Uncertainty Quantification for Machine Learning algorithms
An introduction to Conformal Prediction

Claire Boyer Margaux Zaffran

25-04-2023

MASPIN Days at FEMTO-ST

Machine learning context

Quantile Regression

Split Conformal Prediction (SCP)

Jackknife/cross-val

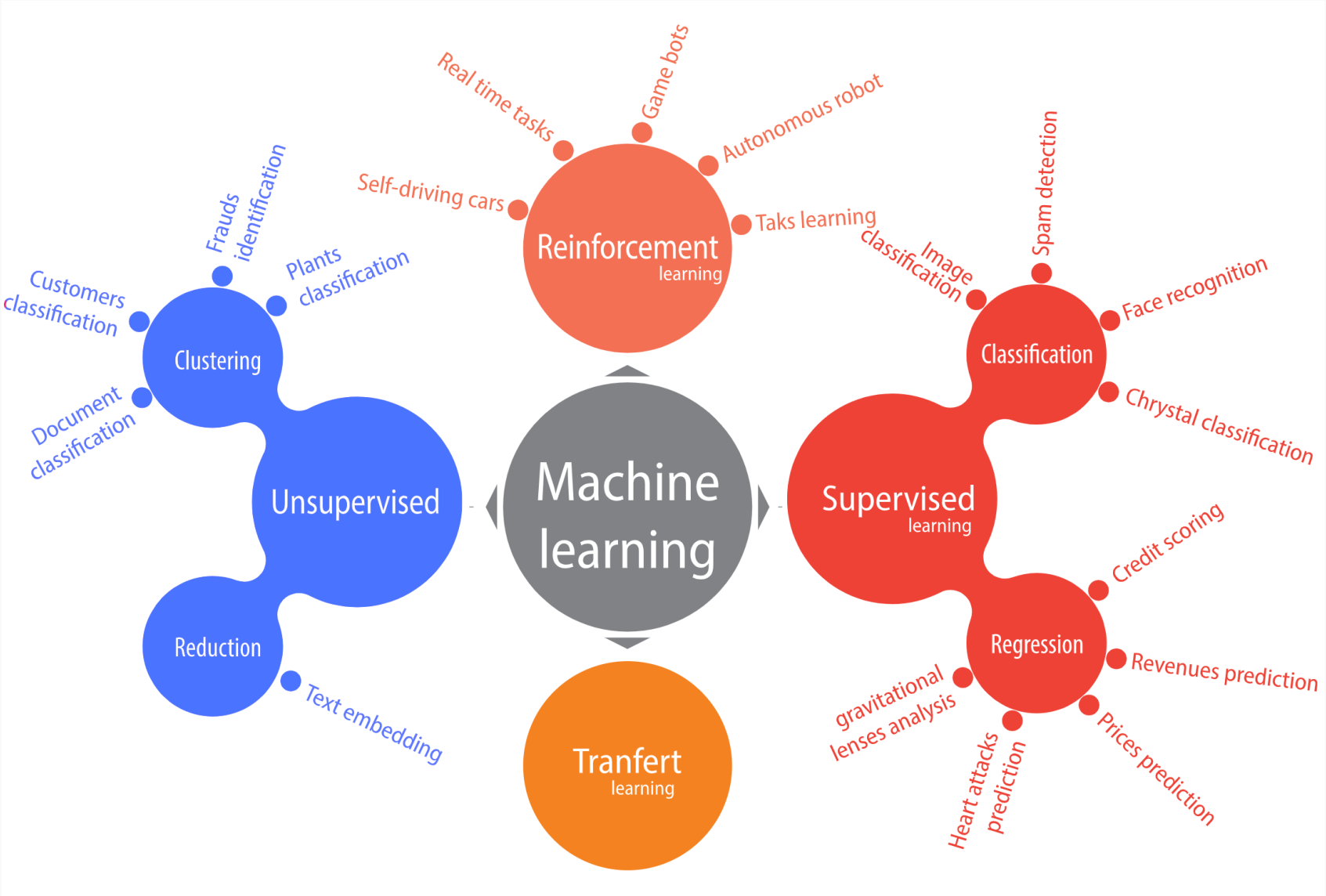
Beyond exchangeability

Learning scenarios

ML develops generic methods for solving different types of problems:

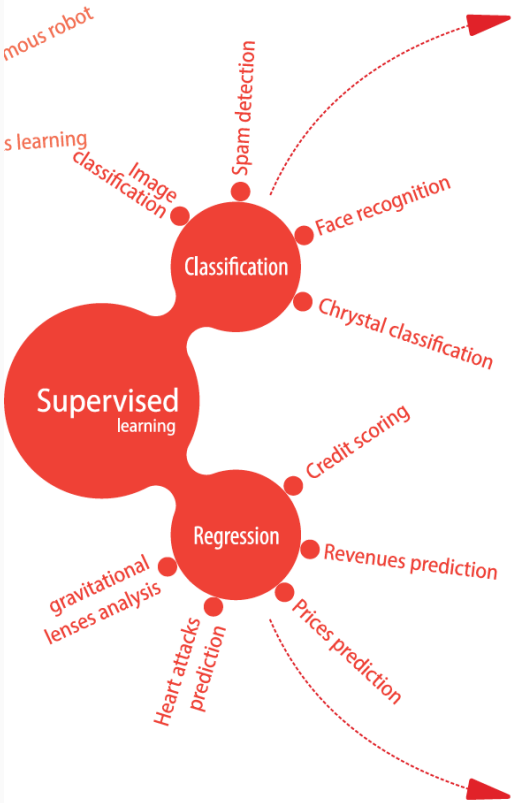
- Supervised learning
  Goal: learn from examples
- Unsupervised learning
  Goal: learn from data alone, extract structure in the data
- Reinforcement learning
  Goal: learn by exploring the environment (e.g. games or autonomous vehicle)

Learning scenarios



source: fidle-cnrs

Supervised learning



Classification :

Predict qualitative informations

This is a cat

This is a rabbit

Tell me, what is it ?

Régression :

Predict quantitative informations

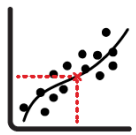
150 K€

400 K€

120 K€

100 K€

Tell me, what's the price ?



source: fidle-cnrs

Supervised learning, more formally

- Supervised learning: given a training sample $(X_i, Y_i)_{1 \leq i \leq n}$, the goal is to "learn" a predictor f_n such that

$$\underbrace{f_n(X_i) \simeq Y_i}_{\text{prediction on training data}} \qquad \text{and above all} \qquad \underbrace{f_n(X_{\text{new}}) \simeq Y_{\text{new}}}_{\text{prediction on test (unseen) data}}$$

- The nature of the output determines the type of supervised learning task
  - (classification) $X \in \mathbb{R}^d$ and $Y \in \{-1, 1\}$
  - (regression) $X \in \mathbb{R}^d$ and $Y \in \mathbb{R}$

How to measure the performance of a predictor?

- Loss function in general: $\ell(Y, f(X))$ measures the goodness of the prediction of Y by $f(X)$
- Examples:
  - (classification) Prediction loss: $\ell(Y, f(X)) = \mathbf{1}_{Y \neq f(X)}$
  - (regression) Quadratic loss: $\ell(Y, f(X)) = |Y - f(X)|^2$
- The performance of a predictor f in regression is usually measured through the risk

$$\text{Risk}_\ell(f) = \mathbb{E}\Big[\ell\big(Y_{\text{new}}, f(X_{\text{new}})\big)\Big]$$

- A minimizer $f^\star$ of the risk is called a Bayes predictor
  - (classification) $f^\star(X) = \text{argmax}_k \mathbb{P}(Y = k|X)$
  - (regression) $f^\star(X) = \mathbb{E}[Y|X]$

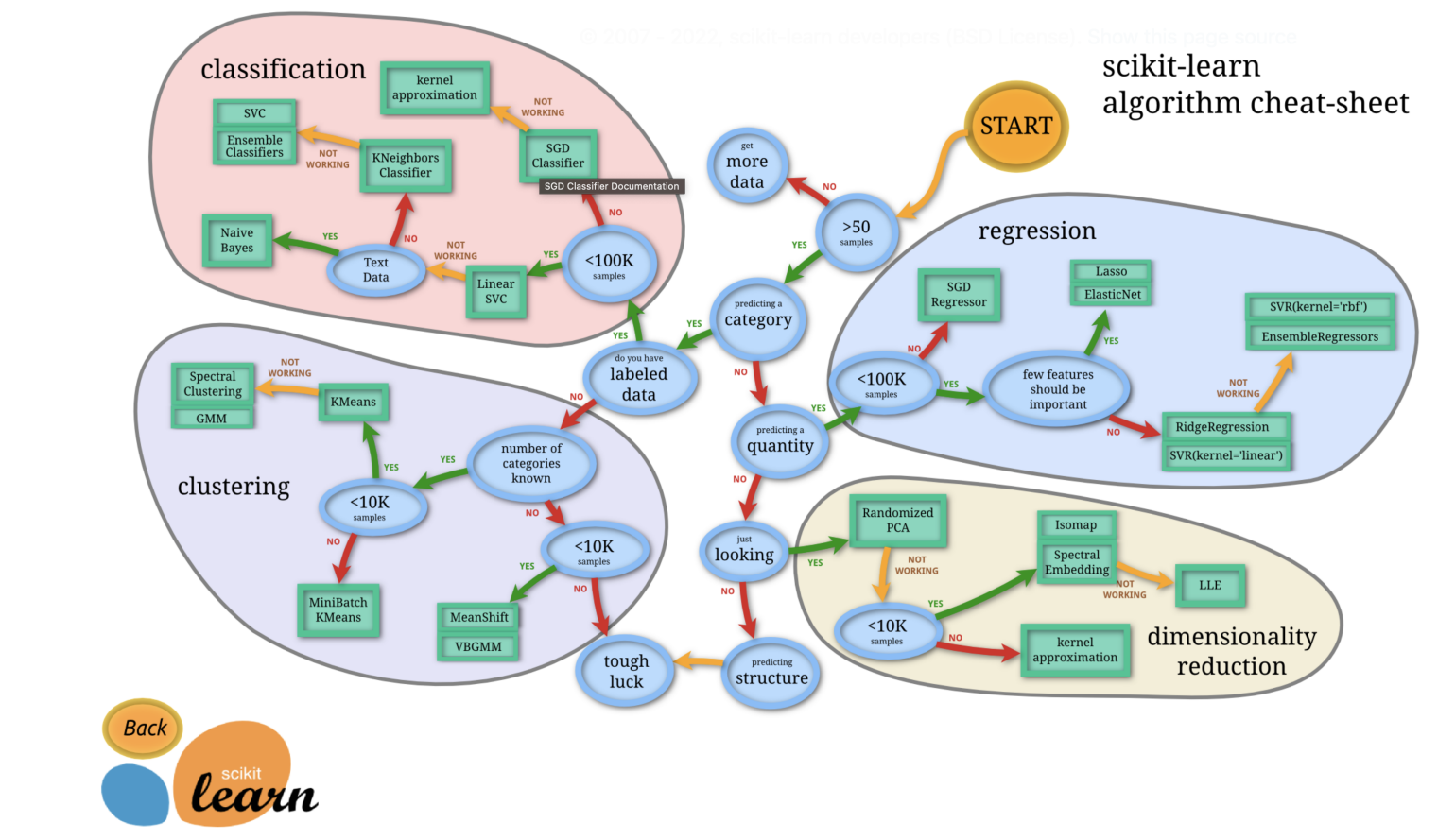
Learning by minimizing the empirical risk

- We want to construct a predictor with a small risk
- or an estimator of the Bayes predictor $f^\star$
- The distribution of the data is in general unknown, so is the risk
- Instead, given some training samples $(X_1, Y_1), \dots (X_n, Y_n)$, find the best predictor f that minimizes the empirical risk

$$\hat{\mathcal{R}}_n(f) := \frac{1}{n} \sum_{i=1}^{n} \ell(Y_i, f(X_i)).$$

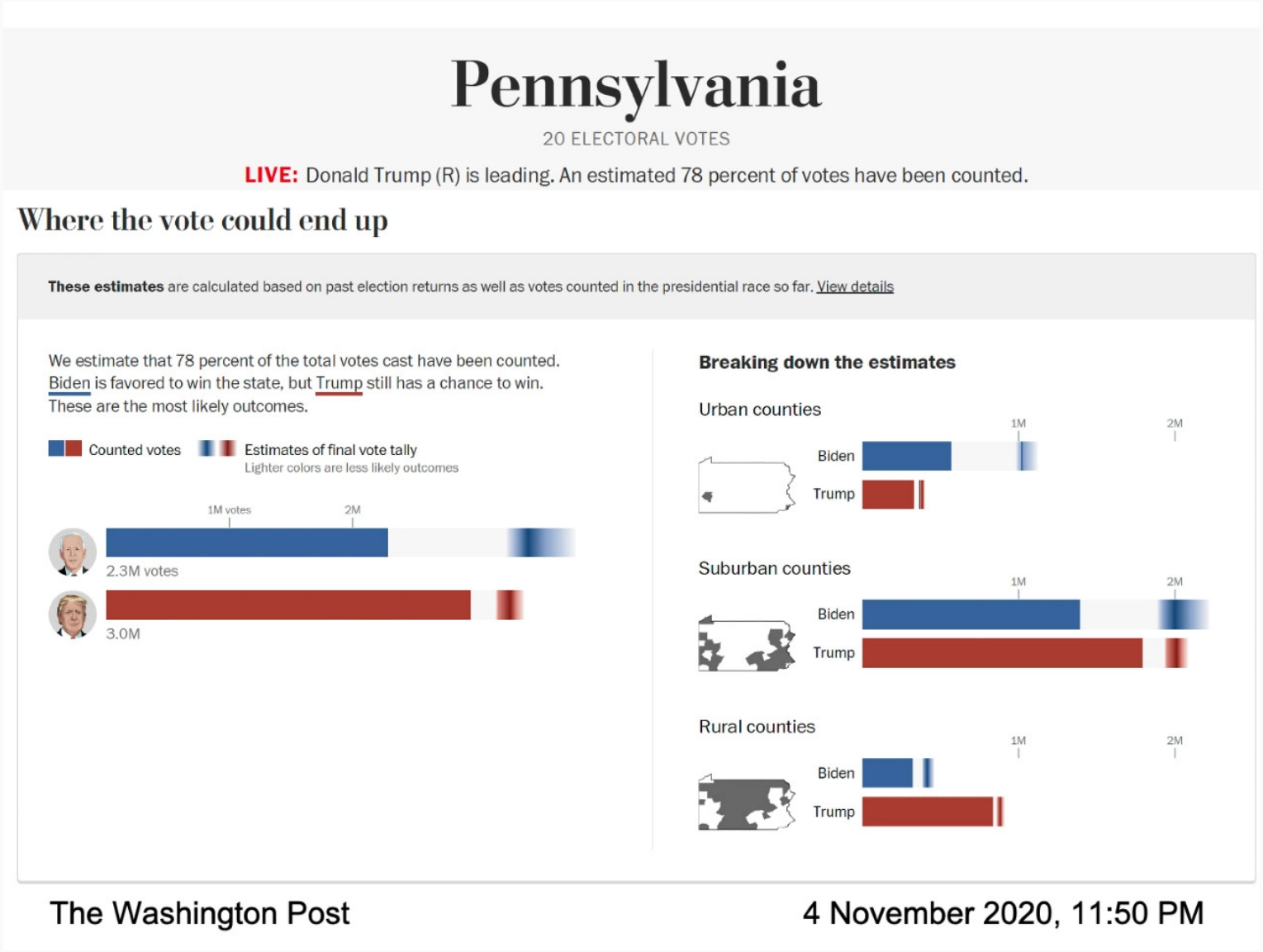
- Learning means retrieving information from training data by constructing a predictor that should have good performance on new data

There exist plenty of learners



see https://scikit-learn.org/stable/tutorial/machine_learning_map/index.html

On the importance of quantifying uncertainty



Machine learning context

Quantile Regression

Split Conformal Prediction (SCP)

Jackknife/cross-val

Beyond exchangeability

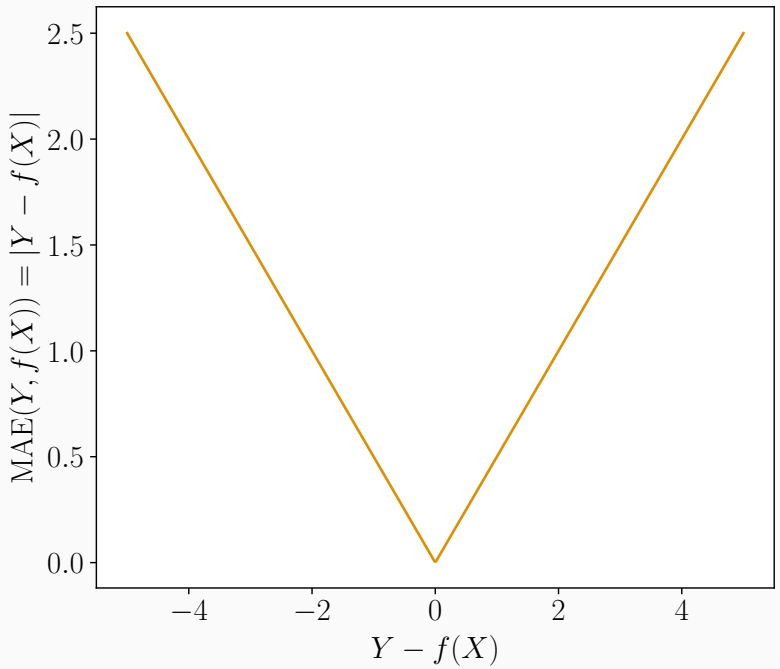
Reminder about quantiles

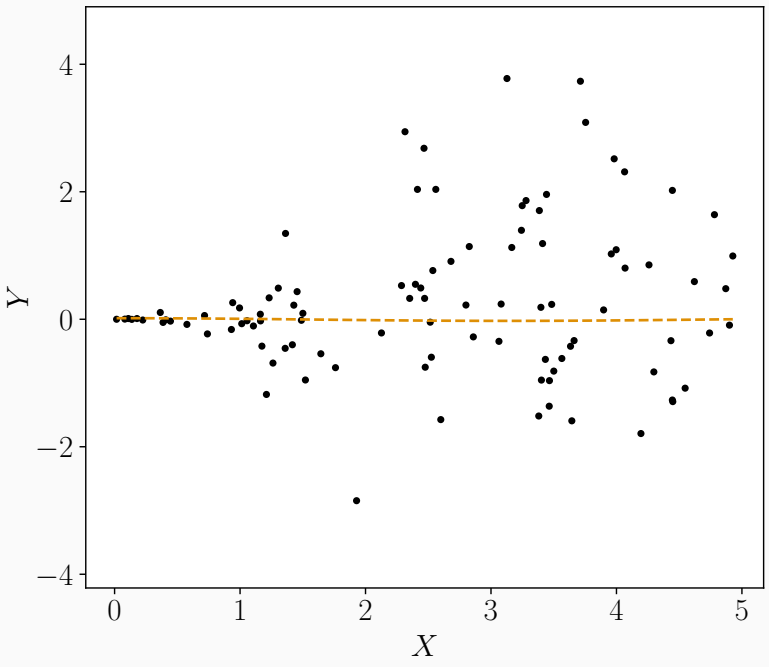
- Quantile level $\beta \in [0, 1]$
- $Q_X(\beta) = \inf\{x \in \mathbb{R}, \mathbb{P}(X \leq x) \geq \beta\} = \inf\{x \in \mathbb{R}, F_X(x) \geq \beta\}$
- $q_\beta(X_1, \ldots, X_n) = \lceil \beta \times n \rceil -$ smallest value of $(X_1, \ldots, X_n)$

Median regression

- The Bayes predictor depends on the chosen loss function
- Mean Absolute Error (MAE) $\ell(Y, Y') = |Y - Y'|$
- Associated risk $\text{Risk}_\ell(f) = \mathbb{E}\left[|Y - f(X)|\right]$
- Bayes predictor $f^\star \in \underset{f}{\text{argmin}}\, \text{Risk}_\ell(f)$

$$f^\star(X) = \text{median}\left[Y|X\right] = Q_{Y|X}(0.5)$$





Generalization: Quantile regression

- Quantile level $\beta \in [0, 1]$
- Pinball loss

$$\ell_\beta(Y, Y') = \beta|Y - Y'|\mathbb{1}_{\{|Y-Y'|\geq 0\}} + (1-\beta)|Y - Y'|\mathbb{1}_{\{|Y-Y'|\leq 0\}}$$

- Associated risk $\text{Risk}_{\ell_\beta}(f) = \mathbb{E}\left[\ell_\beta(Y, f(X))\right]$
- Bayes predictor $f^\star \in \underset{f}{\text{argmin}}\, \text{Risk}_{\ell_\beta}(f)$

$$f^\star(X) = Q_{Y|X}(\beta)$$

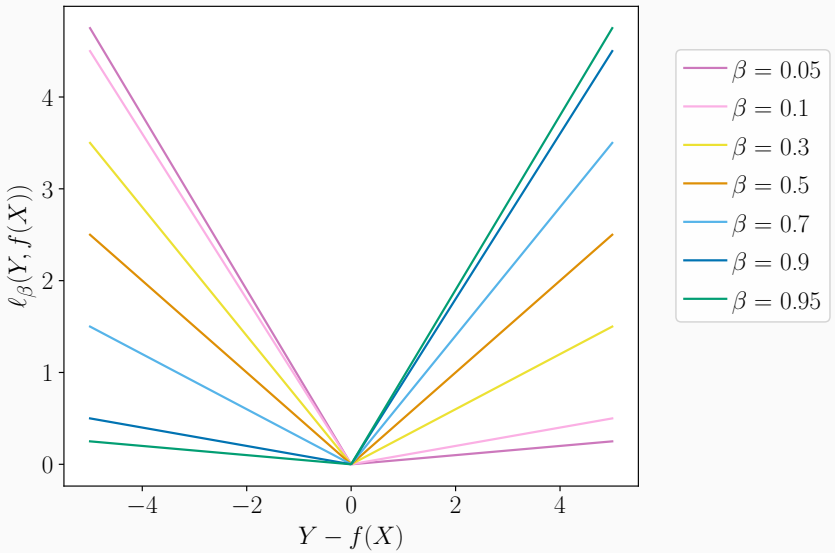


Figure 1: Pinball losses

Quantile regression

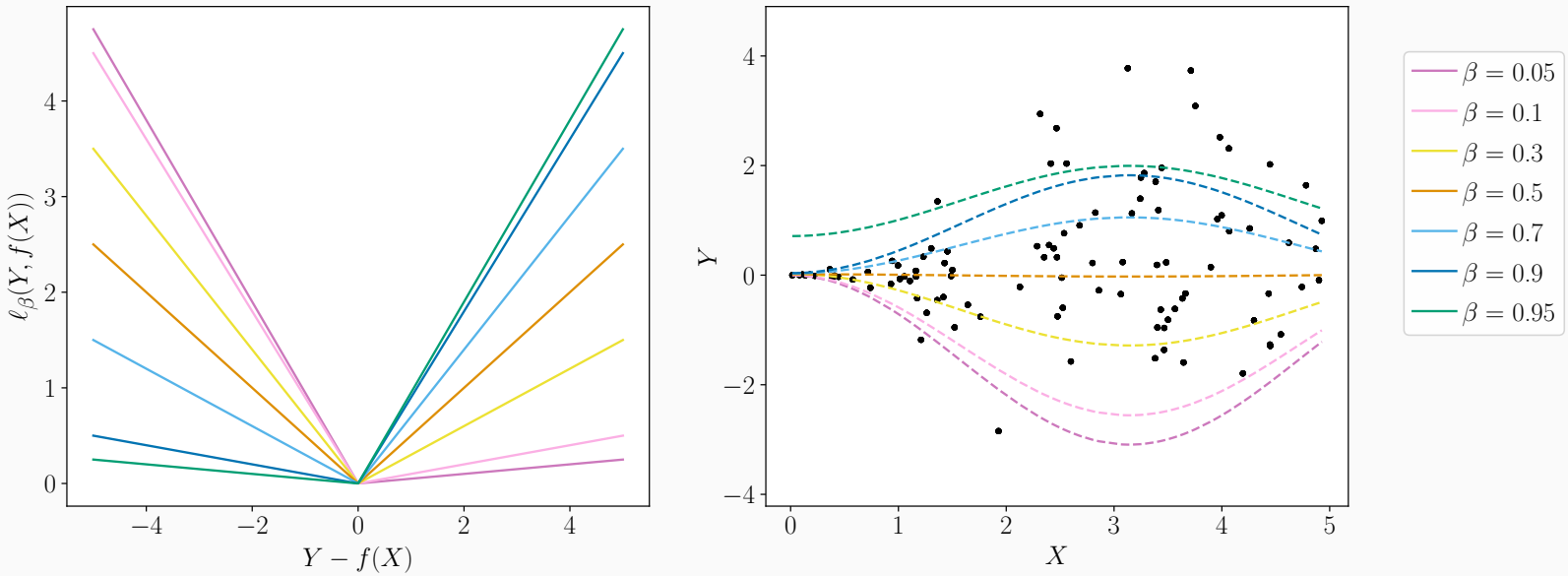
- Link between the pinball loss and the quantiles?
  Set $q^\star := \arg\min_q \mathbb{E}\left[\text{pinball}_\beta(Y-q)\right]$. Then,

$$
\begin{aligned}
0 &= \int_{-\infty}^{+\infty} \text{pinball}'_\beta(y-q) df_Y(y) \\
&= (\beta-1)\int_{-\infty}^{q} df_Y(y) + \beta\int_{q}^{+\infty} df_Y(y) \\
&= (\beta-1)F_Y(q) + \beta(1-F_Y(q))
\end{aligned}
$$

  which gives

$$\beta = F(q^\star) \iff q^\star = F^{-1}(\beta)$$

Quantile regression



Warning

No theoretical guarantee with a finite sample

$$\mathbb{P}\left(Y \in \left[\hat{Q}_{Y|X}(\beta/2); \hat{Q}_{Y|X}(1-\beta/2)\right]\right) \neq 1-\beta$$

Quantifying predictive uncertainty

- $(X, Y) \in \mathbb{R}^d \times \mathbb{R}$ random variables
- n training samples $(X_i, Y_i)_{i=1}^n$
- Goal: predict an unseen point Y_{n+1} at X_{n+1} with **confidence**
- How? Given a miscoverage level $\alpha \in [0, 1]$, build a predictive set $\mathcal{C}_\alpha$ such that:

$$\mathbb{P}\left\{Y_{n+1} \in \mathcal{C}_\alpha\left(X_{n+1}\right)\right\} \geq 1 - \alpha, \tag{1}$$

  and $\mathcal{C}_\alpha$ should be as small as possible, in order to be informative
- Construction of the predictive intervals should be
  - agnostic to the model
  - agnostic to the data distribution
  - valid in finite samples

Some historical landmarks

- 1996-1999: Emergence of Conformal Prediction (CP)
- Makers: Vladimir Vovk, Alexander Gammerman, Vladimir Vapnik, Glenn Shafer
- Popularized (2014+) by Jing Lei and Larry Wasserman
- Recently (2019+), real spotlight thanks to Rina Barber, Emmanuel Candès, Aaditya Ramdas and Ryan J. Tibshirani
- A good review can be found in Angelopoulos and Bates (2023)

SCP in regression

Algorithm

1. Split randomly your training data into a proper training set (size n_{train}) and a calibration set (size n_{cal})
2. Train your algorithm $\hat{A}$ on your proper training set
3. On the calibration set, get prediction values with $\hat{A}$
4. Obtain a set of $n_{\text{cal}} + 1$ conformity scores:

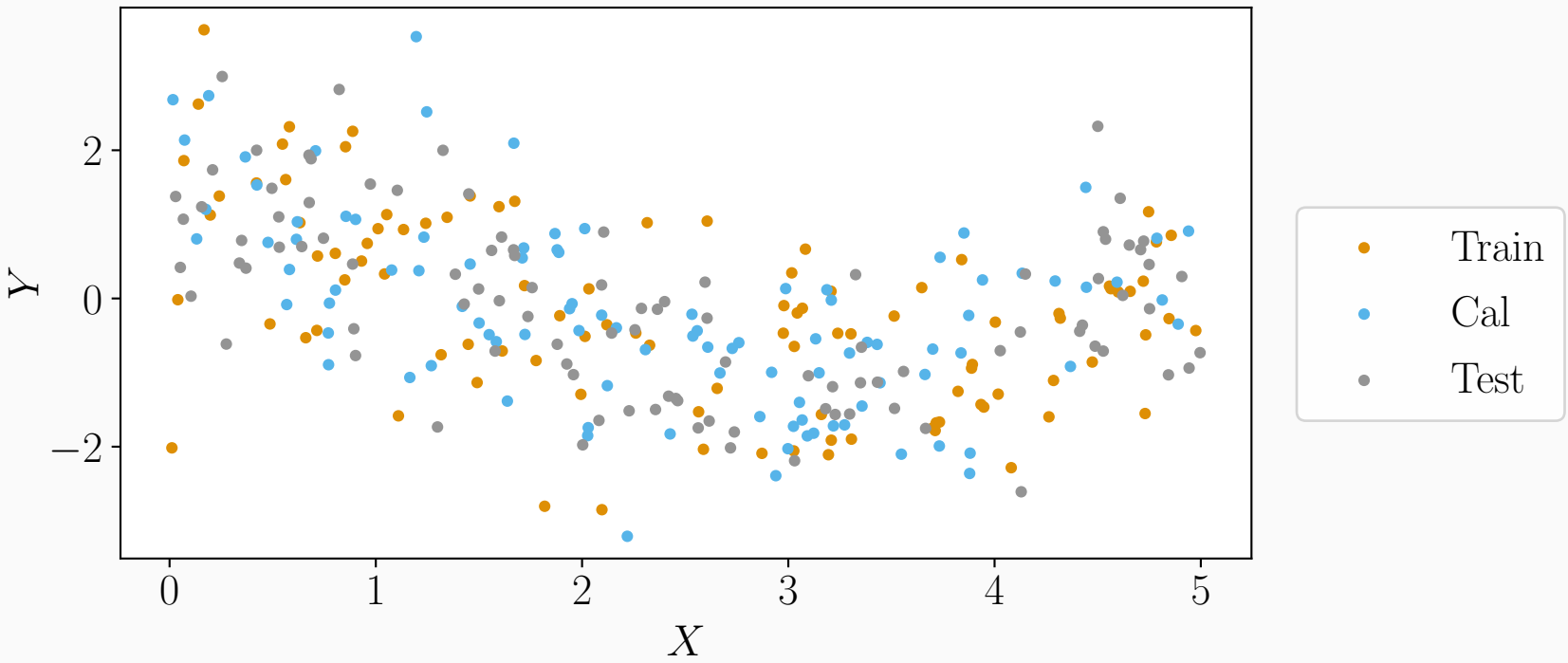
$$\mathcal{S} = \{S_i = |\hat{A}(X_i) - Y_i|, i \in \text{Cal}\} \cup \{+\infty\}$$

(+ worst-case scenario)

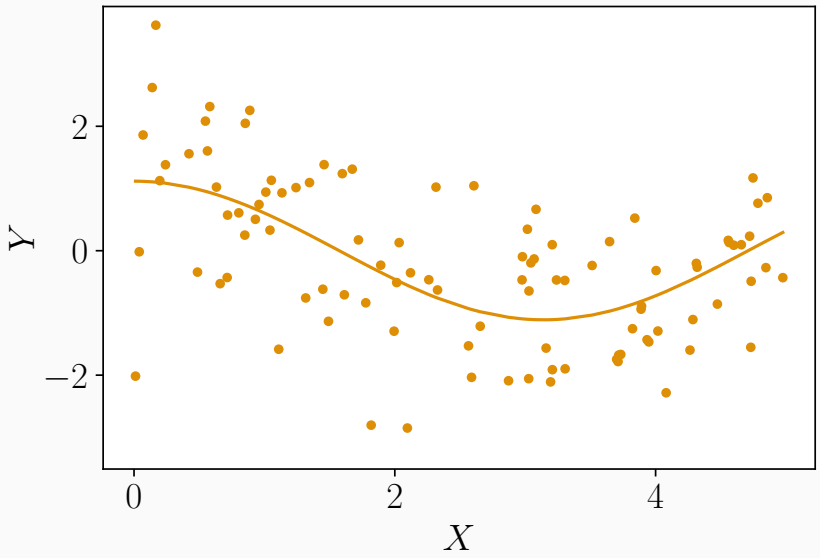
5. Compute the $1 - \alpha$ quantile of these scores, noted $q_{1-\alpha}(\mathcal{S})$
6. For a new point X_{n+1}, return

$$\widehat{\mathcal{C}}_\alpha(X_{n+1}) = \left[\hat{A}(X_{n+1}) - q_{1-\alpha}(\mathcal{S}) ; \hat{A}(X_{n+1}) + q_{1-\alpha}(\mathcal{S})\right]$$

SCP in practice (splitting)

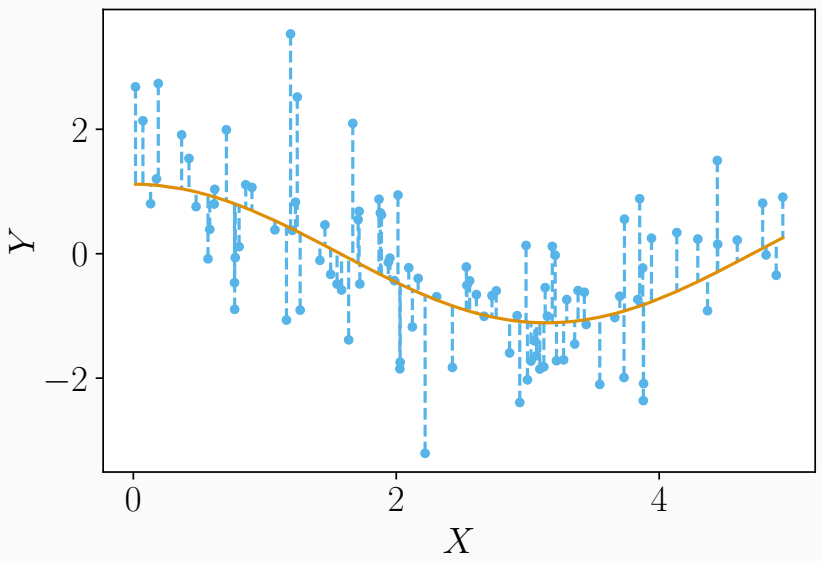


SCP in practice (training)



- Learn $\hat{\mu}$ on the training set

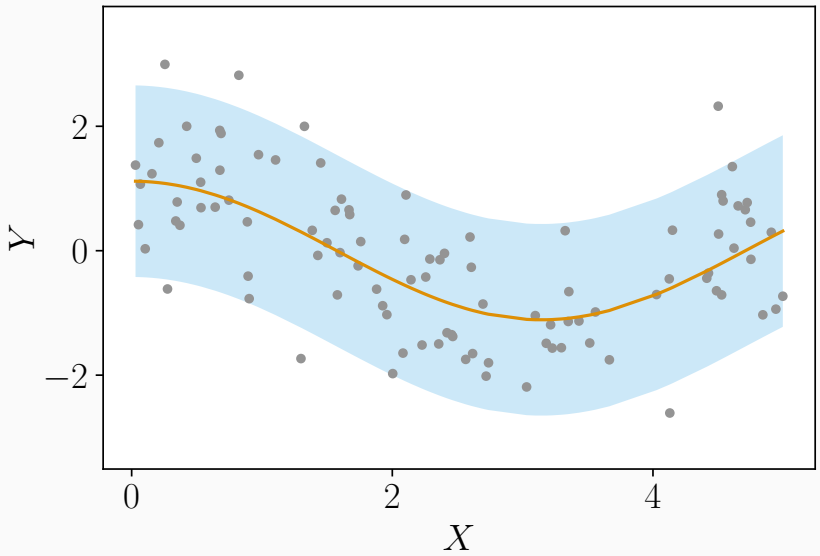
SCP in practice (calibration)



On the calibration set,

- Predict with $\hat{\mu}$
- Get the $|\text{residuals}|$
- Compute the $(1-\alpha)$ empirical quantile of the $|\text{residuals}| \cup \{+\infty\}$, noted $q_{1-\alpha}(\text{residuals})$

SCP in practice (prediction)



On the test set,

- Predict with $\hat{\mu}$
- Build $\widehat{\mathcal{C}}_\alpha(x)$:
  $[\hat{\mu}(x) \pm q_{1-\alpha}\,(\text{residuals})]$

SCP: implementation details

Algorithm 1

1. Split randomly your training data into a proper training set (size n_{train}) and a calibration set (size n_{cal})
2. Train your algorithm $\hat{A}$ on your proper training set
3. On the calibration set, get prediction values with $\hat{A}$
4. Obtain a set of $n_{\text{cal}} + 1$ conformity scores:

$$\mathcal{S} = \{S_i = |\hat{A}(X_i) - Y_i|, i \in \text{Cal}\} \cup \{+\infty\}$$

(+ worst-case scenario)

5. Compute the $1 - \alpha$ quantile of these scores, noted $q_{1-\alpha}(\mathcal{S})$
6. For a new point X_{n+1}, return

$$\widehat{\mathcal{C}}_\alpha(X_{n+1}) = [\hat{A}(X_{n+1}) - q_{1-\alpha}(\mathcal{S}); \hat{A}(X_{n+1}) + q_{1-\alpha}(\mathcal{S})]$$

SCP: implementation details

Algorithm 2

1. Split randomly your training data into a proper training set (size n_{train}) and a calibration set (size n_{cal})
2. Train your algorithm $\hat{A}$ on your proper training set
3. On the calibration set, get prediction values with $\hat{A}$
4. Obtain a set of n_{cal} conformity scores:

$$\mathcal{S} = \{S_i = |\hat{A}(X_i) - Y_i|, i \in \text{Cal}\}$$

5. Compute the $(1-\alpha)\left(\frac{1}{n_{\text{cal}}} + 1\right)$ quantile of these scores, noted $q_{1-\alpha}(\mathcal{S})$
6. For a new point X_{n+1}, return

$$\widehat{\mathcal{C}}_\alpha(X_{n+1}) = [\hat{A}(X_{n+1}) - q_{1-\alpha}(\mathcal{S}); \hat{A}(X_{n+1}) + q_{1-\alpha}(\mathcal{S})]$$

SCP in theory

Definition (Exchangeability)

$(X_i, Y_i)_{i=1}^n$ are exchangeable if for any permutation σ of $\{1, \ldots, n\}$ we have:

$$\mathcal{L}\left((X_1, Y_1), \ldots, (X_n, Y_n)\right) = \mathcal{L}\left(\left(X_{\sigma(1)}, Y_{\sigma(1)}\right), \ldots, \left(X_{\sigma(n)}, Y_{\sigma(n)}\right)\right),$$

where $\mathcal{L}$ designates the joint distribution.

Examples of exchangeable sequences

- i.i.d. samples
- Gaussian samples w/ expectation $m\mathbb{1}_d$ and covariance $\gamma^2 \mathrm{Id}_d + c\mathbb{1}_{d\times d}$

SCP in theory, cont'd

This procedure enjoys the finite sample guarantee proposed and proved in Vovk et al. (2005) and Lei et al. (2018).

Theorem

Suppose $(X_i, Y_i)_{i=1}^{n+1}$ *are exchangeable (or i.i.d.). SCP applied on* $(X_i, Y_i)_{i=1}^{n}$ *outputs an interval* $\widehat{\mathcal{C}}_\alpha(X_{n+1})$ *such that:*

$$\mathbb{P}\left\{Y_{n+1} \in \widehat{\mathcal{C}}_\alpha(X_{n+1})\right\} \geq 1-\alpha.$$

If, in addition, the scores $\{S_i\}_{i\in\mathrm{Cal}}$ *are almost surely distinct, then*

$$\mathbb{P}\left\{Y_{n+1} \in \widehat{\mathcal{C}}_\alpha(X_{n+1})\right\} \leq 1-\alpha+\frac{1}{n_{cal}+1}.$$

✗ Marginal coverage: $\mathbb{P}\left\{Y_{n+1} \in \widehat{\mathcal{C}}_\alpha(X_{n+1}) \,|\, \cancel{X_{n+1}=x}\right\} \geq 1-\alpha$

Proof architecture of SCP guarantees

Lemma (Quantile lemma)

If $(U_1, \ldots, U_n, U_{n+1})$ *are exchangeable, then for any* $\beta \in]0, 1[$*:*

$$\mathbb{P}\left(U_{n+1} \leq q_\beta(U_1, \ldots, U_n, +\infty)\right) \geq \beta.$$

Additionally, if $U_1, \ldots, U_n, U_{n+1}$ *are almost surely distinct, then:*

$$\mathbb{P}\left(U_{n+1} \leq q_\beta(U_1, \ldots, U_n, +\infty)\right) \leq \beta + \frac{1}{n+1}.$$

Note that when $(X_i, Y_i)_{i=1}^{n+1}$ are exchangeable,

- the scores $\{S_i\}_{i \in \mathrm{Cal}} \cup \{S_{n+1}\}$ are exchangeable,
- therefore applying the quantile lemma to the scores concludes the proof.

Proof of the quantile lemma

$$U_{n+1} \leq q_\beta(U_1, \ldots, U_n, +\infty) \iff \frac{|\{i : U_i \leq U_{n+1}\}|}{n+1} \leq \beta$$

$$\iff \text{rank}(U_{n+1}) \leq 1 + \beta(n+1)$$

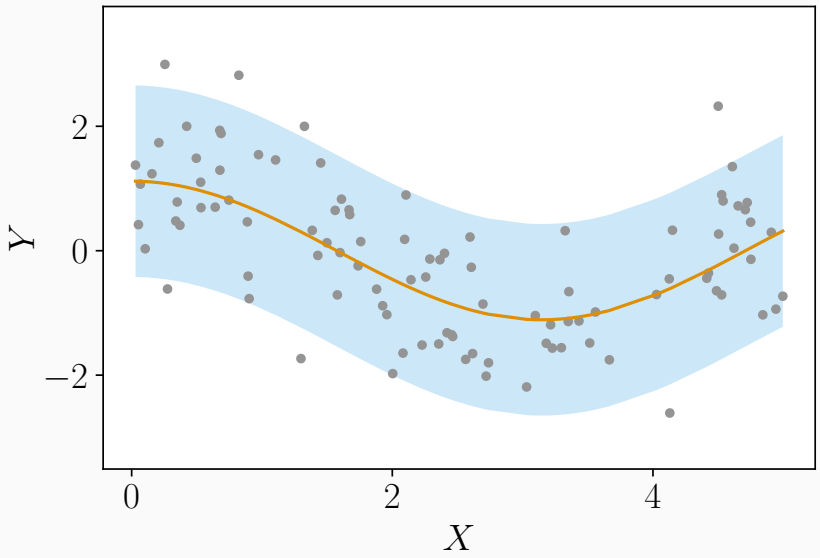
Since $\text{rank}(U_{n+1}) \sim \mathcal{U}(\{1, \ldots, n+1\})$, one gets

$$\mathbb{P}\left(\text{rank}(U_{n+1}) \leq 1 + \beta(n+1)\right) = \frac{\lfloor 1 + \beta(n+1) \rfloor}{n+1}$$

$$\leq \frac{1 + \beta(n+1)}{n+1} = \beta + \frac{1}{n+1}$$

$$\geq \beta \qquad \text{(still true w/ ties)}$$

Standard mean-regression SCP is not adaptive



On the test set,

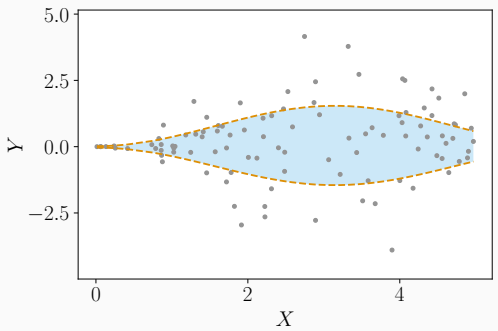
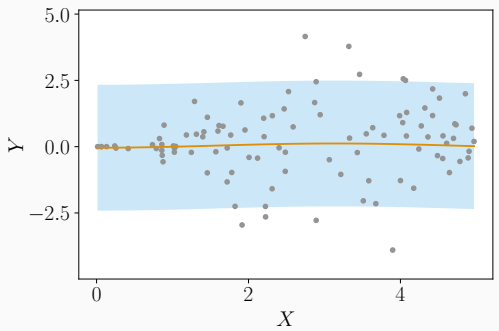
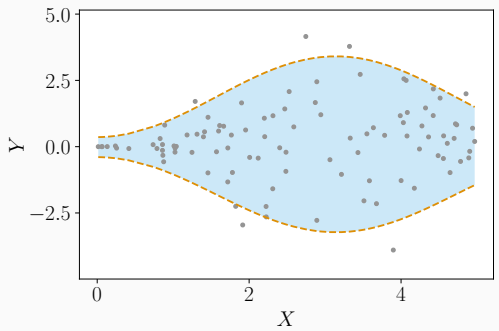
- Predict with $\hat{\mu}$
- Build $\widehat{\mathcal{C}}_\alpha(x)$:
  $[\hat{\mu}(x) \pm q_{1-\alpha}\,(\text{residuals})]$

(Naive) SCP is not adaptive

- Achtung! The conformal prediction procedure with the smallest average set size is not necessarily the best
- A good conformal prediction procedure should give small sets on easy inputs and large sets on hard inputs in a way that faithfully reflects the model's uncertainty
- This adaptivity is not implied by conformal prediction's coverage guarantee
- But it is non-negotiable in practical deployments of conformal prediction

Conditional coverage implies adaptiveness

- Conditional coverage is stronger than marginal coverage
- Marginal coverage: $\mathbb{P}\left\{Y_{n+1} \in \widehat{\mathcal{C}}_\alpha\left(X_{n+1}\right)\right\}$
  the errors may differ across regions of the input space (i.e. non-adaptive)
- Conditional coverage: $\mathbb{P}\left\{Y_{n+1} \in \widehat{\mathcal{C}}_\alpha\left(X_{n+1}\right) | X_{n+1}\right\}$
  errors are evenly distributed (i.e. fully adaptive)

no coverage	marginal	conditional
		

But is conditional coverage possible?

- Impossibility results

  $\hookrightarrow$ Lei and Wasserman (2014); Vovk (2012); Barber et al. (2021a)

  > Without distribution assumption, in finite sample, a perfectly conditionally valid $\widehat{\mathcal{C}}_\alpha$ is such that $\mathbb{E}[\text{mes}(\widehat{\mathcal{C}}_\alpha(x))] = \infty$ for any non-atomic point x.

- Approximate conditional coverage

  $\hookrightarrow$ Romano et al. (2020); Guan (2022); Jung et al. (2023)

  Target $\mathbb{P}(Y_{n+1} \in \widehat{\mathcal{C}}_\alpha | X_{n+1} \in \mathcal{R}(x)) \geq 1 - \alpha$

- Asymptotic (with the sample size) conditional coverage

  $\hookrightarrow$ Romano et al. (2019); Sesia and Romano (2021); Izbicki et al. (2022)

CQR (Romano et al., 2019)

Algorithm 1

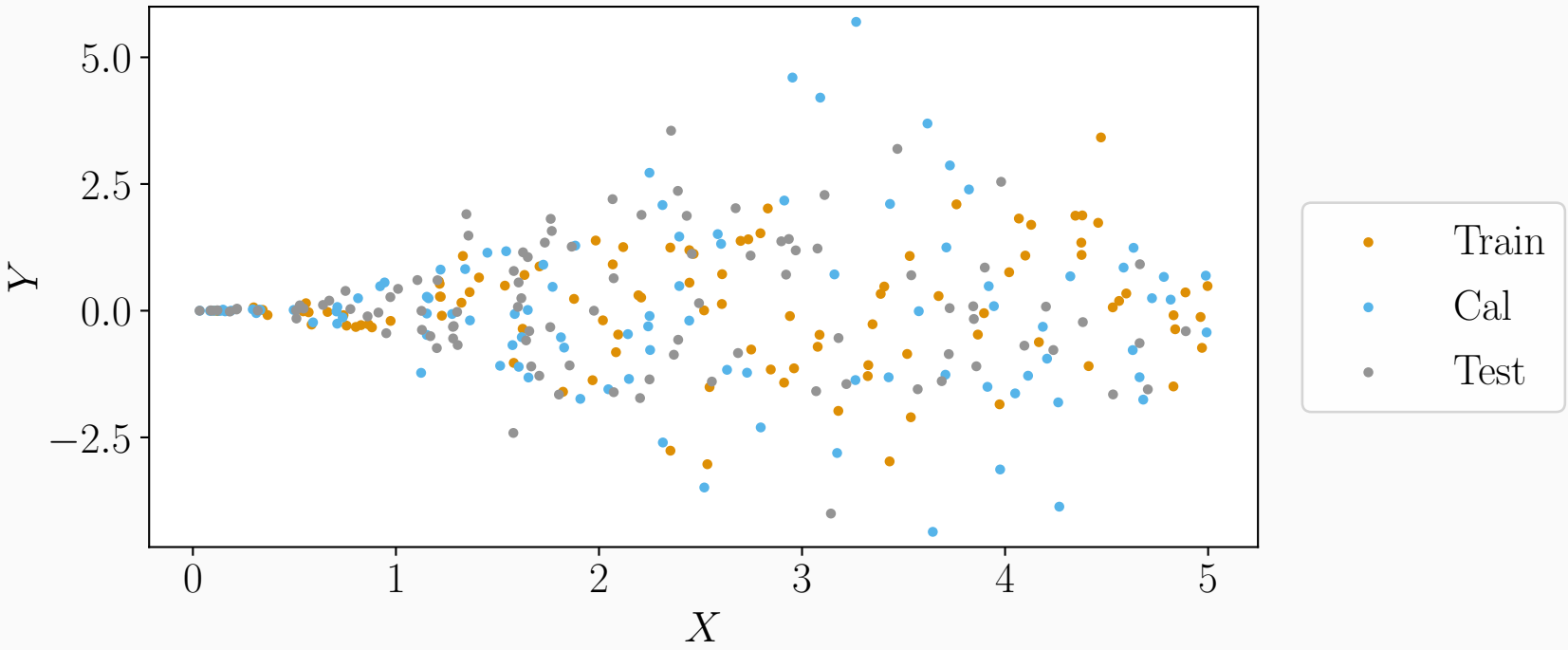
1. Split randomly your training data into a proper training set (size n_{train}) and a calibration set (size n_{cal})
2. Train two algorithms $\widehat{QR}_{\alpha/2}$ and $\widehat{QR}_{1-\alpha/2}$ on the proper training set
3. Obtain a set of $n_{\text{cal}} + 1$ conformity scores $\mathcal{S}$:
   $\mathcal{S} = \{S_i = \max\left(\widehat{QR}_{\alpha/2}(X_i) - Y_i, Y_i - \widehat{QR}_{1-\alpha/2}(X_i)\right), i \in \text{Cal}\} \cup \{+\infty\}$
4. Compute the $1-\alpha$ quantile of these scores, noted $q_{1-\alpha}(\mathcal{S})$
5. For a new point X_{n+1}, return
   $\widehat{\mathcal{C}}_\alpha(X_{n+1}) = [\widehat{QR}_{\alpha/2}(X_{n+1}) - q_{1-\alpha}(\mathcal{S}); \widehat{QR}_{1-\alpha/2}(X_{n+1}) + q_{1-\alpha}(\mathcal{S})]$

CQR (Romano et al., 2019)

Algorithm 2

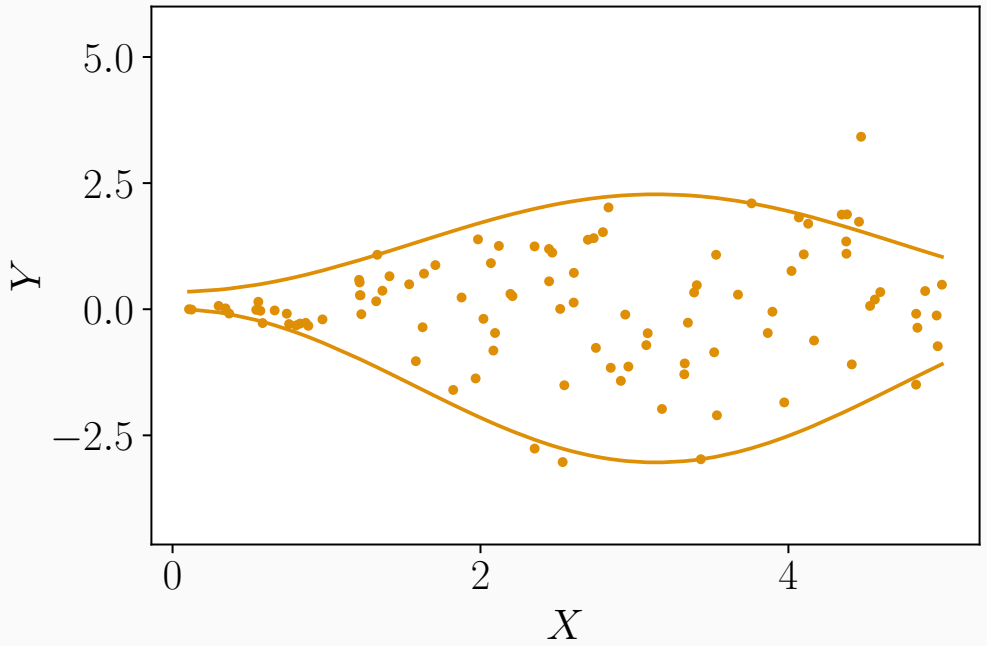
1. Split randomly your training data into a proper training set (size n_{train}) and a calibration set (size n_{cal})
2. Train two algorithms $\widehat{QR}_{\alpha/2}$ and $\widehat{QR}_{1-\alpha/2}$ on the proper training set
3. Obtain a set of n_{cal} conformity scores $\mathcal{S}$:
$$\mathcal{S} = \{S_i = \max\left(\widehat{QR}_{\alpha/2}(X_i) - Y_i, Y_i - \widehat{QR}_{1-\alpha/2}(X_i)\right), i \in \text{Cal}\} \cup \cancel{\{+\infty\}}$$
4. Compute the $(1-\alpha)\left(\frac{1}{n_{\text{cal}}} + 1\right)$ quantile of these scores, noted $q_{1-\alpha}(\mathcal{S})$
5. For a new point X_{n+1}, return
$$\widehat{\mathcal{C}}_\alpha(X_{n+1}) = [\widehat{QR}_{\alpha/2}(X_{n+1}) - q_{1-\alpha}(\mathcal{S}); \widehat{QR}_{1-\alpha/2}(X_{n+1}) + q_{1-\alpha}(\mathcal{S})]$$

CQR in practice



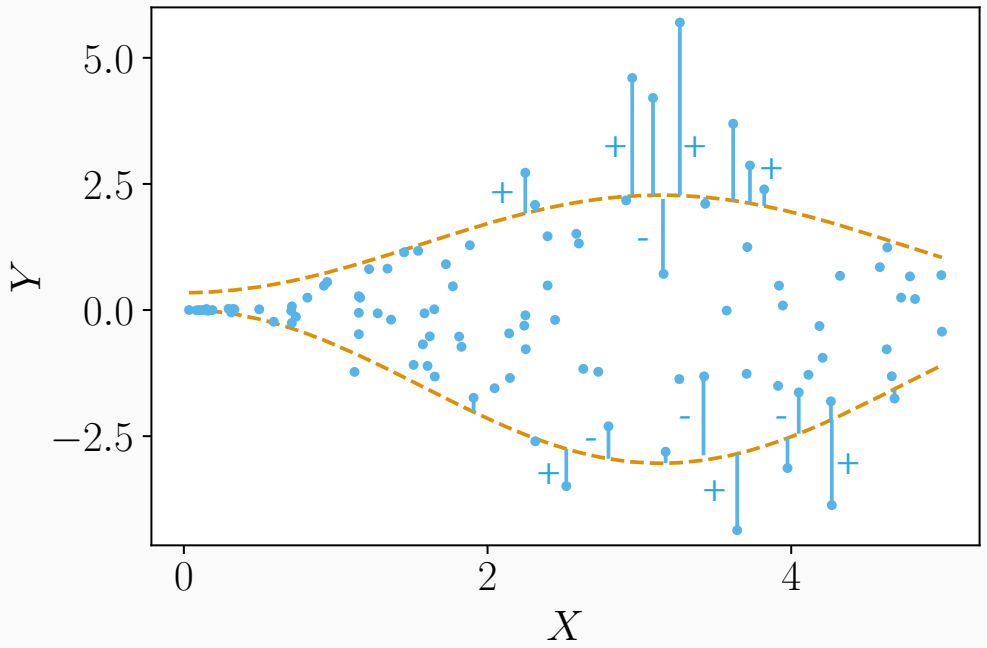
Randomly split the data to obtain a proper training set and a calibration set. Keep the test set.

CQR in practice (training)



- Learn $\widehat{QR}_{\alpha/2}$ and $\widehat{QR}_{1-\alpha/2}$

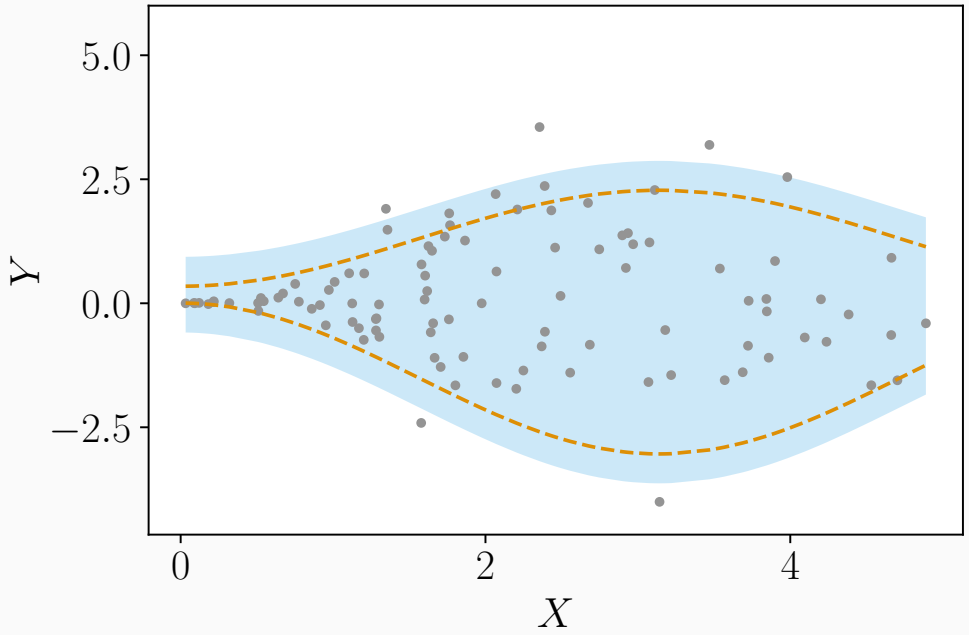
CQR in practice (calibration)



- Predict with $\widehat{QR}_{\alpha/2}$ and $\widehat{QR}_{1-\alpha/2}$
- Compute the scores $\mathcal{S} = \{S_i\}_{\text{Cal}} \cup \{+\infty\}$
- Get the $(1-\alpha)$ empirical quantile of the S_i, noted $q_{1-\alpha}(\mathcal{S})$

$$\hookrightarrow \quad S_i := \max\left\{\widehat{QR}_{\alpha/2}(X_i) - Y_i,\, Y_i - \widehat{QR}_{1-\alpha/2}(X_i)\right\}$$

CQR in practice (prediction)



- Predict with $\widehat{QR}_{\alpha/2}$ and $\widehat{QR}_{1-\alpha/2}$

- Build

$$\widehat{\mathcal{C}}_\alpha(x) = [\widehat{QR}_{\alpha/2}(x) - q_{1-\alpha}(\mathcal{S}); \widehat{QR}_{1-\alpha/2}(x) + q_{1-\alpha}(\mathcal{S})]$$

CQR in theory

This procedure enjoys the finite sample guarantee proposed and proved in Romano et al. (2019).

Theorem

Suppose $(X_i, Y_i)_{i=1}^{n+1}$ *are exchangeable (or i.i.d.). CQR on* $(X_i, Y_i)_{i=1}^{n}$ *outputs* $\widehat{\mathcal{C}}_\alpha(X_{n+1})$ *such that:*

$$\mathbb{P}\left\{Y_{n+1} \in \widehat{\mathcal{C}}_\alpha(X_{n+1})\right\} \geq 1-\alpha.$$

If, in addition, the scores $\{S_i\}_{i\in\mathrm{Cal}}$ *are almost surely distinct, then*

$$\mathbb{P}\left\{Y_{n+1} \in \widehat{\mathcal{C}}_\alpha(X_{n+1})\right\} \leq 1-\alpha+\frac{1}{n_{cal}+1}.$$

Proof: application of the quantile lemma.

✗ Marginal coverage: $\mathbb{P}\left\{Y_{n+1} \in \widehat{\mathcal{C}}_\alpha(X_{n+1}) \,|\, \cancel{X_{n+1}=x}\right\} \geq 1-\alpha$

SCP is defined by the conformity scores

1. Split randomly your training data into a proper training set (size n_{train}) and a calibration set (size n_{cal})
2. Train your algorithm $\hat{A}$ on your proper training set
3. On the calibration set, obtain $n_{\text{cal}} + 1$ conformity scores
   $\mathcal{S} = \{S_i = \text{s}(X_i, Y_i), i \in \text{Cal}\} \cup \{+\infty\}$

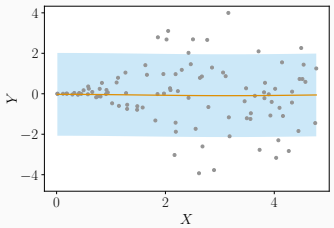
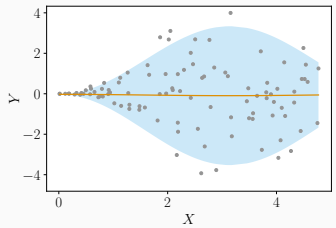
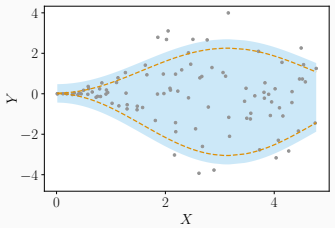
   Ex 1: $\text{s}(X_i, Y_i) = |\hat{A}(X_i) - Y_i|$ in regression with standard scores

   Ex 2: $\text{s}(X_i, Y_i) = \max\left(\widehat{QR}_{\alpha/2}(X_i) - Y_i, Y_i - \widehat{QR}_{1-\alpha/2}(X_i)\right)$ in CQR
4. Compute the $1-\alpha$ quantile of these scores, noted $q_{1-\alpha}(\mathcal{S})$
5. For a new point X_{n+1}, return
   $$\widehat{\mathcal{C}}_\alpha(X_{n+1}) = \{y \text{ such that } \text{s}(\hat{A}(X_{n+1}), y) \leq q_{1-\alpha}(\mathcal{S})\}$$

$\hookrightarrow$ The definition of the conformity scores is crucial, as they incorporate almost all the information: data + underlying model

SCP: what choices for the regression scores?

$$\widehat{\mathcal{C}}_\alpha(X_{n+1}) = \{y \text{ such that } \mathsf{s}(\hat{A}(X_{n+1}), y) \leq q_{1-\alpha}(\mathcal{S})\}$$

	Standard SCP Vovk et al. (2005)	**Locally weighted SCP** Lei et al. (2018)	**CQR** Romano et al. (2019)
$\mathsf{s}(X, Y)$	$\lvert\hat{A}(X) - Y\rvert$	$\dfrac{\lvert\hat{A}(X) - Y\rvert}{\hat{\rho}(X)}$	$\max(\widehat{QR}_{\alpha/2}(X) - Y,$ $Y - \widehat{QR}_{1-\alpha/2}(X))$
$\widehat{\mathcal{C}}_\alpha(x)$	$\left[\hat{A}(x) \pm q_{1-\alpha}(\mathcal{S})\right]$	$\left[\hat{A}(x) \pm q_{1-\alpha}(\mathcal{S})\hat{\rho}(x)\right]$	$[\widehat{QR}_{\alpha/2}(x) - q_{1-\alpha}(\mathcal{S});$ $\widehat{QR}_{1-\alpha/2}(x) + q_{1-\alpha}(\mathcal{S})]$
Visu.			
✓	black-box around a "usable" prediction	black-box around a "usable" prediction	adaptive
✗	not adaptive	limited adaptiveness	no black-box around a "usable" prediction

SCP in classification

- $Y_i \in \{1, \ldots, C\}$ (C classes)
- $\hat{A}(X) = (\hat{p}_1(X), \ldots, \hat{p}_C(X))$ (estimated probabilities)
- Score of the i-th calibration point: $S_i = 1 - (\hat{A}(X_i))_{Y_i}$
- For a new point X_{n+1}, return

$$\widehat{\mathcal{C}}_\alpha(X_{n+1}) = \{y \text{ such that } s(\hat{A}(X_{n+1}), y) \leq q_{1-\alpha}(\mathcal{S})\}$$

SCP in classification in practice

Ex: $Y_i \in \{\text{“dog”}, \text{“tiger”}, \text{“cat”}\}$, with $\alpha = 0.1$

- Scores on the calibration set

Cal_i										
$\hat{p}_{\text{dog}}(X_i)$	0.95	0.90	0.85	0.15	0.15	0.20	0.15	0.15	0.25	0.20
$\hat{p}_{\text{tiger}}(X_i)$	0.02	0.05	0.10	0.60	0.55	0.50	0.45	0.40	0.35	0.45
$\hat{p}_{\text{cat}}(X_i)$	0.03	0.05	0.05	0.25	0.30	0.30	0.40	0.45	0.40	0.35
S_i	0.05	0.1	0.15	0.40	0.45	0.50	0.55	0.55	0.6	0.65

- $q_{1-\alpha}(\mathcal{S}) = 0.65$ $\quad \lceil 0.9 \times (10+1) \rceil = 10$
- $\hat{A}(X_{new}) = (0.05, 0.60, 0.35)$
  - $\hookrightarrow$ $s(\hat{A}(X_{new}), \text{“dog”}) = 0.95$ $\quad$ “dog” $\notin \widehat{\mathcal{C}}_\alpha(X_{new})$
  - $\hookrightarrow$ $s(\hat{A}(X_{new}), \text{“tiger”}) = 0.40 \leq q_{1-\alpha}(\mathcal{S})$ $\quad$ “tiger” $\in \widehat{\mathcal{C}}_\alpha(X_{new})$
  - $\hookrightarrow$ $s(\hat{A}(X_{new}), \text{“cat”}) = 0.65 \leq q_{1-\alpha}(\mathcal{S})$ $\quad$ “cat” $\in \widehat{\mathcal{C}}_\alpha(X_{new})$
- $\widehat{\mathcal{C}}_\alpha(X_{new}) = \{\text{“tiger”}, \text{“cat”}\}$

SCP in classification in practice

Ex: $Y_i \in \{\text{“dog”}, \text{“tiger”}, \text{“cat”}\}$, with $\alpha = 0.1$

- Scores on the calibration set

Cal_i										
$\hat{p}_{\text{dog}}(X_i)$	0.95	0.90	0.85	0.05	0.05	0.05	0.05	0.10	0.10	0.15
$\hat{p}_{\text{tiger}}(X_i)$	0.02	0.05	0.10	0.85	0.80	0.75	0.70	0.25	0.30	0.30
$\hat{p}_{\text{cat}}(X_i)$	0.03	0.05	0.05	0.10	0.15	0.20	0.25	0.65	0.60	0.55
S_i	0.05	0.1	0.15	0.15	0.20	0.25	0.30	0.35	0.40	0.45

- $q_{1-\alpha}(\mathcal{S}) = 0.45$ $\quad \lceil 0.9 \times (10+1) \rceil = 10$
- $\hat{A}(X_{new}) = (0.05, 0.60, 0.35)$
  - $\hookrightarrow$ $s(\hat{A}(X_{new}), \text{“dog”}) = 0.95$ $\quad$ “dog” $\notin \widehat{\mathcal{C}}_\alpha(X_{new})$
  - $\hookrightarrow$ $s(\hat{A}(X_{new}), \text{“tiger”}) = 0.40 \leq q_{1-\alpha}(\mathcal{S})$ $\quad$ “tiger” $\in \widehat{\mathcal{C}}_\alpha(X_{new})$
  - $\hookrightarrow$ $s(\hat{A}(X_{new}), \text{“cat”}) = 0.65$ $\quad$ “cat” $\notin \widehat{\mathcal{C}}_\alpha(X_{new})$
- $\widehat{\mathcal{C}}_\alpha(X_{new}) = \{\text{“tiger”}\}$

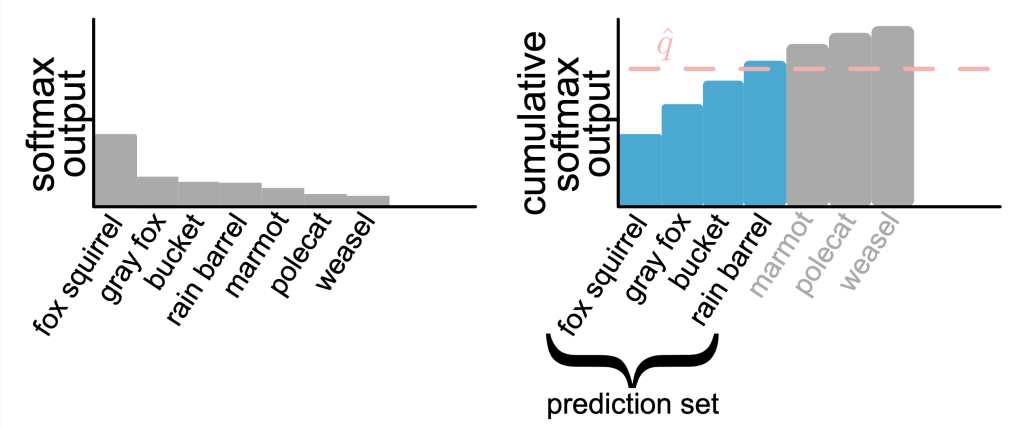
SCP in classification: comments on the naive version

- Facts about the previous method
  - prediction sets with the smallest average size
  - undercover hard subgroups
  - overcover easy ones
- Other types of scores can be used to improve the conditional coverage (as in regression with CQR or localized)

SCP in classification: Adaptive Prediction Sets

1. Sort in decreasing order $\hat{p}_{\sigma_i(1)}(X_i) \geq \ldots \geq \hat{p}_{\sigma_i(C)}(X_i)$
2. $S_i = \sum_{k=1}^{\sigma_i^{-1}(Y_i)} \hat{p}_{\sigma_i(k)}(X_i)$ (sum of the estimated probabilities associated to classes at least as large as that of the true class Y_i)
3. Return the classes $\sigma_{\text{new}}(1), \ldots, \sigma_{\text{new}}(r^\star)$ where

$$r^\star = \underset{1 \leq r \leq C}{\arg\max} \left\{ \sum_{k=1}^{r} \hat{p}_{\sigma_{\text{new}}(k)}(X_{\text{new}}) < q_{1-\alpha}(\mathcal{S}) \right\} + 1$$



SCP in classification in practice: Adaptive Prediction Sets

Ex: $Y_i \in \{\text{“dog”}, \text{“tiger”}, \text{“cat”}\}$, with $\alpha = 0.1$

- Scores on the calibration set

Cal_i										
$\hat{p}_{\text{dog}}(X_i)$	0.95	0.90	0.85	0.05	0.05	0.05	0.10	0.25	0.10	0.15
$\hat{p}_{\text{tiger}}(X_i)$	0.02	0.05	0.10	0.85	0.80	0.75	0.75	0.40	0.30	0.30
$\hat{p}_{\text{cat}}(X_i)$	0.03	0.05	0.05	0.10	0.15	0.20	0.15	0.35	0.60	0.55
S_i	0.95	0.90	0.85	0.85	0.80	0.75	0.75	0.75	0.60	0.55

- $q_{1-\alpha}(\mathcal{S}) = 0.95$
- Ex 1: $\hat{A}(X_{new}) = (0.05, 0.45, 0.5), r^\star = 2$

$$\widehat{\mathcal{C}}_\alpha(X_{new}) = \{\text{“tiger”}, \text{“cat”}\}$$

- Ex 2: $\hat{A}(X_{new}) = (0.03, 0.95, 0.02), r^\star = 1$

$$\widehat{\mathcal{C}}_\alpha(X_{new}) = \{\text{“tiger”}\}$$

Split Conformal prediction: summary

- Simple procedure which
  - quantifies the uncertainty of a predictive model $\hat{A}$
  - by returning predictive regions
- Adapted to any predictive algorithm (neural nets, random forests...)
- Distribution-free as long as the data are exchangeable (and so are the scores)
- Finite-sample guarantees
- Marginal theoretical guarantee over the joint distribution of (X, Y), and not conditional, i.e., no guarantee that $\forall x \in \mathbb{R}$:

$$\mathbb{P}\left\{Y_{n+1} \in \widehat{\mathcal{C}}_\alpha\left(X_{n+1}\right) \cancel{| X_{n+1} = x}\right\} \geq 1 - \alpha.$$

(despite some heuristics)

Challenges: open questions

- Conditional coverage (Previous Sec.)
- Exchangeability (Last Sec.: distribution shift)
- Computational cost vs. statistics power (Next Sec.: Jackknife)

Machine learning context

Quantile Regression

Split Conformal Prediction (SCP)

Jackknife/cross-val

Beyond exchangeability

Beyond the limitations of SCP

- SCP is computationally attractive: it only requires fitting the model one time
- Problem: it sacrifices statistical efficiency
  - requiring splitting the data into training and calibration datasets

$\rightsquigarrow$ Full (or transductive) conformal prediction

  - avoids data splitting
  - at the cost of many more model fits

- Historically, full conformal prediction was developed first
- Idea: we know that the true label Y_{n+1} lives somewhere in $\mathcal{Y}$ so if we loop over all possible $y \in \mathcal{Y}$, then we will eventually hit the data point (X_{n+1}, Y_{n+1}), which is statistically plausible with the first n data points
- Hence the name as full conformal prediction directly computes this loop

Full conformal prediction

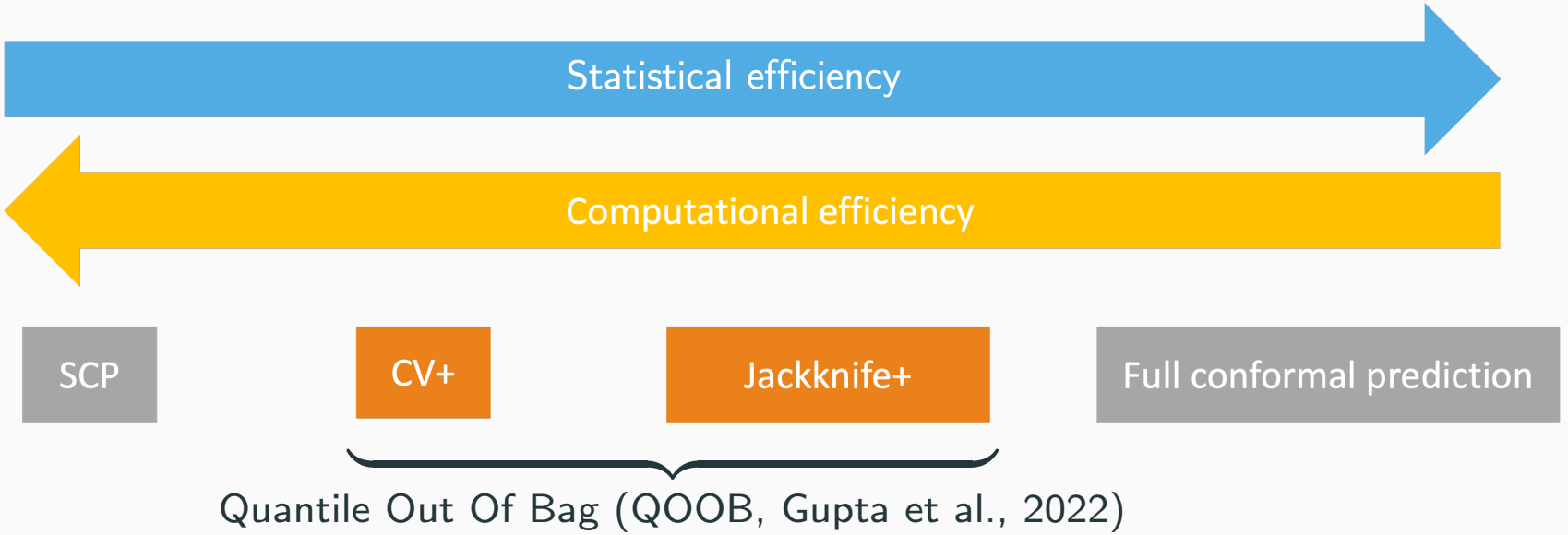
Method: for a candidate (X_{new}, y),

1. Train the algorithm $\hat{A}_y$ on $\{(X_1, Y_1), \ldots, (X_n, Y_n)\} \cup \{(X_{\text{new}}, y)\}$
2. Scores
$$\mathcal{S}^{(\text{train})} = \left\{ s(\hat{A}_y(X_i), Y_i) \right\} \cup \{ s(\hat{A}_y(X_{\text{new}}), y) \}$$
3. $y \in \widehat{C}_\alpha(X_{\text{new}})$ if $s(\hat{A}_y(X_{\text{new}}), y) \leq q_{1-\alpha}(\mathcal{S})$

✓ Theoretical guarantees (provided that $\hat{A}$ handles exchangeable training data in a symmetric way)

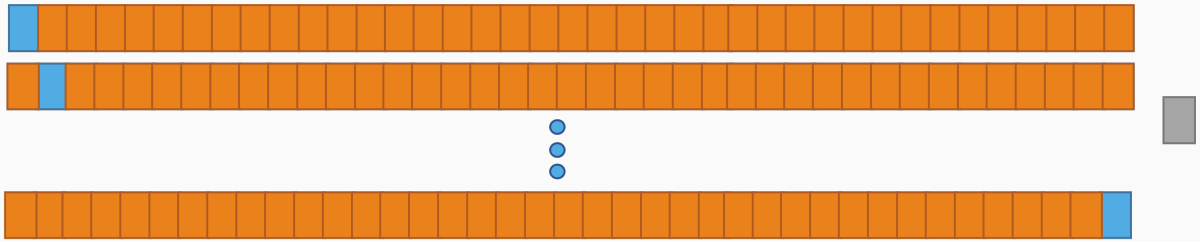
✗ Computationally costly: not used in practice

Other methods for conformal prediction



Jackknife: naive predictive interval

- Based on leave-one-out (LOO) residuals



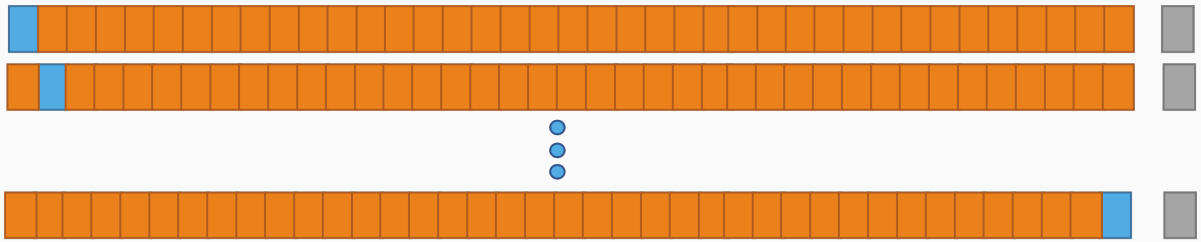
- $\mathcal{D}_n = \{(X_1, Y_1), \ldots, (X_n, Y_n)\}$ training data
- Train $\hat{A}_{-i}$ on $\mathcal{D}_n \setminus (X_i, Y_i)$
- LOO scores $\mathcal{S} = \left\{ |\hat{A}_{-i}(X_i) - Y_i| \right\}_i \cup \{+\infty\}$ (in standard reg)
- Train $\hat{A}$ on $\mathcal{D}_n$
- Build the predictive interval: $\left[\hat{A}(X_{n+1}) \pm q_{1-\alpha}(\mathcal{S}) \right]$

Warning

No guarantee on the prediction of $\hat{A}$ with scores based on $(\hat{A}_{-i})_i$

Jackknife+ (Barber et al., 2021b)

- Based on leave-one-out (LOO) residuals



- $\mathcal{D}_n = \{(X_1, Y_1), \ldots, (X_n, Y_n)\}$ training data
- Train $\hat{A}_{-i}$ on $\mathcal{D}_n \setminus (X_i, Y_i)$
- LOO predictions (in standard reg)

$$\mathcal{S}_{\text{up/down}} = \left\{ \hat{A}_{-i}(X_{n+1}) \pm |\hat{A}_{-i}(X_i) - Y_i| \right\}_i \cup \{\pm\infty\}$$

- Build the predictive interval: $\left[q_{\alpha/2}(\mathcal{S}_{\text{down}}); q_{1-\alpha/2}(\mathcal{S}_{\text{up}})\right]$

Theorem

If $\mathcal{D}_n \cup (X_{new}, Y_{new})$ are exchangeable and the algorithm treats the data points symmetrically, then $\mathbb{P}(Y_{new} \in \widehat{\mathcal{C}}_\alpha(X_{new})) \geq 1 - 2\alpha$.

CV+ (Barber et al., 2021b)

- Based on cross-validation residuals
- $\mathcal{D}_n = \{(X_1, Y_1), \ldots, (X_n, Y_n)\}$ training data

1. Split $\mathcal{D}_n$ into K folds $F_1, \ldots, F_K$
2. Train $\hat{A}_{-F_k}$ on $\mathcal{D}_n \setminus F_k$
3. Cross-val predictions (in standard reg)

$$\mathcal{S}_{\text{up/down}} = \left\{ \left\{ \hat{A}_{-F_k}(X_{n+1}) \pm |\hat{A}_{-F_k}(X_i) - Y_i| \right\}_{i \in F_k} \right\}_k \cup \{\pm\infty\}$$

4. Build the predictive interval: $[q_\alpha(\mathcal{S}_{\text{down}}); q_{1-\alpha}(\mathcal{S}_{\text{up}})]$

Theorem

Under data exchangeability and algorithm symmetry, then

$$\mathbb{P}(Y_{new} \in \widehat{\mathcal{C}}_\alpha(X_{new})) \geq 1 - 2\alpha - \min\left(\frac{2(1-1/K)}{n/K+1}, \frac{1-K/n}{K+1}\right) \geq 1 - 2\alpha - \sqrt{2/n}.$$

Machine learning context

Quantile Regression

Split Conformal Prediction (SCP)

Jackknife/cross-val

Beyond exchangeability

Exchangeability does not hold in many practical applications

- CP requires exchangeable data points to ensure validity
- ✗ Covariate shift, i.e. $\mathcal{L}_X$ changes but $\mathcal{L}_{Y|X}$ stays constant
- ✗ Label shift, i.e. $\mathcal{L}_Y$ changes but $\mathcal{L}_{X|Y}$ stays constant
- ✗ Arbitrary distribution shift
- ✗ Possibly many shifts, not only one

Covariate shift (Tibshirani et al., 2019)

- Setting:
  - $(X_1, Y_1), \ldots, (X_n, Y_n) \overset{i.i.d.}{\sim} P_X \times P_{Y|X}$
  - $(X_{n+1}, Y_{n+1}) \sim \tilde{P}_X \times P_{Y|X}$
- Idea: give more importance to calibration points that are closer in distribution to the test point
- In practice:
  1. estimate the likelihood ratio $w(X_i) = \dfrac{\mathrm{d}\tilde{P}_X(X_i)}{\mathrm{d}P_X(X_i)}$
  2. normalize the weights, i.e. $\omega_i = \omega(X_i) = \dfrac{w(X_i)}{\sum_{j=1}^{n+1} w(X_j)}$
  3. outputs $\widehat{\mathcal{C}}_\alpha(X_{n+1}) = \left\{ y : s(\hat{A}(X_{n+1}), y) \leq q_{1-\alpha}\left(\{\omega_i S_i\}_{i \in \mathrm{Cal}} \cup \{+\infty\} \right) \right\}$

Label shift (Podkopaev and Ramdas, 2021)

- Setting:
  - $(X_1, Y_1), \ldots, (X_n, Y_n) \overset{i.i.d.}{\sim} P_{X|Y} \times P_Y$
  - $(X_{n+1}, Y_{n+1}) \sim P_{X|Y} \times \tilde{P}_Y$
  - Classification
- Idea: give more importance to calibration points that are closer in distribution to the test point
- Trouble: the actual test labels are unknown
- In practice:
  1. estimate the likelihood ratio $w(Y_i) = \dfrac{\mathrm{d}\tilde{P}_Y(Y_i)}{\mathrm{d}P_Y(Y_i)}$ using algorithms from the existing label shift literature
  2. normalize the weights, i.e. $\omega_i^y = \omega^y(X_i) = \dfrac{w(Y_i)}{\sum_{j=1}^n w(Y_j) + w(y)}$
  3. outputs $\widehat{\mathcal{C}}_\alpha(X_{n+1}) = \left\{ y : s(\hat{A}(X_{n+1}), y) \leq q_{1-\alpha}\left(\{\omega_i^y S_i\}_{i \in \mathrm{Cal}} \cup \{+\infty\} \right.\right.$

Generalizations

- Arbitrary distribution shift: Cauchois et al. (2020) leverages ideas from the distributionally robust optimization literature
- Two major general theoretical results beyond exchangeability:
  - Chernozhukov et al. (2018)
    $\hookrightarrow$ If the learnt model is accurate and the data noise is strongly mixing, then CP is valid asymptotically ✓
  - Barber et al. (2022)
    $\hookrightarrow$ Quantifies the coverage loss depending on the strength of exchangeability violation
    $$\mathbb{P}(Y_{n+1} \in \widehat{C}_\alpha(X_{n+1})) \geq 1 - \alpha - \underbrace{\text{average violation of exchangeability by each calibration point}}$$
    $\hookrightarrow$ proposed algorithm: reweighting again!
    e.g., in a temporal setting, give higher weights to more recent points.

Online setting

- Data: T_0 random variables $(X_1, Y_1), \ldots, (X_{T_0}, Y_{T_0})$ in $\mathbb{R}^d \times \mathbb{R}$
- Aim: predict the response values as well as predictive intervals for T_1 subsequent observations $X_{T_0+1}, \ldots, X_{T_0+T_1}$ sequentially: at any prediction step $t \in [\![T_0 + 1, T_0 + T_1]\!]$, $Y_{t-T_0}, \ldots, Y_{t-1}$ have been revealed
- Build the smallest interval $\widehat{\mathcal{C}}_\alpha^t$ such that:

$$\mathbb{P}\left\{Y_t \in \widehat{\mathcal{C}}_\alpha^t(X_t)\right\} \geq 1 - \alpha, \quad \text{for } t \in [\![T_0 + 1, T_0 + T_1]\!].$$

Recent developments

- Consider splitting strategies that respect the temporal structure
- Gibbs and Candès (2021) propose a method which reacts faster to temporal evolution
  - Idea: track the previous coverages of the predictive intervals $(\mathbb{1}\{Y_t \in \widehat{\mathcal{C}}_\alpha(X_t)\})$
  - Tool: update the empirical quantile level with a learning rate γ
  - Asymptotic guarantee (on average) for any distribution (even adversarial)
- Zaffran et al. (2022) studies the influence of this learning rate γ and proposes, along with Gibbs and Candès (2022), a method that does not require to choose γ

References i

Angelopoulos, A. N. and Bates, S. (2023). Conformal prediction: A gentle introduction. *Foundations and Trends® in Machine Learning*, 16(4):494–591.

Barber, R. F., Candès, E. J., Ramdas, A., and Tibshirani, R. J. (2021a). The limits of distribution-free conditional predictive inference. *Information and Inference: A Journal of the IMA*, 10(2):455–482.

Barber, R. F., Candès, E. J., Ramdas, A., and Tibshirani, R. J. (2021b). Predictive inference with the jackknife+. *The Annals of Statistics*, 49(1):486–507.

Barber, R. F., Candès, E. J., Ramdas, A., and Tibshirani, R. J. (2022). Conformal prediction beyond exchangeability. To appear in *Annals of Statistics (2023)*.

References ii

Cauchois, M., Gupta, S., Ali, A., and Duchi, J. C. (2020). Robust Validation: Confident Predictions Even When Distributions Shift. *arXiv:2008.04267 [cs, stat]*. arXiv: 2008.04267.

Chernozhukov, V., Wüthrich, K., and Yinchu, Z. (2018). Exact and Robust Conformal Inference Methods for Predictive Machine Learning with Dependent Data. In *Conference On Learning Theory*, pages 732–749. PMLR. ISSN: 2640-3498.

Gibbs, I. and Candès, E. (2021). Adaptive conformal inference under distribution shift. In *Advances in Neural Information Processing Systems*, volume 34, pages 1660–1672.

Gibbs, I. and Candès, E. (2022). Conformal inference for online prediction with arbitrary distribution shifts.

References iii

Guan, L. (2022). Localized conformal prediction: a generalized inference framework for conformal prediction. *Biometrika*, 110(1):33–50.

Gupta, C., Kuchibhotla, A. K., and Ramdas, A. (2022). Nested conformal prediction and quantile out-of-bag ensemble methods. *Pattern Recognition*, 127:108496.

Izbicki, R., Shimizu, G., and Stern, R. B. (2022). Cd-split and hpd-split: Efficient conformal regions in high dimensions. *Journal of Machine Learning Research*, 23(87):1–32.

Jung, C., Noarov, G., Ramalingam, R., and Roth, A. (2023). Batch multivalid conformal prediction. In *International Conference on Learning Representations*.

References iv

Lei, J., G'Sell, M., Rinaldo, A., Tibshirani, R. J., and Wasserman, L. (2018). Distribution-Free Predictive Inference for Regression. *Journal of the American Statistical Association*.

Lei, J. and Wasserman, L. (2014). Distribution-free prediction bands for non-parametric regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 76(1):71–96.

Podkopaev, A. and Ramdas, A. (2021). Distribution-free uncertainty quantification for classification under label shift. In *Proceedings of the Thirty-Seventh Conference on Uncertainty in Artificial Intelligence*, volume 161, pages 844–853. PMLR.

References v

Romano, Y., Barber, R. F., Sabatti, C., and Candès, E. (2020). With Malice Toward None: Assessing Uncertainty via Equalized Coverage. *Harvard Data Science Review*, 2(2).

Romano, Y., Patterson, E., and Candès, E. (2019). Conformalized Quantile Regression. *Advances in Neural Information Processing Systems*, 32.

Sesia, M. and Romano, Y. (2021). Conformal prediction using conditional histograms. In *Advances in Neural Information Processing Systems*, volume 34, pages 6304–6315.

Tibshirani, R. J., Barber, R. F., Candès, E., and Ramdas, A. (2019). Conformal Prediction Under Covariate Shift. *Advances in Neural Information Processing Systems*, 32:11.

References vi

Vovk, V. (2012). Conditional Validity of Inductive Conformal Predictors. In *Asian Conference on Machine Learning*, pages 475–490. PMLR. ISSN: 1938-7228.

Vovk, V., Gammerman, A., and Shafer, G. (2005). *Algorithmic Learning in a Random World*. Springer US.

Zaffran, M., Féron, O., Goude, Y., Josse, J., and Dieuleveut, A. (2022). Adaptive conformal predictions for time series. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162, pages 25834–25866. PMLR.