Robust Gas Demand Forecasting With Conformal Prediction

Mouhcine Mendil, Luca Mossina, Marc Nahban, Kevin Passini

August 26, 2022

Confiance.ai: Tackling the challenges of AI industrialization

- **Confiance.ai**: French program to design and industrialize trustworthy AI critical systems.
- 4 years project - 45M€ budget
- Academic and industrial entities in the fields of defense, transport, manufacturing industry and energy.

Source: www.peoplematters.in

- Hot topics: explainability, robustness and uncertainty quantification

Uncertainty quantification for Industry

- Meaningful and rigorous measures of uncertainty in predictions is important for industrial applications relying on AI systems:
  - Safety: failure or malfunction may result in life or severe material/environmental harm. E.g, transport and health.
  - Malfunctions may result in heavy capital or infrastructure loss. E.g, energy industry.
- Usually, ML models output:
  - A point prediction with no measure of uncertainty
  - Hardly interpretable uncertainty measures
- **We want** scalable and industrializable methods for uncertainty quantification !

Uncertainty quantification in Regression Tasks

- Prediction intervals (PIs) with respect to a significance level α:
  - Restrict the frequency of errors that the algorithm is allowed to make

- PIs are trivial in case of perfect knowledge of the data generating distribution $\mathbb{P}_{Y/X}$:
  - Upper and lower conditional quantiles (Fig.1)

- Quantile-based PIs achieve conditional coverage validity.

- Conditional coverage is **impossible in practice** when $\mathbb{P}_{Y/X}$ is unknown and arbitrary (without distributional assumptions) **(Balasubramanian et al. '14)(BaCaRaTi '19)**

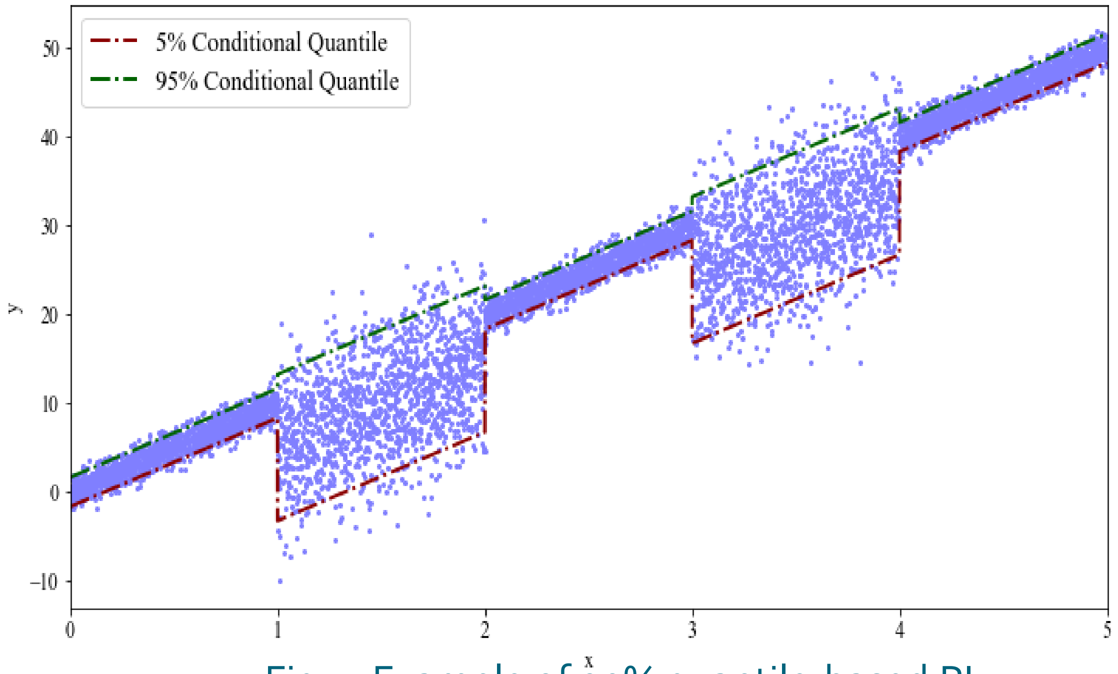


Fig.1- Example of 90% quantile-based PI

Conformal prediction for Uncertainty Quantification

Conformal prediction:

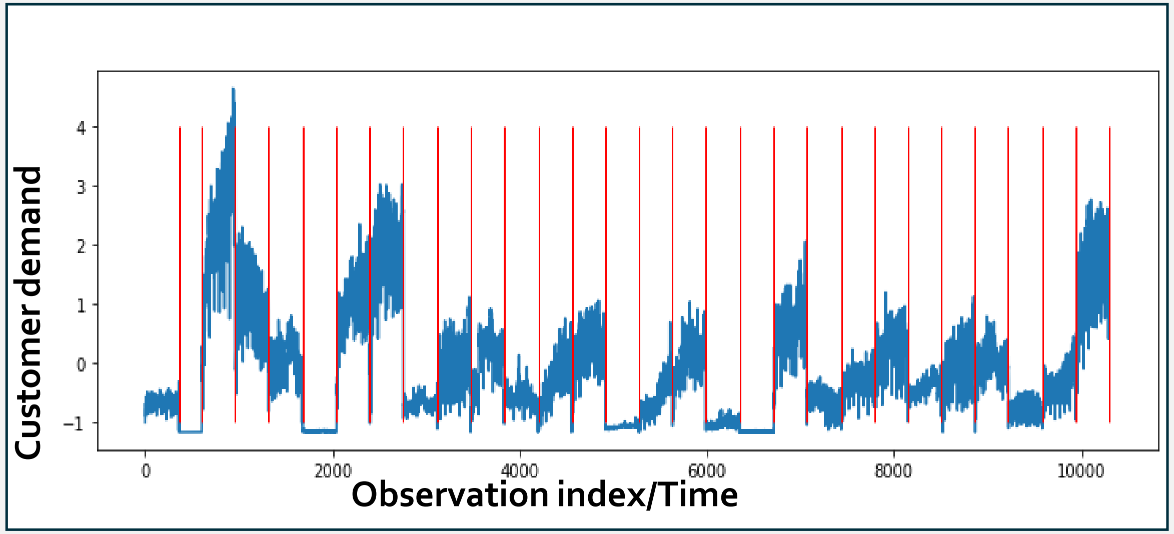
- ✓ Distribution-free, model agnostic and non-asymptotic methods with marginal coverage guarantee.
- ✓ In industrial environments, relevancy of conformal prediction for black-box models is a substantial asset
  - Low cost to exploit the existing and post-process for uncertainty quantification
- ✓ Marginal validity if exchangeable data
- Conditional coverage depends on underlying model and nonconfomity measure

➢ **Our feedback on using conformal prediction for industrial gas distribution in France**

Use case: Industrial Gas Distribution By Air Liquide

CONTEXT

- The production units should guarantee that all customers are always in supply
- Precisely predicting the future customer demands highly critical for the production sites



- Balance between gas supply (production sites) and demand (customers)
- Need for good estimation of future customers demand
- The dispatchers make an educated guess about future trends.

Air Liquide

Use case: Industrial Gas Distribution By Air Liquide

CONTEXT

- The production units should guarantee that all customers are always in supply
- Precisely predicting the future customer demands highly critical for the production sites

INFERENCE ENGINE (DEPLOYED SOLUTION)

→ XGBoost Regression – Time series forecast model

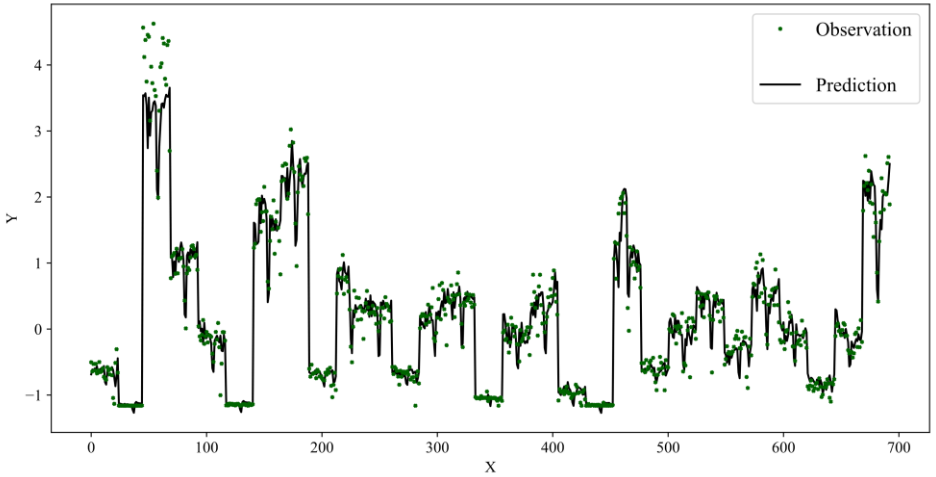


Fig.2- Prediction of customers' gas demand for the following week(s)

Use case: Industrial Gas Distribution By Air Liquide

CONTEXT

- The production units should guarantee that all customers are always in supply
- Precisely predicting the future customer demands highly critical for the production sites

INFERENCE ENGINE (DEPLOYED SOLUTION)

→ XGBoost Regression – Time series forecast model

OBJECTIVES

Valid and **efficient prediction Intervals** that quantify uncertainty in the forecast

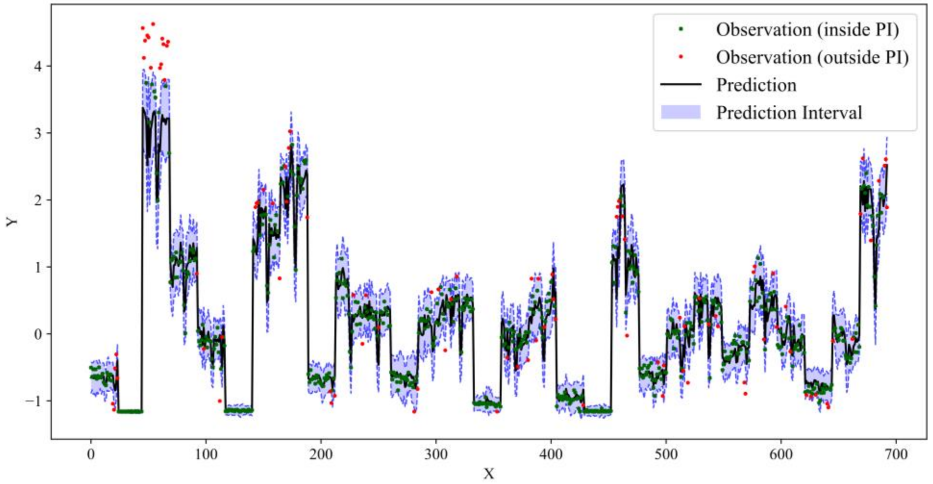


Fig.3- Prediction Intervals of customers' gas demand for the following week(s)

Methodology and Experiments

- **The validity of conformal prediction relies on the assumption of data exchangeability**

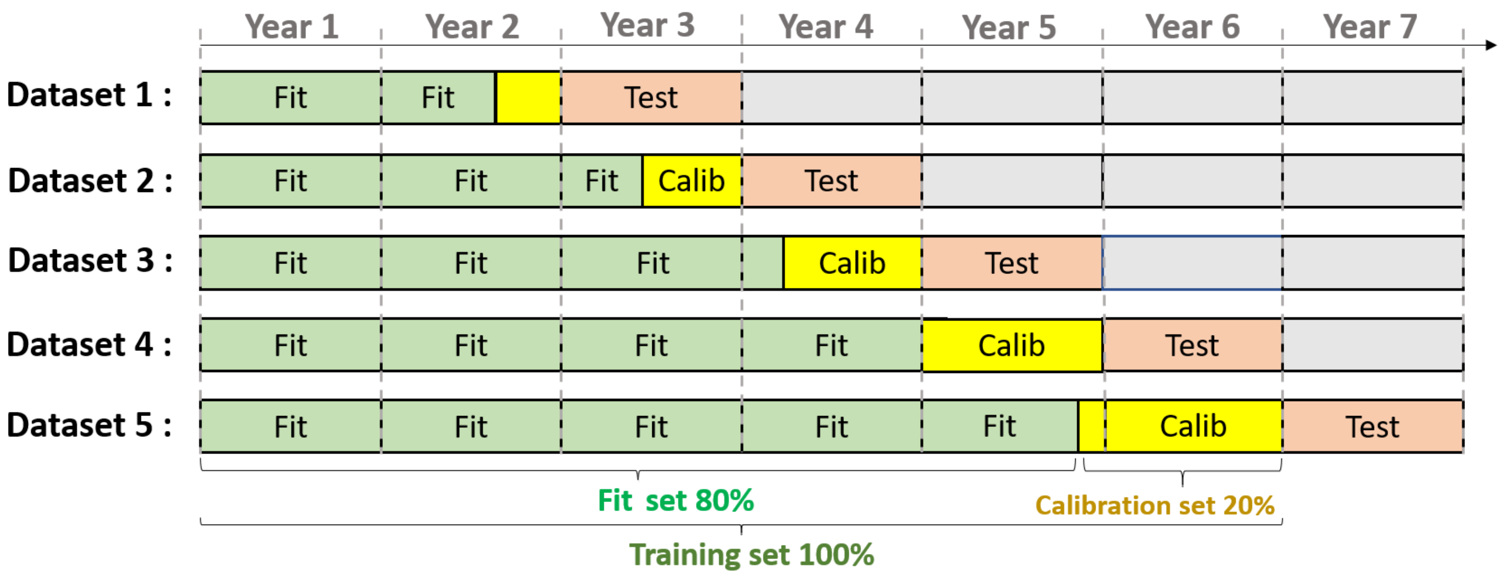
- **Time series are not necessarily exchangeable (distribution shift in case of customers gas demand)**

- **Straightforward benchmark of SotA methods regardless of data exchangeability:**
  - Does conformal prediction improve uncertainty estimation ?
  - Are they systematically better (more valid/efficient) CP methods than others ?
  - How difficult is it implement/deploy conformal prediction in operations ?

Methodology and Experiments

- **We considered inductive conformal prediction:**
  - Training set is split into fit and calibration subsets with a 80%-20% ratio (Sesia and Candès, 2020)
  - One year worth of test data

- **Sequential cross-validation scheme with five datasets (for results robustness)**



Methodology and Experiments

- **Four time series forecast models:**
  - eXtreme Gradient Boosting (XGB)
  - Gradient Boosting Quantile Regression (GBQR)
  - Quantile Random Forests (QRF)
  - Random Forest Mean Variance(RFMV)

Predictor	Conformalization					
	without[1]	SCP[2]	EnbPI[1]	CQR[2]	LACP[2]	aEnbPI[1]
XGB		✓	✓			
GBQR	✓			✓		
QRF	✓			✓		
RFMV	✓				✓	✓

Table 1: Summary of the methods evaluated in the experiments.
[1]Approaches without conformalization are trained on 100% of the training set. [2]Sequential cross-validation CP procedure; 80% (resp. 20%) of the train data are assigned to the fit (resp. calibration) set (see Figure 2).

- **Five calibration methods:**
  - Split Conformal Prediction (SCP) (H. Papadopoulos et al. 2002)
  - Locally Adaptive Conformal Prediction (LACP) (H. Papadopoulos et al. 2008) (J. Lei et al. 2015)
  - Conformalized Quantile Regression (CQR) (Y. Romano et al. 2019)
  - Ensemble batch Prediction Interval (EnbPI) (C. Xu et al. 2021)
  - Adaptive Ensemble batch Prediction interval (aEnbPI)

Inductive CP methods: Reminder

- **General steps of Inductive CP:**
  1. Choose (receive) estimator(s) $\hat{f}$
  2. Choose nonconformity score $R = s\left(\hat{f}(x), y\right)$
  3. Choose data scheme $\{D_{fit}, D_{calibration}\}$
  4. Fit and calibrate: fit $\hat{f}$ on D_{fit} and compute scores $\bar{R} = \{R_i\}, i = 1, \dots, |D_{calibration}|$ on $D_{calibration}$
  5. Get error margin $\delta_\alpha = (1-\alpha)(1 + \frac{1}{|D_{calibration}|})$-th empirical quantile of $\bar{R}$
  6. Inference: build CP interval $\widehat{C_\alpha}(X_{new})$ for new example X_{new}

	Split CP	Locally adaptive CP	CQR
Estimators	$\hat{f}$: conditional mean $\mathbb{E}\,(Y\|X)$	$(\hat{f}, \hat{\sigma})$: conditional mean $\mathbb{E}\,(Y\|X)$ and conditional mean absolute deviation	$(\hat{q}_{\alpha_{lo}}, \hat{q}_{1-\alpha_{hi}})$: α_{lo}-th and $1-\alpha_{hi}$-th quantiles
Nonconformity score	$R_i = \|\hat{f}(x_i) - y_i\|$	$R_i = \frac{\|\hat{f}(x_i) - y_i\|}{\hat{\sigma}(x_i)}$	$R_i = \max\{\hat{q}_{\alpha_{lo}}(x_i) - y_i, y_i - \hat{q}_{1-\alpha_{hi}}(x_i)\}$
Prediction interval	$\widehat{C_\alpha}(x) = [\hat{f}(x) - \delta_\alpha, \hat{f}(x) + \delta_\alpha]$	$\widehat{C_\alpha}(x) = [\hat{f}(x) - \hat{\sigma}(x) \cdot \delta_\alpha, \hat{f}(x) + \hat{\sigma}(x) \cdot \hat{\sigma}(x)\delta_\alpha]$	$\widehat{C_\alpha}(x) = [\hat{q}_{\alpha_{lo}}(x) - \delta_\alpha, \hat{q}_{1-\alpha_{hi}}(x) - \delta_\alpha]$

Adaptive EnbPI

- EnbPI: modification of Jackknife+-after-Bootstrap using out-of-bag trick of (Breiman 1996) to estimate leave-one-out (LOO) nonconformity scores and aggregated predictors.

- Prediction errors should be « well-behaved » (strongly mixing or even i.i.d)

- Adaptive EnbPI: extension of the locally adaptive conformal prediction to EnbPI:
    - LOO estimates of conditional mean absolute deviation (MAD)
    - Residuals and prediction intervales are scaled w.r.t the conditional MAD

Algorithm 1: *Adaptive Ensemble batch Prediction Interval* (aEnbPI). Additions or modifications of EnbPI (Xu and Xie, 2021b) are highlighted.

Input: Training data $\{(x_i, y_i)\}_{i=1}^{T}$, point prediction algorithm A^f, variability prediction algorithm A^σ, miscoverage level α, aggregation function ϕ, number of bootstrap models B, batch size s, and test data $\{(x_t, y_t)\}_{t=T+1}^{T+T_1}$; y_t is revealed only after the batch of s PIs with t in the batch are constructed.

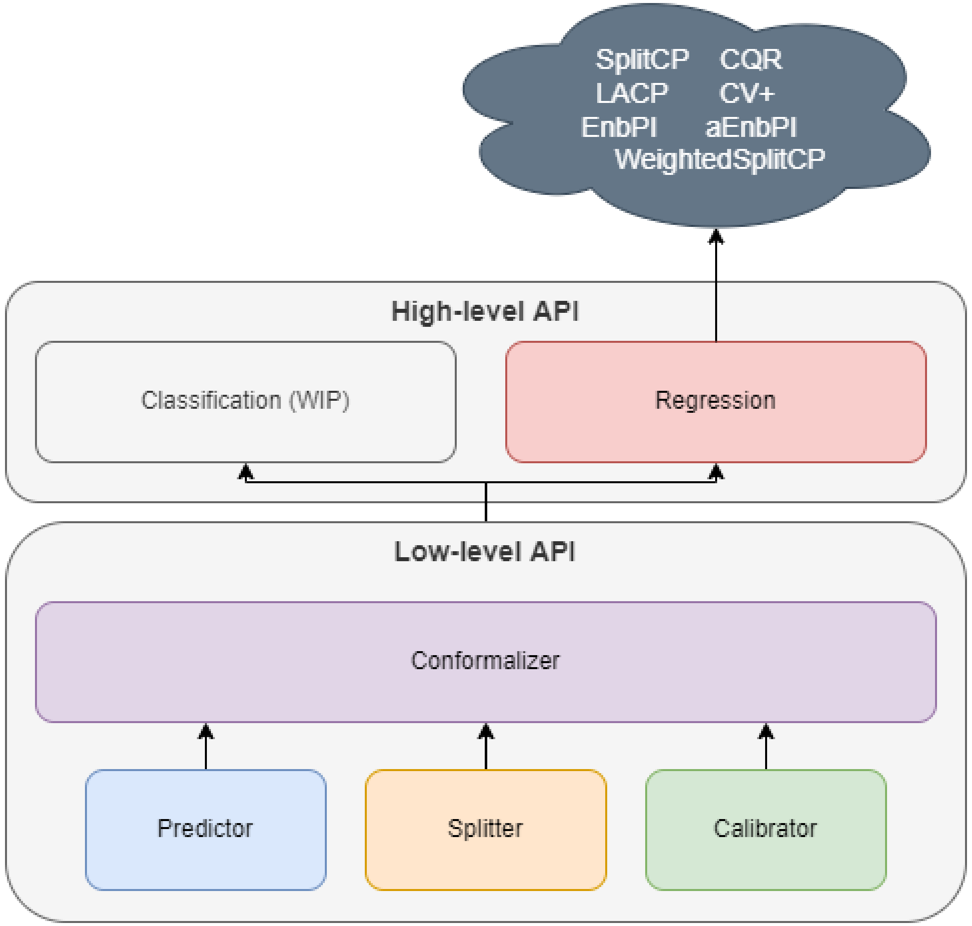
for $b = 1, \dots, B$ do
  Sample with replacement an index set $S_b = (i_1, \dots, i_T)$ from indices $(1, \dots, T)$;
  Compute $\hat{f}^b \leftarrow A^f(\{(x_i, y_i) | i \in S_b\})$;
  Compute $\hat{\sigma}^b \leftarrow A^\sigma(\{(x_i, y_i) | i \in S_b\})$;
end
$\hat{\epsilon} \leftarrow \{\}$;
for $i = 1, \dots, T$ do
  $\hat{f}^\phi_{-i}(x_i) \leftarrow \phi(\{\hat{f}^b(x_i) | i \notin S_b\})$;
  $\hat{\sigma}^\phi_{-i}(x_i) \leftarrow \phi(\{\hat{\sigma}^b(x_i) | i \notin S_b\})$;
  $\hat{\epsilon}^\phi_i \leftarrow \frac{|y_i - \hat{f}^\phi_{-i}(x_i)|}{\hat{\sigma}^\phi_{-i}(x_i)}$;
  $\hat{\epsilon} \leftarrow \hat{\epsilon} \cup \{\hat{\epsilon}^\phi_i\}$;
end
$\hat{C} \leftarrow \{\}$;
for $t = T+1, \dots, T+T_1$ do
  $\hat{f}^\phi_{-t}(x_t) \leftarrow (1-\alpha)$ quantile of $\{\hat{f}^\phi_{-i}(x_t)\}_{i=1}^{T}$;
  $\hat{\sigma}^\phi_{-t}(x_t) \leftarrow \phi\{\hat{\sigma}^\phi_{-i}(x_t)\}_{i=1}^{T}$;
  $w^\phi_{T,t} \leftarrow (1-\alpha)$ quantile of ϵ;
  $C^{\phi,\alpha}_{T,t}(x_t) \leftarrow [\hat{f}^\phi_{-t}(x_t) \pm w^\phi_t\, \hat{\sigma}^\phi_{-t}(x_t)]$;
  $\hat{C} \leftarrow \hat{C} \cup C^{\phi,\alpha}_{T,t}(x_t)$;
  if $t - T = 0 \bmod s$ then
    for $j = t-s, \dots, t-1$ do
      $\hat{\epsilon}^\phi_j \leftarrow \frac{|y_j - \hat{f}^\phi_{-j}(x_t)|}{\hat{\sigma}^\phi_{-j}(x_t)}$;
      $\hat{\epsilon} \leftarrow (\hat{\epsilon} - \{\hat{\epsilon}^\phi_1\}) \cup \{\hat{\epsilon}^\phi_j\}$ and reset index of $\hat{\epsilon}$;
    end
  end
end
return *Ensemble prediction intervals* $\hat{C} = \{C^{\phi,\alpha}_t(x_t)\}_{t=T+1}^{T+T_1}$

Predictive Uncertainty Calibration and Conformalization (PUNCC) Library

- Open source python library
- High-level API (Fast prototyping)
  - Preconfigured and ready-to-use conformal prediction wrappers

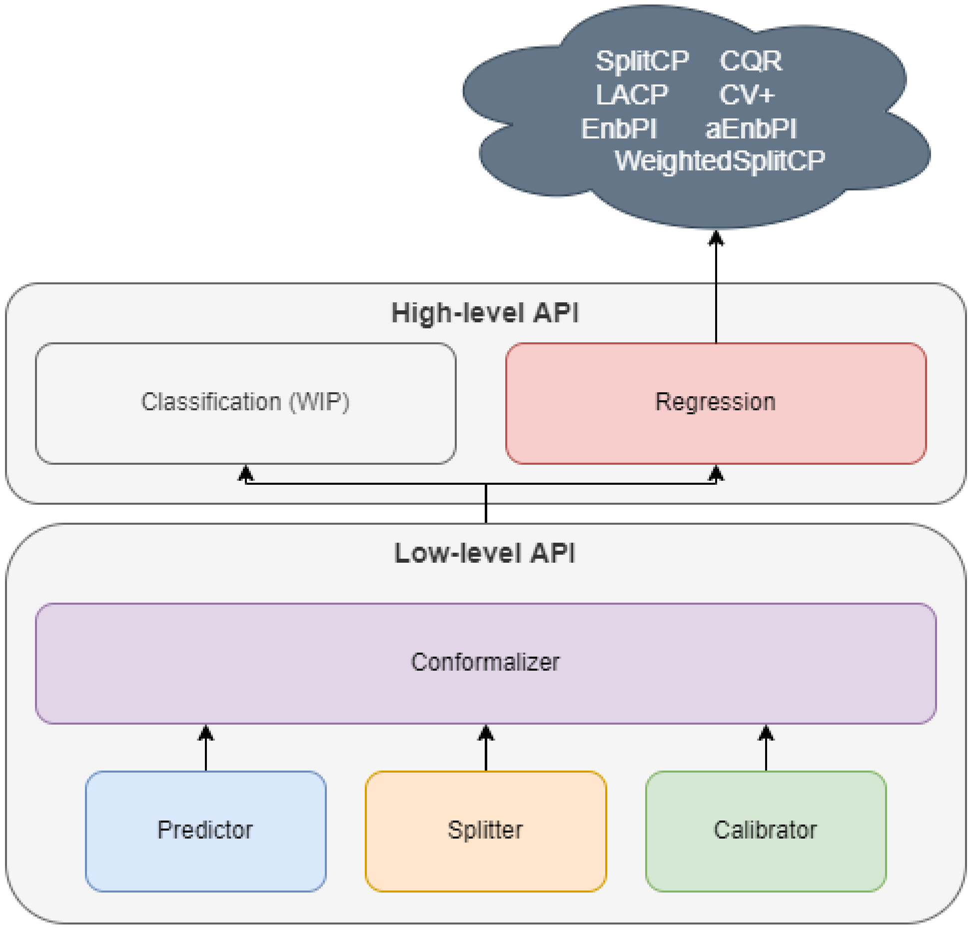
```
from deel.puncc.regression import SplitCP

# Coverage target is 1-alpha = 90%
alpha=.1
# Instanciate the split cp wrapper on the linear model
split_cp = SplitCP(regr)
# Train model on the fitting dataset and compute residuals on the calibration
# dataset
split_cp.fit(X_fit, y_fit, X_calib, y_calib)
# Estimate the prediction interval
y_pred, y_pred_lower, y_pred_upper = split_cp.predict(X_test, alpha=alpha)
```



Predictive Uncertainty Calibration and Conformalization (PUNCC) Library

- Open source python library
- Low-level API
  - Full customization of calibration based on three components:
    - Predictor: interface standardize for ML/DL models
    - Splitter: split data scheme (K-folds, random, …)
    - Calibrator: estimator of nonconformity scores and prediction intervals
  - Enable to design new conformal workflows:
    - E.g. conformalized cross-validation quantile reg



Predictive Uncertainty Calibration and Conformalization (PUNCC) Library

- Open source python library
- Low-level API
  - Full customization based on three components:
    - Predictor: interface standardize for ML/DL models
    - Splitter: split data scheme (K-folds, random, …)
    - Calibrator: estimator of nonconformity scores and prediction intervals
  - Enable to design new conformal workflows:
    - E.g. conformalized cross-validation quantile reg

```python
from deel.puncc.api import conformalization, prediction, calibration, splitting

# ...

## Quantile Predictors
predictor = prediction.QuantilePredictor(q_lo_model=q_lo_model,
                                         q_hi_model=q_hi_model,
                                         is_trained=False)
## CQR (A. Romano) Calibrator
calibrator = calibration.QuantileCalibrator()
## KFold Splitter
splitter = KFoldSplitter(K=K, random_state=random_state)
## Init Conformal prediction (CV+/CQR)
conformalizer = conformalization.ConformalPredictor(predictor=predictor,
                                                    calibrator=calibrator,
                                                    splitter=splitter,
                                                    method="cv+")
# The fit method trains the model and computes the residuals on the
# calibration set
conformalizer.fit()
# The predict method infers prediction intervals with respect to
# the risk alpha
_, y_pred_lower, y_pred_upper, _ = conformalizer.predict(X_test, alpha=alpha)
```

Metrics

- **Let n be the number of samples in a test dataset $D_{test} = \{(x_i, y_i)\}_{i=1}^{n}$ and L the range of labels. Two metrics are considered:**

  - Prediction Interval Coverage Probability (PICP) => empirical coverage

$$\text{PICP} = \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}_{\{y_i \in \hat{C}_\alpha(x_i)\}}$$

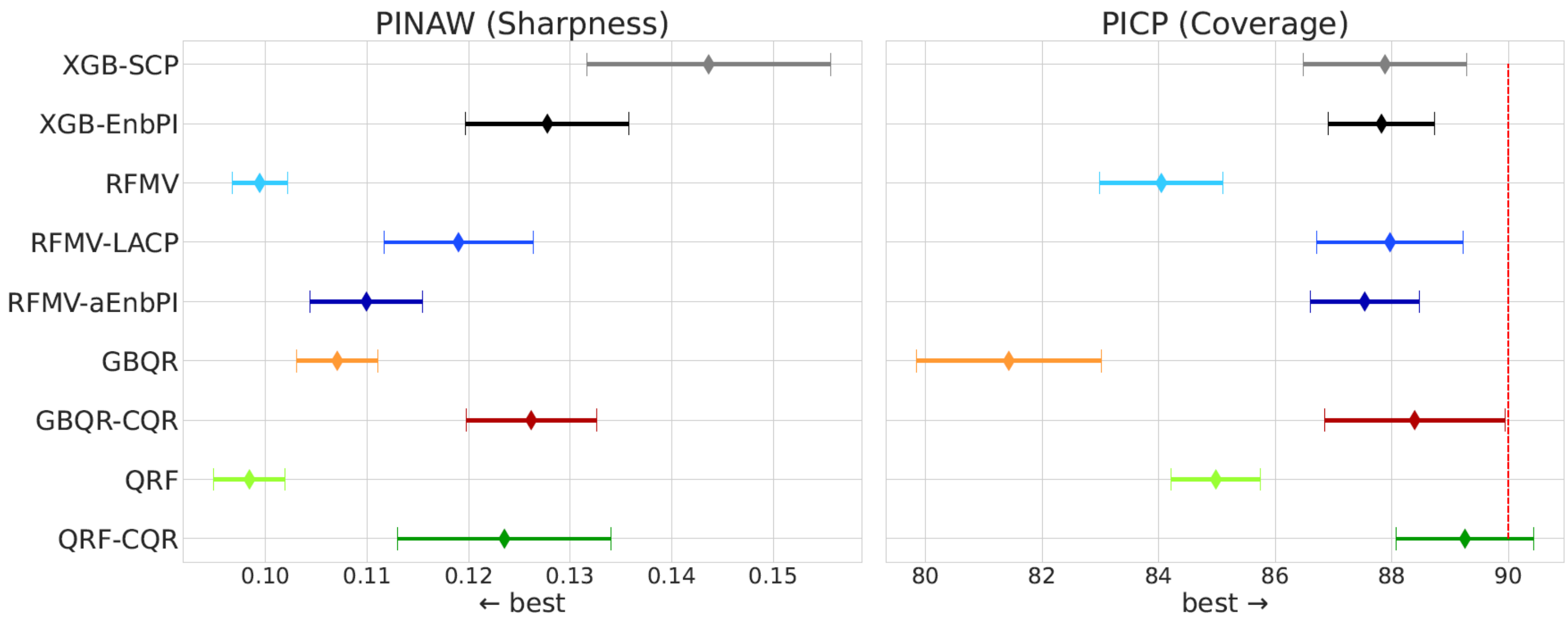
  - Prediction Interval Normalized Average Width (PINAW):

$$\text{PINAW} = \frac{1}{n \cdot L}\sum_{i=1}^{n} \text{len}\left(\hat{C}_\alpha(x_i)\right)$$

Results: all you need is conformal !

- The target significance level $\alpha = 10\%$; Results are averaged over the five datasets

Predictor	CP approach	PINAW (*sharpness*)	PICP [%] (*coverage*)
XGB	SCP	0.144 ± 0.024	87.88 ± 2.86
XGB	EnbPI	0.128 ± 0.016	87.82 ± 1.82
RFMV	-	0.099 ± 0.005	84.05 ± 2.12
RFMV	LACP	0.119 ± 0.015	87.97 ± 2.51
RFMV	aEnbPI	0.110 ± 0.011	87.54 ± 1.86
GBQR	-	0.107 ± 0.008	81.43 ± 3.17
GBQR	CQR	0.126 ± 0.013	88.39 ± 3.10
QRF	-	0.098 ± 0.007	84.98 ± 1.53
QRF	CQR	0.124 ± 0.021	89.26 ± 2.35



- Most CP methods are nearly valid for our timeseries !
- CP improves uncertainty quantification for point and interval estimators
- Conformalized quantile regression: simple yet effective

Conclusions

- Conformal prediction is a lightweight post-processing set of methods that builds prediction intervals with theoretical coverage guarantee

- Some CP methods are relatively simple to adopt (and deploy) in industry

- CP can improve other uncertainty quantification methods (no competition ?)

- We recommend CQR on time series as a starting point (to be validated empirically on in-house data)

- Ongoing field testing phase (Air Liquide gas distribution)

Thank you

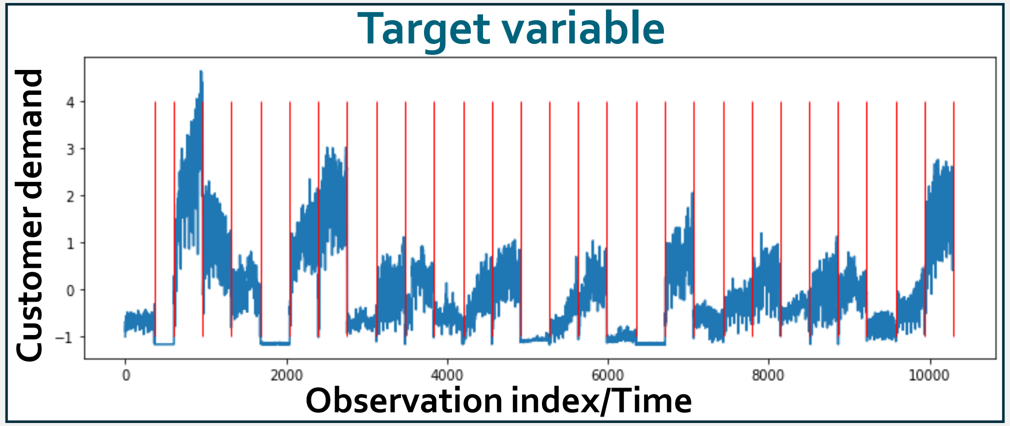
Use case: Industrial Gas Distribution By Air Liquide

CONTEXT

- The production units should guarantee that all customers are always in supply
- Precisely predicting the future customer demands highly critical for the production sites

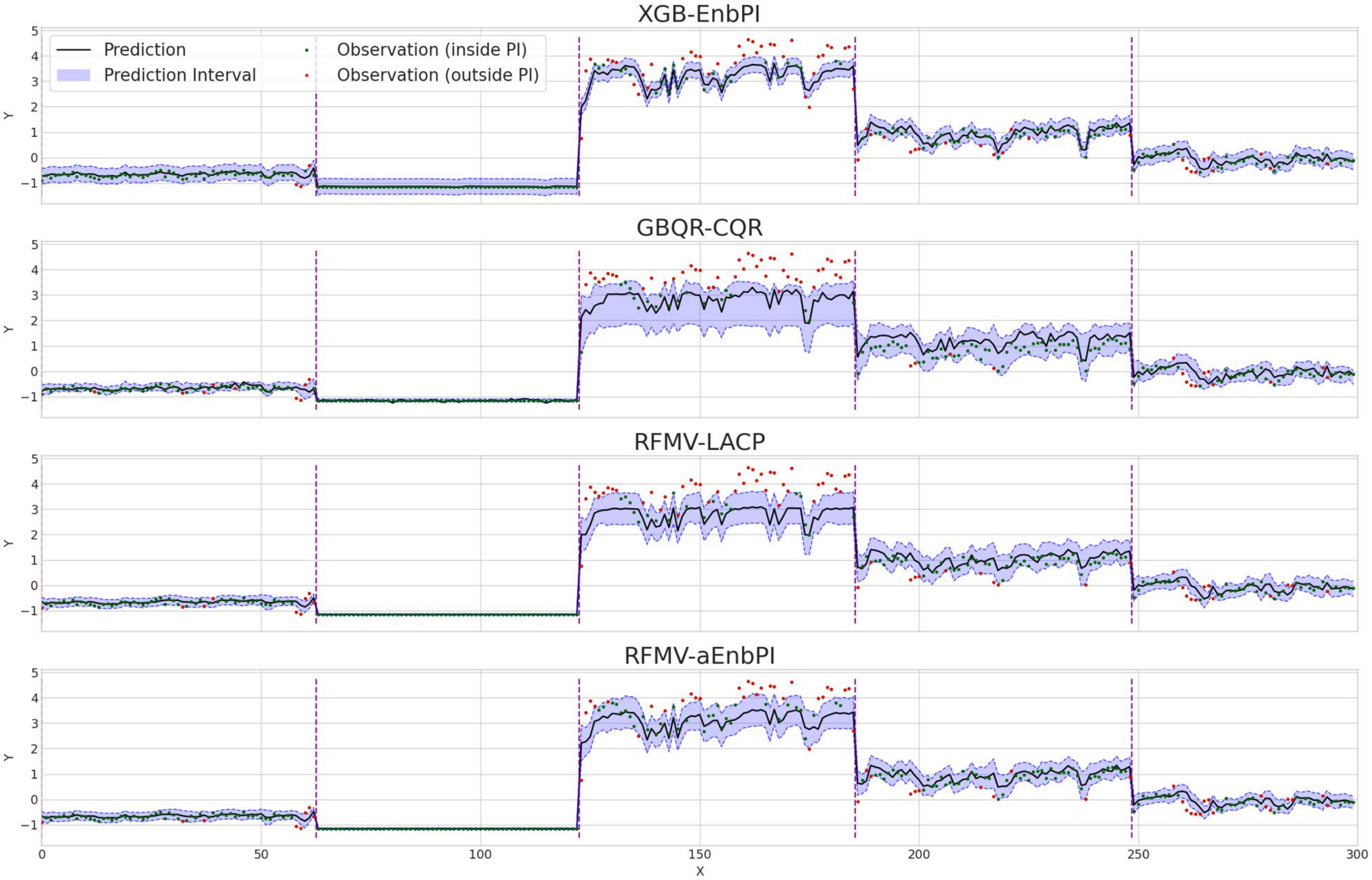
DATA (ANONYMIZED)

Historical consumption, geographical distribution, customers info, orders, contextual data, seasonality

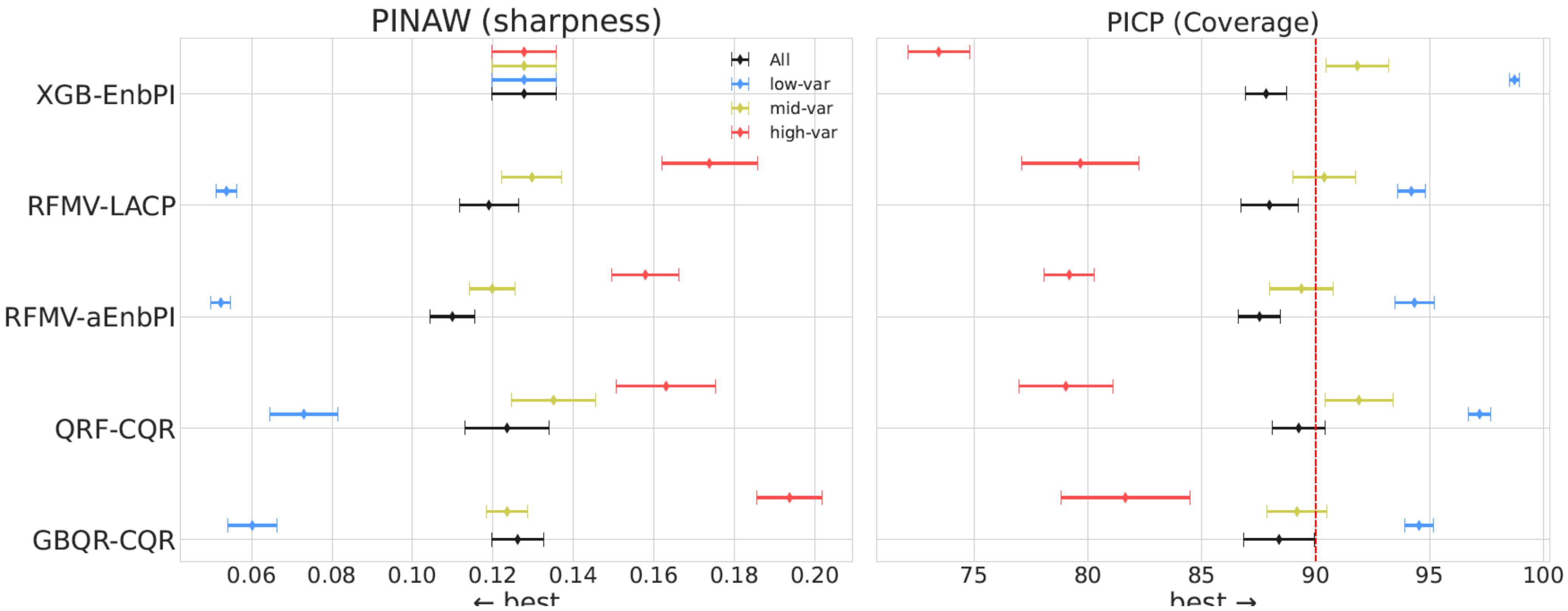


- Real data: gas products, customers and their demand
- 7 years of weekly data points
- Anonymization and transformations of sensitive data for confidentiality
- Historical and exogeneous data can influence the customers demand

Results


XGB-EnbPI
GBQR-CQR
RFMV-LACP
RFMV-aEnbPI
Prediction
Prediction Interval
Observation (inside PI)
Observation (outside PI)
Y
X
0
50
100
150
200
250
300
5
4
3
2
1
0
−1

Results



PINAW (sharpness)
PICP (Coverage)
All
low-var
mid-var
high-var
XGB-EnbPI
RFMV-LACP
RFMV-aEnbPI
QRF-CQR
GBQR-CQR
0.06
0.08
0.10
0.12
0.14
0.16
0.18
0.20
← best
75
80
85
90
95
100
best →