power p_t.
  Past observations: driving patterns, prices λ'_{t-l} or p'_{t-l}, $\forall l = 1, 2, \ldots$.
- We maximize the aggregate welfare of the EV fleet

Forecasting model

Parameters to be estimated in pink $\Longrightarrow$ Inverse optimization

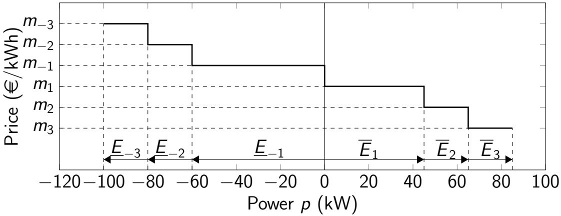
$$\max_{p_b} \quad \sum_{b \in \mathcal{B}} p_b \left(m_b - \lambda \right)$$

subject to:

$$\begin{aligned}
&\underline{P} \leq \textstyle\sum_{b \in \mathcal{B}} p_b \leq \overline{P} && \Longrightarrow && (\underline{\beta}, \overline{\beta}) \\
&0 \leq p_b \leq \overline{E}_b, \forall b \in \mathcal{B}^c && \Longrightarrow && (\underline{\phi}^c_b, \overline{\phi}^c_b) \\
&\underline{E}_b \leq p_b \leq 0, \forall b \in \mathcal{B}^d && \Longrightarrow && (\underline{\phi}^d_b, \overline{\phi}^d_b)
\end{aligned}$$

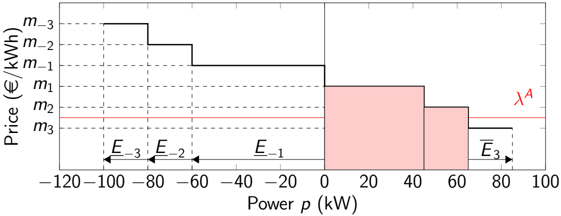
Forecasting Model

Let us assume a three-block stepwise offer (bid) price function of the EVs aggregator:



Forecasting Model

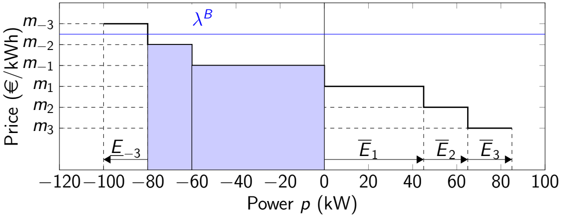
Let us assume a three-block stepwise offer (bid) price function of the EVs aggregator:



- If $\lambda = \lambda^A$, then the aggregator will consume power and the accepted blocks will be $\{1, 2\}$

Forecasting Model

Let us assume a three-block stepwise offer (bid) price function of the EVs aggregator:



- If $\lambda = \lambda^B$, then the aggregator will produce power and the accepted blocks will be $\{-1, -2\}$

Accounting for Past Information

We use kernel regression to estimate power bounds $\underline{P}_t$ and $\overline{P}_t$, and marginal utility $m_{b,t}$:

$$\left.\begin{aligned}
\underline{P}_t &= \underline{\mu} + \sum_{\tau \in \Omega^{tr}} \underline{\alpha}_\tau K_{t,\tau}, \quad \forall t \in \mathcal{T} \\
\overline{P}_t &= \overline{\mu} + \sum_{\tau \in \Omega^{tr}} \overline{\alpha}_\tau K_{t,\tau}, \quad \forall t \in \mathcal{T} \\
m_{b,t} &= \nu_b + \sum_{\tau \in \Omega^{tr}} \rho_\tau K_{t,\tau}, \quad \forall t \in \mathcal{T}
\end{aligned}\right\} \text{Kernel regression functions}$$

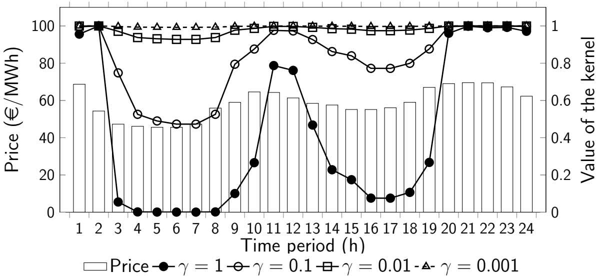
$K_{t,\tau}$ is a kernel function on two feature vectors at time periods t and τ

similarity measure!

Example of a Kernel

Gaussian kernel: $K_{t,\tau} = K\left(\boldsymbol{z}_t, \boldsymbol{z}_\tau\right) = e^{-\gamma\|\boldsymbol{z}_t - \boldsymbol{z}_\tau\|_2^2}, \quad \forall t \in \mathcal{T}, \tau \in \Omega^{tr}$

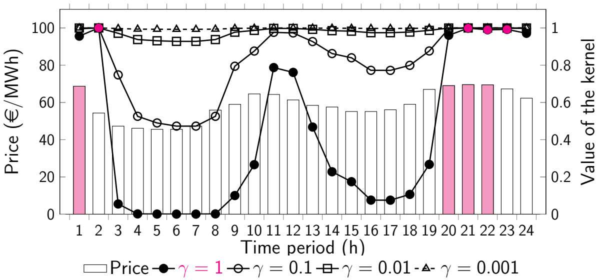
Let us assume that $\boldsymbol{z}_t = \lambda_{t-1}$ and that $\boldsymbol{z}_{\tau=2} = \lambda_1$



Example of a Kernel

Gaussian kernel: $K_{t,\tau} = K\left(\boldsymbol{z}_t, \boldsymbol{z}_\tau\right) = e^{-\gamma\|\boldsymbol{z}_t - \boldsymbol{z}_\tau\|_2^2}, \quad \forall t \in \mathcal{T}, \tau \in \Omega^{tr}$

Let us assume that $\boldsymbol{z}_t = \lambda_{t-1}$ and that $\boldsymbol{z}_{\tau=2} = \lambda_1$

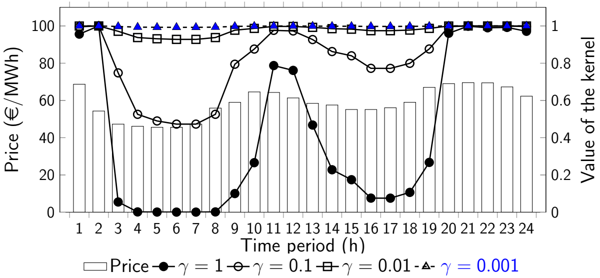


High values of γ (low variance) lead to kernel values equal to 1 just when the regressors are very close to each other

Example of a Kernel

Gaussian kernel: $K_{t,\tau} = K\left(\boldsymbol{z}_t, \boldsymbol{z}_\tau\right) = e^{-\gamma\|\boldsymbol{z}_t - \boldsymbol{z}_\tau\|_2^2}, \quad \forall t \in \mathcal{T}, \tau \in \Omega^{tr}$

Let us assume that $\boldsymbol{z}_t = \lambda_{t-1}$ and that $\boldsymbol{z}_{\tau=2} = \lambda_1$



Low values of γ (high variance) lead to kernel values equal to 1 even when the regressors are very different from each other

Two-step Estimation Procedure

- Then, how do we estimate $\Phi = \{\underline{E}_{b,t}, \overline{E}_{b,t}, m_{b,t}, \underline{P}_t, \overline{P}_t\}$ and the corresponding coefficient estimates $\underline{\mu}, \underline{\alpha}_t, \overline{\mu}, \overline{\alpha_t}, \nu_b, \rho_t$?
- Inverse optimization methodology: two convex programs!

Feasibility problem

What?
Power bounds $\underline{P}_t, \overline{P_t}$ and $\underline{\mu}, \underline{\alpha}_t, \overline{\mu}, \overline{\alpha_t}$

Why?
Feasibility

How?
Shaping the power bounds based on observed EV-fleet power values

Optimality problem

What?
Marginal utility $m_{b,t}$ and ν_b, ρ_t

Why?
Optimality

How?
Minimizing the duality gap of the forecasting problem

Feasibility Problem

The feasibility problem is formulated as:

$$\min_{\Xi^{fp}} \sum_{t \in \Omega^{tr}} H \left(\overline{\xi}_t^- + \underline{\xi}_t^- \right) + \sum_{t \in \Omega^{tr}} (1 - H) \left(\overline{\xi}_t^+ + \underline{\xi}_t^+ \right)$$

subject to:

$$\overline{P}_t - p'_t = \overline{\xi}_t^+ - \overline{\xi}_t^-, \quad \forall t \in \Omega^{tr}$$

$$p'_t - \underline{P}_t = \underline{\xi}_t^+ - \underline{\xi}_t^-, \quad \forall t \in \Omega^{tr}$$

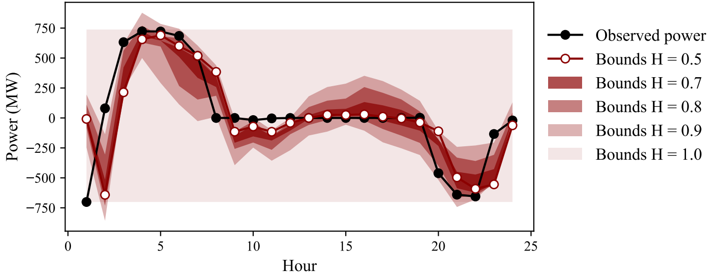
$$\overline{P}_t \geq \underline{P}_t, \quad \forall t \in \Omega^{tr}$$

Kernel regression functions (Gaussian - γ)

$$\overline{\xi}_t^+, \underline{\xi}_t^+, \overline{\xi}_t^-, \underline{\xi}_t^- \geq 0, \quad \forall t \in \Omega^{tr},$$

Feasibility Problem

Graphical explanation of the feasibility problem:



Parameter $H \in [0, 1)$ controls the width of the power bounds and thus the price-responsiveness of the EV fleet

Optimality Problem

Let us recall the forecasting problem and its dual counterpart:

Primal problem

$$\max_{\Xi^P} \quad \sum_{b \in \mathcal{B}} p_b \left(m_b - \lambda \right)$$

subject to:

$$\underline{P} \leq \sum_{b \in \mathcal{B}} p_b \leq \overline{P}$$

$$0 \leq p_b \leq \overline{E}_b, \forall b \in \mathcal{B}^c$$

$$\underline{E}_b \leq p_b \leq 0, \forall b \in \mathcal{B}^d$$

where $\Xi^P = \{p_b\}$

Dual problem

$$\min_{\Xi^D} \quad \widehat{\overline{P}}\,\overline{\beta} - \widehat{\underline{P}}\,\underline{\beta} + \sum_{b \in \mathcal{B}^c} \widehat{\overline{E}}_b \overline{\phi}^c_b - \sum_{b \in \mathcal{B}^d} \widehat{\underline{E}}_b \underline{\phi}^d_b$$

subject to:

$$-\underline{\phi}^c_b + \overline{\phi}^c_b - \underline{\beta} + \overline{\beta} = m_b - \lambda, \forall b \in \mathcal{B}^c$$

$$-\underline{\phi}^d_b + \overline{\phi}^d_b - \underline{\beta} + \overline{\beta} = m_b - \lambda, \forall b \in \mathcal{B}^d$$

$$\underline{\beta}, \overline{\beta}, \underline{\phi}^c_b, \overline{\phi}^c_b, \underline{\phi}^d_b, \overline{\phi}^d_b \geq 0$$

where $\Xi^D = \{\underline{\beta}, \overline{\beta}, \underline{\phi}^c_b, \overline{\phi}^c_b, \underline{\phi}^d_b, \overline{\phi}^d_b\}$

Optimality Problem

Then, the optimality problem can be formulated as:

$$\min_{\Xi^{op}} \quad \sum_{t \in \Omega^{tr}} \epsilon_t$$

subject to:

Dual objective function $- \epsilon_t =$ Primal objective function, $\quad \forall t \in \Omega^{tr}$

Dual feasibility constraints

$$m_{b,t} = \nu_b + \sum_{\tau \in \Omega^{tr}} \rho_\tau K_{t,\tau}, \quad \forall t \in \Omega^{tr}$$

$$\nu_b \geq \nu_{b+1}, \quad \forall b \in \mathcal{B} \setminus \{b = N_B\}$$

Computation of Hyper-parameters

- We have presented all the *tools* needed to forecast the EV-fleet power
- Feasibility and optimality problems rely on the knowledge of hyper-parameters: H, γ
- Grid search technique by using training and validation sets

Comparison Methodologies

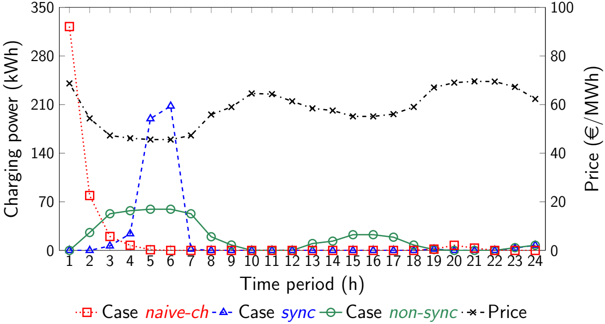
Acronym	**Methodology**
kio	**Kernel-based inverse optimization approach**
svr	Support vector regression
krr	Kernel-ridge regression
lio	Inverse optimization approach with linear kernels
h-naive, *d-naive*, *w-naive*	Persistence models

Data

- No real-life data $\Longrightarrow$ Simulation of the EV-fleet behavior
- Residential aggregator with 100 EVs
- Synthetic database for a pool of EVs
  - Availability profiles ($\varsigma_{v,t}$) and energy required for transportation: National Household Travel Survey 2017
  - Electricity prices λ_t: ENTSO-e Transparency platform

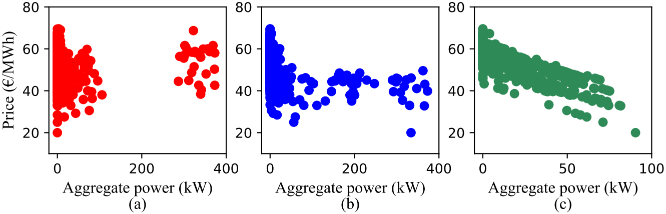
Cases: Without V2G Capabilities

- *naive-ch*: EVs satisfy their energy needs with a naive charging $\neq f(\lambda)$
- *sync*: Charging is highly synchronized $= f(\lambda)$
- *non-sync*: Charging synchronization is avoided $= f(\lambda)$



Cases

- Power versus price relationship

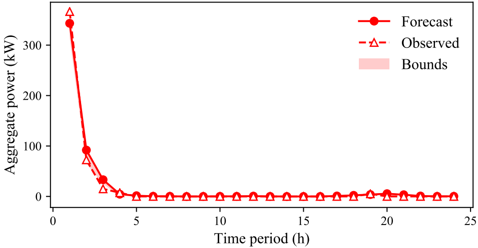


- Explanatory variables per case:
    - *naive-ch*: p_{t-l} and $\sum_v \varsigma_{v,t-l}$, $\forall l = 1...6$, and 5 categorical variables
    - *sync* and *non-sync*: λ_{t-l} and p_{t-l}, $\forall l = 1...6$
- We assume 6 energy blocks
- Training Ω^{tr}, validation Ω^{v}, test Ω^{test} sets: 672, 168, 168 h

Results for the *naive-ch* Case

kio	*lio*
$H^* = 0.64$	$H^* = 0.91$
$\gamma^* = 0.1$	

Model	RMSE	MAE
kio	8.6	3.7
krr	9.0	3.5
svr	10.4	5.7
lio	16.8	6.4
h-naive	90.3	29.3
d-naive	13.2	4.8
w-naive	10.8	4.6

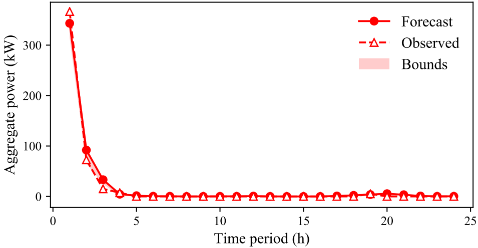


- Low value of H:
  - Bounds are almost coincident
  - No dependence on the price
  - Power directly explained by estimating the bounds
- *kio* reduces the RMSE by 4.4% and 17.3% compared to *krr* and *svr*

Results for the *naive-ch* Case

kio	*lio*
$H^* = 0.64$	$H^* = 0.91$
$\gamma^* = 0.1$	

Model	RMSE	MAE
kio	8.6	3.7
krr	9.0	3.5
svr	10.4	5.7
lio	16.8	6.4
h-naive	90.3	29.3
d-naive	13.2	4.8
w-naive	10.8	4.6



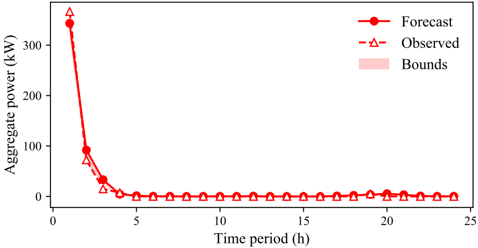
- Low value of H:
  - Bounds are almost coincident
  - No dependence on the price
  - Power directly explained by estimating the bounds

- *kio* reduces the RMSE by 4.4% and 17.3% compared to *krr* and *svr*

Results for the *naive-ch* Case

kio	*lio*
$H^* = 0.64$	$H^* = 0.91$
$\gamma^* = 0.1$	

Model	RMSE	MAE
kio	8.6	3.7
krr	9.0	3.5
svr	10.4	5.7
lio	16.8	6.4
h-naive	90.3	29.3
d-naive	13.2	4.8
w-naive	10.8	4.6



- Low value of H:
  - Bounds are almost coincident
  - No dependence on the price
  - Power directly explained by estimating the bounds

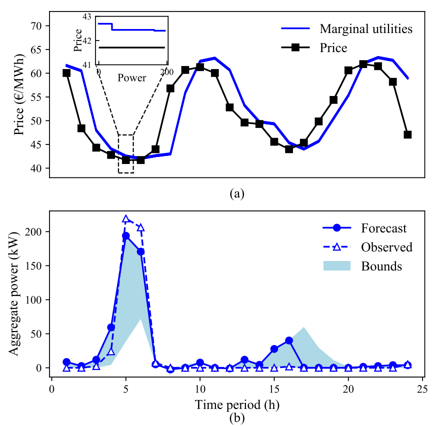
- *kio* reduces the RMSE by 4.4% and 17.3% compared to *krr* and *svr*

Results for the *sync* Case

kio	*lio*
$H^* = 0.82$	$H^* = 0.89$
$\gamma^* = 0.1$	

Model	RMSE	MAE
kio	35.2	13.3
krr	35.5	15.7
svr	41.7	14.7
lio	59.3	23.0
h-naive	72.7	25.3
d-naive	64.8	22.3
w-naive	49.1	15.7

- High value of H:
  - Bounds are wider and shape the forecast
  - Power is sensitive to the price



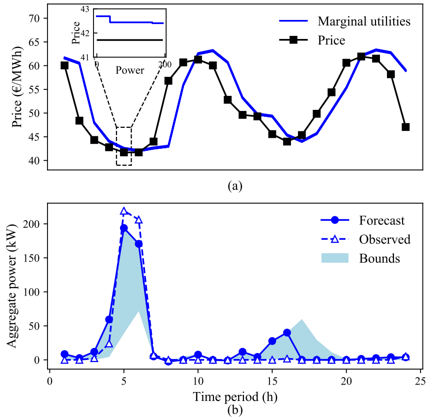
- H values for *kio* and *lio* are quite similar
- *kio* outperforms *lio*: 40.6% ↓ and 42.2% ↓
- *kio* outperforms *w-naive*: 28.3% ↓ and 15.3% ↓
- *kio* performance is comparable to other machine-learning techniques

Results for the *sync* Case

kio	*lio*
$H^* = 0.82$	$H^* = 0.89$
$\gamma^* = 0.1$	

Model	RMSE	MAE
kio	35.2	13.3
krr	35.5	15.7
svr	41.7	14.7
lio	59.3	23.0
h-naive	72.7	25.3
d-naive	64.8	22.3
w-naive	49.1	15.7

- High value of H:
  - Bounds are wider and shape the forecast
  - Power is sensitive to the price

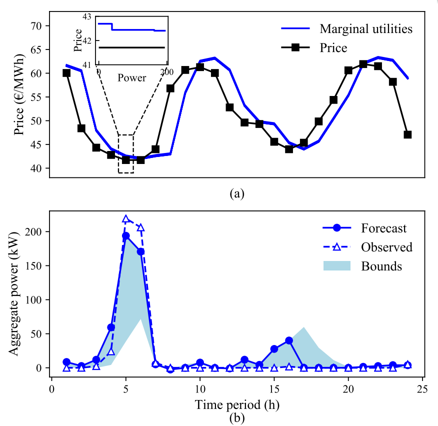


- H values for *kio* and *lio* are quite similar
- *kio* outperforms *lio*: 40.6% ↓ and 42.2% ↓
- *kio* outperforms *w-naive*: 28.3% ↓ and 15.3% ↓
- *kio* performance is comparable to other machine-learning techniques

Results for the *sync* Case

kio	*lio*
$H^* = 0.82$	$H^* = 0.89$
$\gamma^* = 0.1$	

Model	RMSE	MAE
kio	35.2	13.3
krr	35.5	15.7
svr	41.7	14.7
lio	59.3	23.0
h-naive	72.7	25.3
d-naive	64.8	22.3
w-naive	49.1	15.7



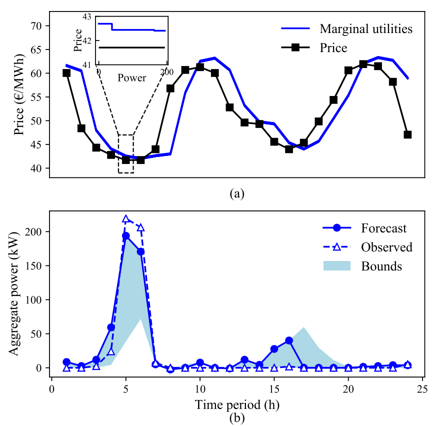
- High value of H:
  - Bounds are wider and shape the forecast
  - Power is sensitive to the price

- H values for *kio* and *lio* are quite similar
- *kio* outperforms *lio*: 40.6% ↓ and 42.2% ↓
- *kio* outperforms *w-naive*: 28.3% ↓ and 15.3% ↓
- *kio* performance is comparable to other machine-learning techniques

Results for the *sync* Case

kio	*lio*
$H^* = 0.82$	$H^* = 0.89$
$\gamma^* = 0.1$	

Model	RMSE	MAE
kio	35.2	13.3
krr	35.5	15.7
svr	41.7	14.7
lio	59.3	23.0
h-naive	72.7	25.3
d-naive	64.8	22.3
w-naive	49.1	15.7



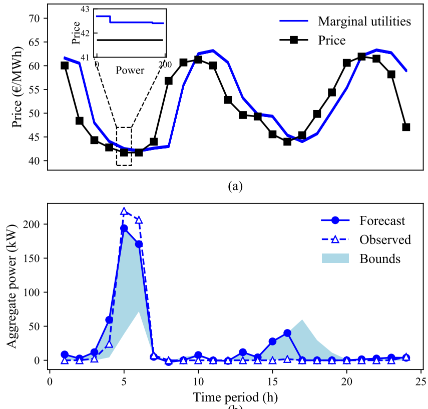
- High value of H:
  - Bounds are wider and shape the forecast
  - Power is sensitive to the price

- H values for *kio* and *lio* are quite similar
- *kio* outperforms *lio*: 40.6% ↓ and 42.2% ↓
- *kio* outperforms *w-naive*: 28.3% ↓ and 15.3% ↓
- *kio* performance is comparable to other machine-learning techniques

Results for the *sync* Case

kio	*lio*
$H^* = 0.82$	$H^* = 0.89$
$\gamma^* = 0.1$	

Model	RMSE	MAE
kio	35.2	13.3
krr	35.5	15.7
svr	41.7	14.7
lio	59.3	23.0
h-naive	72.7	25.3
d-naive	64.8	22.3
w-naive	49.1	15.7



(a)
(b)

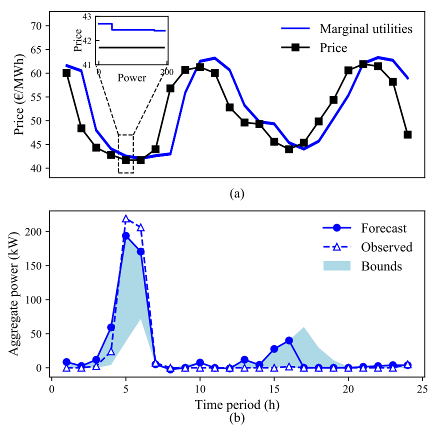
- High value of H:
  - Bounds are wider and shape the forecast
  - Power is sensitive to the price
- H values for *kio* and *lio* are quite similar
- *kio* outperforms *lio*: 40.6% ↓ and 42.2% ↓
- *kio* outperforms *w-naive*: 28.3% ↓ and 15.3% ↓
- *kio* performance is comparable to other machine-learning techniques

Results for the *sync* Case

kio	*lio*
$H^* = 0.82$	$H^* = 0.89$
$\gamma^* = 0.1$	

Model	RMSE	MAE
kio	35.2	13.3
krr	35.5	15.7
svr	41.7	14.7
lio	59.3	23.0
h-naive	72.7	25.3
d-naive	64.8	22.3
w-naive	49.1	15.7

- High value of H:
  - Bounds are wider and shape the forecast
  - Power is sensitive to the price

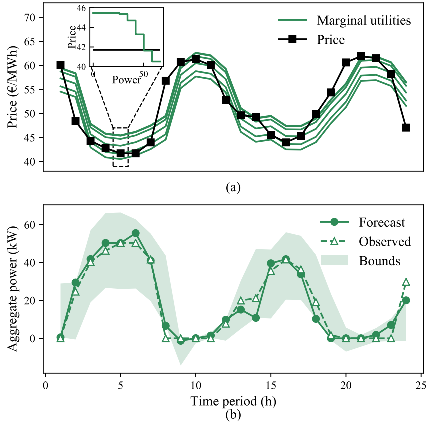


- H values for *kio* and *lio* are quite similar
- *kio* outperforms *lio*: 40.6% ↓ and 42.2% ↓
- *kio* outperforms *w-naive*: 28.3% ↓ and 15.3% ↓
- *kio* performance is comparable to other machine-learning techniques

Results for the *non-sync* Case

kio	*lio*
$H^* = 0.94$	$H^* = 0.94$
$\gamma^* = 0.01$	

Model	RMSE	MAE
kio	5.5	3.8
krr	7.4	5.2
svr	7.6	5.0
lio	5.9	3.9
h-naive	11.3	7.1
d-naive	17.3	13.3
w-naive	13.0	9.1



- Higher value of H:
  - Wider bounds than for the *sync* case
  - Power is sensitive to the price
  - Wider range of marginal utilities

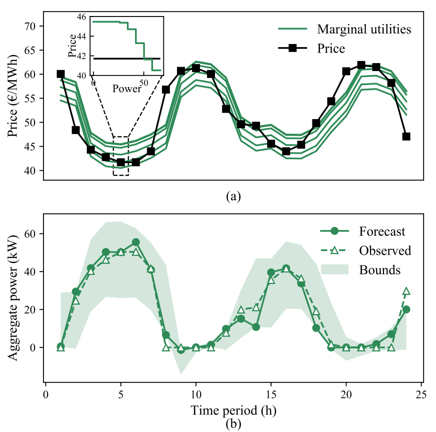
- H values for *kio* and *lio* are quite similar
- *kio* and *lio* give rise to the least errors
- *kio* outperforms *h-naive*, *krr*, *svr*: 51.3% ↓, 25.7% ↓, 27.6% ↓

Results for the *non-sync* Case

kio	*lio*
$H^* = 0.94$	$H^* = 0.94$
$\gamma^* = 0.01$	

Model	RMSE	MAE
kio	5.5	3.8
krr	7.4	5.2
svr	7.6	5.0
lio	5.9	3.9
h-naive	11.3	7.1
d-naive	17.3	13.3
w-naive	13.0	9.1

- Higher value of H:
  - Wider bounds than for the *sync* case
  - Power is sensitive to the price
  - Wider range of marginal utilities

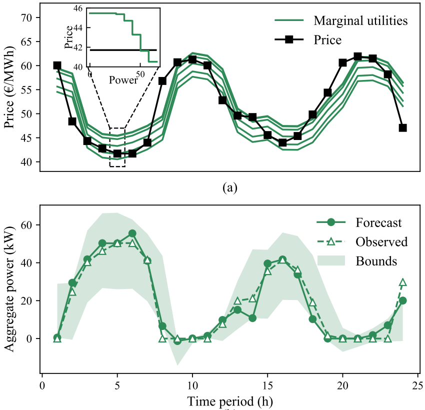


- H values for *kio* and *lio* are quite similar
- *kio* and *lio* give rise to the least errors
- *kio* outperforms *h-naive*, *krr*, *svr*: 51.3% ↓, 25.7% ↓, 27.6% ↓

Results for the *non-sync* Case

kio	*lio*
$H^* = 0.94$	$H^* = 0.94$
$\gamma^* = 0.01$	

Model	RMSE	MAE
kio	5.5	3.8
krr	7.4	5.2
svr	7.6	5.0
lio	5.9	3.9
h-naive	11.3	7.1
d-naive	17.3	13.3
w-naive	13.0	9.1



(b)

- Higher value of *H*:
  - Wider bounds than for the *sync* case
  - Power is sensitive to the price
  - Wider range of marginal utilities

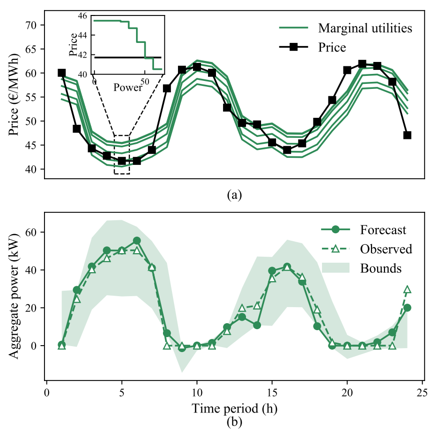
- *H* values for *kio* and *lio* are quite similar
- *kio* and *lio* give rise to the least errors
- *kio* outperforms *h-naive*, *krr*, *svr*: 51.3% ↓, 25.7% ↓, 27.6% ↓

Results for the *non-sync* Case

kio	*lio*
$H^* = 0.94$	$H^* = 0.94$
$\gamma^* = 0.01$	

Model	RMSE	MAE
kio	5.5	3.8
krr	7.4	5.2
svr	7.6	5.0
lio	5.9	3.9
h-naive	11.3	7.1
d-naive	17.3	13.3
w-naive	13.0	9.1

- Higher value of H:
  - Wider bounds than for the *sync* case
  - Power is sensitive to the price
  - Wider range of marginal utilities



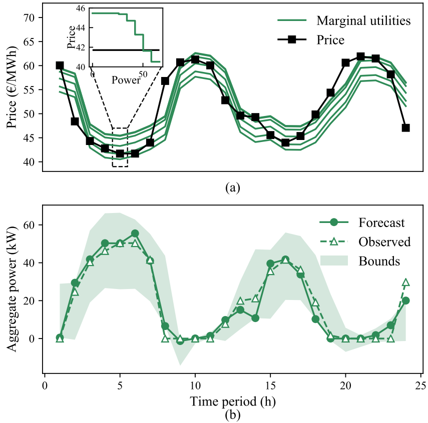
- H values for *kio* and *lio* are quite similar
- *kio* and *lio* give rise to the least errors
- *kio* outperforms *h-naive*, *krr*, *svr*: 51.3% ↓, 25.7% ↓, 27.6% ↓

Results for the *non-sync* Case

kio	*lio*
$H^* = 0.94$	$H^* = 0.94$
$\gamma^* = 0.01$	

Model	RMSE	MAE
kio	5.5	3.8
krr	7.4	5.2
svr	7.6	5.0
lio	5.9	3.9
h-naive	11.3	7.1
d-naive	17.3	13.3
w-naive	13.0	9.1

- Higher value of H:
  - Wider bounds than for the *sync* case
  - Power is sensitive to the price
  - Wider range of marginal utilities



- H values for *kio* and *lio* are quite similar
- *kio* and *lio* give rise to the least errors
- *kio* outperforms *h-naive*, *krr*, *svr*: 51.3% ↓, 25.7% ↓, 27.6% ↓

To wrap up...

- Data-driven two-step forecasting approach
    - Inverse optimization
    - kernel regression
- Unlike other techniques
    - Forecasting
    - Bidding
- It achieves a similar or better performance than other techniques
- Adjectives that best describe our model
    - Versatile
    - Interpretable

Contacts

Any questions?

Ricardo Fernández-Blanco
ricardo.fcarramolino@uma.es
OASYS Webpage: oasys.uma.es

Results with V2G Capabilities

- *sync* case:
  - *kio* outperforms better than *lio*
  - Similar performance of *kio* and other machine-learning techniques
- *non-sync* case:
  - *kio* predicts better than the rest of methods

Model	*sync*		*non-sync*	
	RMSE	MAE	RMSE	MAE
kio	148.6	94.3	33.5	20.9
krr	146.9	108.4	35.2	23.6
svr	147.1	92.4	35.6	22.4
lio	172.1	120.0	36.2	23.7
h-naive	235.4	142.2	49.5	30.0
d-naive	261.8	162.5	71.1	50.2
w-naive	199.5	112.3	60.4	37.7

Results with V2G Capabilities

- *sync* case:
  - *kio* outperforms better than *lio*
  - Similar performance of *kio* and other machine-learning techniques
- *non-sync* case:
  - *kio* predicts better than the rest of methods

Model	*sync*		*non-sync*	
	RMSE	MAE	RMSE	MAE
kio	148.6	94.3	33.5	20.9
krr	146.9	108.4	35.2	23.6
svr	147.1	92.4	35.6	22.4
lio	172.1	120.0	36.2	23.7
h-naive	235.4	142.2	49.5	30.0
d-naive	261.8	162.5	71.1	50.2
w-naive	199.5	112.3	60.4	37.7

Results with V2G Capabilities

- *sync* case:
  - *kio* outperforms better than *lio*
  - Similar performance of *kio* and other machine-learning techniques
- *non-sync* case:
  - *kio* predicts better than the rest of methods

Model	*sync*		*non-sync*	
	RMSE	MAE	RMSE	MAE
kio	148.6	94.3	33.5	20.9
krr	146.9	108.4	35.2	23.6
svr	147.1	92.4	35.6	22.4
lio	172.1	120.0	36.2	23.7
h-naive	235.4	142.2	49.5	30.0
d-naive	261.8	162.5	71.1	50.2
w-naive	199.5	112.3	60.4	37.7

Results with V2G Capabilities

- *sync* case:
  - *kio* outperforms better than *lio*
  - Similar performance of *kio* and other machine-learning techniques
- *non-sync* case:
  - *kio* predicts better than the rest of methods

Model	*sync*		*non-sync*	
	RMSE	MAE	RMSE	MAE
kio	148.6	94.3	33.5	20.9
krr	146.9	108.4	35.2	23.6
svr	147.1	92.4	35.6	22.4
lio	172.1	120.0	36.2	23.7
h-naive	235.4	142.2	49.5	30.0
d-naive	261.8	162.5	71.1	50.2
w-naive	199.5	112.3	60.4	37.7