NB: Open with an advanced pdf reader (e.g., Acrobat to have animations)

Lecture on Conformal Prediction

Margaux Zaffran

December 1-5, 2025
ECAS-SFdS School

Previous lectures

1. On exchangeability (theory)
2. Split conformal prediction (methods) (theory)
3. Towards conditional coverage? (practical session) (theory) (case studies)
4. Beyond exchangeability (methods) (case studies)

Achievable definition of distribution-free validity

$\widehat{C}_\alpha$ = estimated predictive set based on n data points.

Definition (Distribution-free validity).

$\widehat{C}_\alpha$ achieves distribution-free validity if:

- for any distribution $\mathcal{D}$,
- for any associated exchangeable joint distribution $\mathcal{D}^{\mathrm{exch}(n+1)}$,

we have that:

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

SCP is defined by the conformity score function

1. Randomly split the training data into a proper training set (size $\#\mathrm{Tr}$) and a calibration set (size $\#\mathrm{Cal}$)

SCP is defined by the conformity score function

1. Randomly split the training data into a proper training set (size $\#\mathrm{Tr}$) and a calibration set (size $\#\mathrm{Cal}$)
2. Get $\hat{A}$ *by training the algorithm* $\mathcal{A}$ *on the proper training set*

SCP is defined by the conformity score function

1. Randomly split the training data into a proper training set (size $\#\mathrm{Tr}$) and a calibration set (size $\#\mathrm{Cal}$)
2. Get $\hat{A}$ *by training the algorithm* $\mathcal{A}$ *on the proper training set*
3. On the calibration set, obtain $\#\mathrm{Cal} + 1$ conformity scores

$$\mathcal{S} = \{S_i = \mathsf{s}\left(X_i, Y_i; \hat{A}\right), i \in \mathrm{Cal}\} \cup \{+\infty\}$$

SCP is defined by the conformity score function

1. Randomly split the training data into a proper training set (size $\#\mathrm{Tr}$) and a calibration set (size $\#\mathrm{Cal}$)
2. Get $\hat{A}$ *by training the algorithm* $\mathcal{A}$ *on the proper training set*
3. On the calibration set, obtain $\#\mathrm{Cal} + 1$ conformity scores

$$\mathcal{S} = \{S_i = \mathsf{s}\left(X_i, Y_i; \hat{A}\right), i \in \mathrm{Cal}\} \cup \{+\infty\}$$

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

SCP is defined by the conformity score function

1. Randomly split the training data into a proper training set (size $\#\mathrm{Tr}$) and a calibration set (size $\#\mathrm{Cal}$)
2. Get $\hat{A}$ *by training the algorithm* $\mathcal{A}$ *on the proper training set*
3. On the calibration set, obtain $\#\mathrm{Cal}+1$ conformity scores

$$\mathcal{S} = \{S_i = \mathsf{s}\left(X_i, Y_i; \hat{A}\right), i \in \mathrm{Cal}\} \cup \{+\infty\}$$

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

Ex 2: $\mathsf{s}(\hat{A}(X_i), Y_i) := \max\left(\widehat{\mathrm{QR}}_{\mathrm{lower}}(X_i) - Y_i, Y_i - \widehat{\mathrm{QR}}_{\mathrm{upper}}(X_i)\right)$ in CQR

SCP is defined by the conformity score function

1. Randomly split the training data into a proper training set (size $\#\mathrm{Tr}$) and a calibration set (size $\#\mathrm{Cal}$)
2. Get $\hat{A}$ *by training the algorithm* $\mathcal{A}$ *on the proper training set*
3. On the calibration set, obtain $\#\mathrm{Cal} + 1$ conformity scores

$$\mathcal{S} = \{S_i = \mathrm{s}\left(X_i, Y_i; \hat{A}\right), i \in \mathrm{Cal}\} \cup \{+\infty\}$$

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

Ex 2: $\mathrm{s}(\hat{A}(X_i), Y_i) := \max\left(\widehat{\mathrm{QR}}_{\mathrm{lower}}(X_i) - Y_i, Y_i - \widehat{\mathrm{QR}}_{\mathrm{upper}}(X_i)\right)$ in CQR

4. Compute the $1 - \alpha$ quantile of these scores, noted $q_{1-\alpha}(\mathcal{S})$

SCP is defined by the conformity score function

1. Randomly split the training data into a proper training set (size $\#\mathrm{Tr}$) and a calibration set (size $\#\mathrm{Cal}$)
2. Get $\hat{A}$ *by training the algorithm* $\mathcal{A}$ *on the proper training set*
3. On the calibration set, obtain $\#\mathrm{Cal}+1$ conformity scores

$$\mathcal{S} = \{S_i = \mathsf{s}\left(X_i, Y_i; \hat{A}\right), i \in \mathrm{Cal}\} \cup \{+\infty\}$$

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

   Ex 2: $\mathsf{s}(\hat{A}(X_i), Y_i) := \max\left(\widehat{\mathrm{QR}}_{\mathrm{lower}}(X_i) - Y_i, Y_i - \widehat{\mathrm{QR}}_{\mathrm{upper}}(X_i)\right)$ 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

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

SCP is defined by the conformity score function

1. Randomly split the training data into a proper training set (size $\#\mathrm{Tr}$) and a calibration set (size $\#\mathrm{Cal}$)
2. Get $\hat{A}$ *by training the algorithm* $\mathcal{A}$ *on the proper training set*
3. On the calibration set, obtain $\#\mathrm{Cal}+1$ conformity scores

$$\mathcal{S} = \{S_i = \mathsf{s}\left(X_i, Y_i; \hat{A}\right), i \in \mathrm{Cal}\} \cup \{+\infty\}$$

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

Ex 2: $\mathsf{s}(\hat{A}(X_i), Y_i) := \max\left(\widehat{\mathrm{QR}}_{\mathrm{lower}}(X_i) - Y_i, Y_i - \widehat{\mathrm{QR}}_{\mathrm{upper}}(X_i)\right)$ 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

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

Ex 1: $\widehat{C}_\alpha(X_{n+1}) = [\hat{\mu}(X_{n+1}) \pm q_{1-\alpha}(\mathcal{S})]$

SCP is defined by the conformity score function

1. Randomly split the training data into a proper training set (size $\#\mathrm{Tr}$) and a calibration set (size $\#\mathrm{Cal}$)
2. Get $\hat{A}$ *by training the algorithm* $\mathcal{A}$ *on the proper training set*
3. On the calibration set, obtain $\#\mathrm{Cal}+1$ conformity scores

$$\mathcal{S} = \{S_i = \mathsf{s}\left(X_i, Y_i; \hat{A}\right), i \in \mathrm{Cal}\} \cup \{+\infty\}$$

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

   Ex 2: $\mathsf{s}(\hat{A}(X_i), Y_i) := \max\left(\widehat{\mathrm{QR}}_{\mathrm{lower}}(X_i) - Y_i, Y_i - \widehat{\mathrm{QR}}_{\mathrm{upper}}(X_i)\right)$ 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

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

   Ex 2: $\widehat{C}_\alpha(X_{n+1}) = [\widehat{QR}_{\mathrm{lower}}(X_{n+1}) - q_{1-\alpha}(\mathcal{S});$

$$\widehat{QR}_{\mathrm{upper}}(X_{n+1}) + q_{1-\alpha}(\mathcal{S})]$$

SCP is defined by the conformity score function

1. Randomly split the training data into a proper training set (size $\#\mathrm{Tr}$) and a calibration set (size $\#\mathrm{Cal}$)
2. Get $\hat{A}$ *by training the algorithm* $\mathcal{A}$ *on the proper training set*
3. On the calibration set, obtain $\#\mathrm{Cal} + 1$ conformity scores

$$\mathcal{S} = \{S_i = \mathrm{s}\left(X_i, Y_i; \hat{A}\right), i \in \mathrm{Cal}\} \cup \{+\infty\}$$

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

   Ex 2: $\mathrm{s}(\hat{A}(X_i), Y_i) := \max\left(\widehat{\mathrm{QR}}_{\mathrm{lower}}(X_i) - Y_i, Y_i - \widehat{\mathrm{QR}}_{\mathrm{upper}}(X_i)\right)$ 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

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

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

Split Conformal Prediction: summary

- **Simple** procedure which quantifies the uncertainty of **any** predictive model $\hat{A}$ by returning predictive regions
- **Finite-sample** guarantees
- **Distribution-free** as long as the data are exchangeable (and so are the scores)

Split Conformal Prediction: summary

- **Simple** procedure which quantifies the uncertainty of **any** predictive model $\hat{A}$ by returning predictive regions
- **Finite-sample** guarantees
- **Distribution-free** as long as the data are exchangeable (and so are the scores)
- **Marginal** theoretical guarantee over the joint (X, Y) distribution, and not conditional, i.e., no guarantee that for any x:

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

Split Conformal Prediction: summary

- **Simple** procedure which quantifies the uncertainty of **any** predictive model $\hat{A}$ by returning predictive regions
- **Finite-sample** guarantees
- **Distribution-free** as long as the data are exchangeable (and so are the scores)
- **Marginal** theoretical guarantee over the joint (X, Y) distribution, and not conditional, i.e., no guarantee that for any x:

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

$\hookrightarrow$ marginal also over the whole calibration set and the test point!

Calibration-condition coverage distribution under no tie

Theorem (Distribution conditional on the calibration data)**.**

If the scores are a.s. distinct, SCP outputs $\widehat{C}_\alpha$ such that for any distribution $\mathcal{D}$:

$$\mathbb{P}_{\mathcal{D}}\left(Y_{n+1} \in \widehat{C}_\alpha(X_{n+1}) | (X_i, Y_i)_{i\in \mathrm{Cal}}\right) \sim \beta\left(k_\alpha, \#\mathrm{Cal} + 1 - k_\alpha\right),$$

with $k_\alpha = \lceil (1-\alpha)(\#\mathrm{Cal}+1) \rceil$.

From the β distribution, we get that it has

expectation $\dfrac{k_\alpha}{k_\alpha + \#\mathrm{Cal} + 1 - k_\alpha} = \dfrac{k_\alpha}{\#\mathrm{Cal}+1} = \dfrac{\lceil (1-\alpha)(\#\mathrm{Cal}+1) \rceil}{\#\mathrm{Cal}+1} \geq 1-\alpha$,

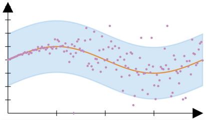
and variance $\dfrac{k_\alpha\left(\#\mathrm{Cal}+1-k_\alpha\right)}{\left(\#\mathrm{Cal}+1\right)^2\left(\#\mathrm{Cal}+2\right)} \approx \dfrac{\alpha(1-\alpha)}{\#\mathrm{Cal}+2}$.

SCP: what choices for the regression scores?

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

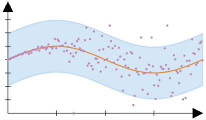
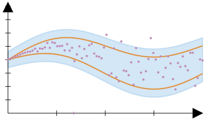
SCP: what choices for the regression scores?

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

	Standard SCP Vovk et al. (2005)		
$s(\hat{A}(X), Y)$	$\lvert\hat{\mu}(X) - Y\rvert$		
$\widehat{C}_\alpha(x)$	$[\hat{\mu}(x) \pm q_{1-\alpha}(\mathcal{S})]$		
Visu.			
✓	black-box around a "usable" prediction		
✗	not adaptive		

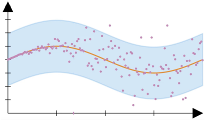
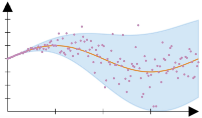
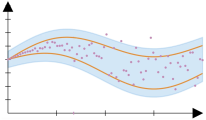
SCP: what choices for the regression scores?

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

	Standard SCP Vovk et al. (2005)		**CQR** Romano et al. (2019)
$s(\hat{A}(X), Y)$	$\lvert\hat{\mu}(X) - Y\rvert$		$\max(\widehat{\mathrm{QR}}_{\mathrm{lower}}(X) - Y, Y - \widehat{\mathrm{QR}}_{\mathrm{upper}}(X))$
$\widehat{C}_\alpha(x)$	$[\hat{\mu}(x) \pm q_{1-\alpha}(\mathcal{S})]$		$[\widehat{\mathrm{QR}}_{\mathrm{lower}}(x) - q_{1-\alpha}(\mathcal{S}); \widehat{\mathrm{QR}}_{\mathrm{upper}}(x) + q_{1-\alpha}(\mathcal{S})]$
Visu.			
✓	black-box around a "usable" prediction		adaptive
✗	not adaptive		no black-box around a "usable" prediction

SCP: what choices for the regression scores?

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

	Standard SCP Vovk et al. (2005)	**Locally weighted SCP** Lei et al. (2018)	**CQR** Romano et al. (2019)
$\mathrm{s}(\hat{A}(X), Y)$	$\lvert\hat{\mu}(X) - Y\rvert$	$\dfrac{\lvert\hat{\mu}(X) - Y\rvert}{\hat{\rho}(X)}$	$\max(\widehat{\mathrm{QR}}_{\text{lower}}(X) - Y, Y - \widehat{\mathrm{QR}}_{\text{upper}}(X))$
$\widehat{C}_\alpha(x)$	$[\hat{\mu}(x) \pm q_{1-\alpha}(\mathcal{S})]$	$[\hat{\mu}(x) \pm q_{1-\alpha}(\mathcal{S})\hat{\rho}(x)]$	$[\widehat{\mathrm{QR}}_{\text{lower}}(x) - q_{1-\alpha}(\mathcal{S}); \widehat{\mathrm{QR}}_{\text{upper}}(x) + q_{1-\alpha}(\mathcal{S})]$
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

Another view on SCP (nested sets, Gupta et al., 2022)

1. Randomly split the training data into a proper training set (size $\#\mathrm{Tr}$) and a calibration set (size $\#\mathrm{Cal}$)

Another view on SCP (nested sets, Gupta et al., 2022)

1. Randomly split the training data into a proper training set (size $\#\mathrm{Tr}$) and a calibration set (size $\#\mathrm{Cal}$)
2. Get $\hat{A}$ *by training the algorithm* $\mathcal{A}$ *on the proper training set*

Another view on SCP (nested sets, Gupta et al., 2022)

1. Randomly split the training data into a proper training set (size $\#\mathrm{Tr}$) and a calibration set (size $\#\mathrm{Cal}$)
2. Get $\hat{A}$ *by training the algorithm* $\mathcal{A}$ *on the proper training set*
3. Build a sequence of nested predictive sets taking their values in $\mathcal{Y}$: $\left(R_t\left(\cdot; \hat{A}\right)\right)_{t \in \mathcal{T}}$ for some $\mathcal{T} \subseteq \mathbb{R}$, such that for $t \leq t'$: $R_t\left(x; \hat{A}\right) \subseteq R_{t'}\left(x; \hat{A}\right)$

Another view on SCP (nested sets, Gupta et al., 2022)

1. Randomly split the training data into a proper training set (size $\#\mathrm{Tr}$) and a calibration set (size $\#\mathrm{Cal}$)
2. Get $\hat{A}$ *by training the algorithm* $\mathcal{A}$ *on the proper training set*
3. Build a sequence of nested predictive sets taking their values in $\mathcal{Y}$: $\left(R_t \left(\cdot ; \hat{A} \right) \right)_{t \in \mathcal{T}}$

   for some $\mathcal{T} \subseteq \mathbb{R}$, such that for $t \leq t'$: $R_t \left(x; \hat{A} \right) \subseteq R_{t'} \left(x; \hat{A} \right)$

   Ex 1: $R_t \ (\cdot ; \hat{\mu}) \equiv [\hat{\mu}(\cdot) \pm t]$ and $\mathcal{T} = \mathbb{R}_+$ in regression with standard scores

1. Randomly split the training data into a proper training set (size $\#\mathrm{Tr}$) and a calibration set (size $\#\mathrm{Cal}$)
2. Get $\hat{A}$ *by training the algorithm* $\mathcal{A}$ *on the proper training set*
3. Build a sequence of nested predictive sets taking their values in $\mathcal{Y}$: $\left(R_t\left(\cdot;\hat{A}\right)\right)_{t\in\mathcal{T}}$ for some $\mathcal{T}\subseteq\mathbb{R}$, such that for $t\leq t'$: $R_t\left(x;\hat{A}\right)\subseteq R_{t'}\left(x;\hat{A}\right)$

   Ex 1: $R_t\left(\cdot;\hat{\mu}\right)\equiv\left[\hat{\mu}(\cdot)\pm t\right]$ and $\mathcal{T}=\mathbb{R}_+$ in regression with standard scores
4. Entry radius of y in the sets given by x are then computed on each of the calibration points as: $\hat{r}\left(X_i,Y_i\right):=\inf\left\{t\in\mathcal{T}:Y_i\in R_t\left(X_i;\hat{A}\right)\right\}, i\in\mathrm{Cal}$

   $\hookrightarrow$ They inherit from data exchangeability and play the role of the conformity scores

Another view on SCP (nested sets, Gupta et al., 2022)

1. Randomly split the training data into a proper training set (size $\#\mathrm{Tr}$) and a calibration set (size $\#\mathrm{Cal}$)
2. Get $\hat{A}$ *by training the algorithm* $\mathcal{A}$ *on the proper training set*
3. Build a sequence of nested predictive sets taking their values in $\mathcal{Y}$: $\left(R_t\left(\cdot; \hat{A}\right)\right)_{t \in \mathcal{T}}$ for some $\mathcal{T} \subseteq \mathbb{R}$, such that for $t \leq t'$: $R_t\left(x; \hat{A}\right) \subseteq R_{t'}\left(x; \hat{A}\right)$

   Ex 1: $R_t(\cdot; \hat{\mu}) \equiv [\hat{\mu}(\cdot) \pm t]$ and $\mathcal{T} = \mathbb{R}_+$ in regression with standard scores
4. Entry radius of y in the sets given by x are then computed on each of the calibration points as: $\hat{r}(X_i, Y_i) := \inf\left\{t \in \mathcal{T} : Y_i \in R_t\left(X_i; \hat{A}\right)\right\}, i \in \mathrm{Cal}$

   $\hookrightarrow$ They inherit from data exchangeability and play the role of the conformity scores

$\Rightarrow$ Denote the set of entry radii $\mathcal{R} = \left\{(\hat{r}(X_i, Y_i))_{i \in \mathrm{Cal}}\right\} \cup \{+\infty\}$

Another view on SCP (nested sets, Gupta et al., 2022)

1. Randomly split the training data into a proper training set (size $\#\mathrm{Tr}$) and a calibration set (size $\#\mathrm{Cal}$)
2. Get $\hat{A}$ *by training the algorithm $\mathcal{A}$ on the proper training set*
3. Build a sequence of nested predictive sets taking their values in $\mathcal{Y}$: $\left(R_t\left(\cdot;\hat{A}\right)\right)_{t\in\mathcal{T}}$ for some $\mathcal{T}\subseteq\mathbb{R}$, such that for $t\leq t'$: $R_t\left(x;\hat{A}\right)\subseteq R_{t'}\left(x;\hat{A}\right)$

   Ex 1: $R_t\left(\cdot;\hat{\mu}\right)\equiv[\hat{\mu}(\cdot)\pm t]$ and $\mathcal{T}=\mathbb{R}_+$ in regression with standard scores
4. Entry radius of y in the sets given by x are then computed on each of the calibration points as: $\hat{r}\left(X_i,Y_i\right):=\inf\left\{t\in\mathcal{T}:Y_i\in R_t\left(X_i;\hat{A}\right)\right\}, i\in\mathrm{Cal}$

   $\hookrightarrow$ They inherit from data exchangeability and play the role of the conformity scores

$\Rightarrow$ Denote the set of entry radii $\mathcal{R}=\left\{\left(\hat{r}\left(X_i,Y_i\right)\right)_{i\in\mathrm{Cal}}\right\}\cup\{+\infty\}$

5. Compute the $1-\alpha$ quantile of these radii, noted $q_{1-\alpha}\left(\mathcal{R}\right)$

Another view on SCP (nested sets, Gupta et al., 2022)

1. Randomly split the training data into a proper training set (size $\#\mathrm{Tr}$) and a calibration set (size $\#\mathrm{Cal}$)
2. Get $\hat{A}$ *by training the algorithm* $\mathcal{A}$ *on the proper training set*
3. Build a sequence of nested predictive sets taking their values in $\mathcal{Y}$: $\left(R_t\left(\cdot;\hat{A}\right)\right)_{t\in\mathcal{T}}$ for some $\mathcal{T}\subseteq\mathbb{R}$, such that for $t\leq t'$: $R_t\left(x;\hat{A}\right)\subseteq R_{t'}\left(x;\hat{A}\right)$

   Ex 1: $R_t\left(\cdot;\hat{\mu}\right)\equiv\left[\hat{\mu}(\cdot)\pm t\right]$ and $\mathcal{T}=\mathbb{R}_+$ in regression with standard scores
4. Entry radius of y in the sets given by x are then computed on each of the calibration points as: $\hat{r}\left(X_i,Y_i\right):=\inf\left\{t\in\mathcal{T}:Y_i\in R_t\left(X_i;\hat{A}\right)\right\}, i\in\mathrm{Cal}$

   $\hookrightarrow$ They inherit from data exchangeability and play the role of the conformity scores

$\Rightarrow$ Denote the set of entry radii $\mathcal{R}=\left\{\left(\hat{r}\left(X_i,Y_i\right)\right)_{i\in\mathrm{Cal}}\right\}\cup\{+\infty\}$

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

$$\widehat{C}_\alpha(X_{n+1}):=R_{q_{1-\alpha}(\mathcal{R})}\left(x;\hat{A}\right)=\left\{y\in\mathcal{Y}\text{ such that }\hat{r}\left(x,y\right)\leq q_{1-\alpha}\left(\mathcal{R}\right)\right\}$$

Some examples of nested sets (Gupta et al., 2022)

Example (Nested sets for the absolute value of the mean-regression residuals).

$$s\left(x, y; \hat{\mu}\right) = |y - \hat{\mu}(x)| \quad \Longleftrightarrow \quad \begin{cases} R_t\left(\cdot; \hat{\mu}\right) \equiv [\hat{\mu}(\cdot) \pm t] \\ \mathcal{T} = \mathbb{R}_+ \end{cases}$$

Some examples of nested sets (Gupta et al., 2022)

Example (Nested sets for the absolute value of the mean-regression residuals).

$$s\left(x, y; \hat{\mu}\right) = |y - \hat{\mu}(x)| \iff \begin{cases} R_t\left(\cdot; \hat{\mu}\right) \equiv [\hat{\mu}(\cdot) \pm t] \\ \mathcal{T} = \mathbb{R}_+ \end{cases}$$

Example (Nested sets for the absolute value of the mean-regression residuals).

$$s\left(x, y; \hat{\mu}, \hat{\rho}\right) = \frac{|y - \hat{\mu}(x)|}{\hat{\rho}(x)} \iff \begin{cases} R_t\left(\cdot; \hat{\mu}, \hat{\rho}\right) \equiv [\hat{\mu}(\cdot) \pm t\hat{\rho}(x)] \\ \mathcal{T} = \mathbb{R}_+ \end{cases}$$

Some examples of nested sets (Gupta et al., 2022)

Example (Nested sets for the absolute value of the mean-regression residuals).

$$s\left(x, y; \hat{\mu}\right) = |y - \hat{\mu}(x)| \quad \Longleftrightarrow \quad \begin{cases} R_t\left(\cdot; \hat{\mu}\right) \equiv [\hat{\mu}(\cdot) \pm t] \\ \mathcal{T} = \mathbb{R}_+ \end{cases}$$

Example (Nested sets for the absolute value of the mean-regression residuals).

$$s\left(x, y; \hat{\mu}, \hat{\rho}\right) = \frac{|y - \hat{\mu}(x)|}{\hat{\rho}(x)} \quad \Longleftrightarrow \quad \begin{cases} R_t\left(\cdot; \hat{\mu}, \hat{\rho}\right) \equiv [\hat{\mu}(\cdot) \pm t\hat{\rho}(x)] \\ \mathcal{T} = \mathbb{R}_+ \end{cases}$$

Example (Nested sets for CQR).

$$\begin{aligned} & s\left(x, y; (\widehat{\mathrm{QR}}_{\mathrm{lower}}, \widehat{\mathrm{QR}}_{\mathrm{upper}})\right) \\ &= \max\left(\widehat{\mathrm{QR}}_{\mathrm{lower}}(x) - y, y - \widehat{\mathrm{QR}}_{\mathrm{upper}}(x)\right) \end{aligned} \quad \Longleftrightarrow \quad \begin{cases} R_t\left(\cdot; (\widehat{\mathrm{QR}}_{\mathrm{lower}}, \widehat{\mathrm{QR}}_{\mathrm{upper}})\right) \\ \equiv \left[\widehat{\mathrm{QR}}_{\mathrm{lower}}(\cdot) - t; \widehat{\mathrm{QR}}_{\mathrm{upper}}(\cdot) + t\right] \\ \mathcal{T} = \mathbb{R} \end{cases}$$

Where are we now?

1. On exchangeability (theory)
2. Split conformal prediction (methods) (theory) (practical session)
3. Towards conditional coverage? (practical session) (theory) (case studies)
4. Beyond exchangeability (methods) (case studies)

Where are we now?

1. On exchangeability (theory)
2. Split conformal prediction (methods) (theory) (practical session)
3. Towards conditional coverage? (practical session) (theory) (case studies)
4. Beyond exchangeability (methods) (case studies)
5. Computational and statistical trade-offs (methods) (theory)

Where are we now?

1. On exchangeability (theory)
2. Split conformal prediction (methods) (theory) (practical session)
3. Towards conditional coverage? (practical session) (theory) (case studies)
4. Beyond exchangeability (methods) (case studies)
5. Computational and statistical trade-offs (methods) (theory)
6. Handling missing data (methods)

Avoiding data splitting: full conformal and out-of-bags approaches

Full Conformal Prediction

Jackknife+

Handling missing data

Avoiding data splitting: full conformal and out-of-bags approaches

Full Conformal Prediction

Jackknife+

Handling missing data

Splitting the data might not be desired

SCP suffers from data splitting:

- lower statistical efficiency (lower model accuracy and higher predictive set size)
- higher statistical variability

Splitting the data might not be desired

SCP suffers from data splitting:

- lower statistical efficiency (lower model accuracy and higher predictive set size)
- higher statistical variability

Can we avoid splitting the data set?

The naive idea does not enjoy valid coverage (even empirically)

- A naive idea:
  - Get $\hat{A}$ by training the algorithm $\mathcal{A}$ on $\{(X_1, Y_1), \ldots, (X_n, Y_n)\}$.

The naive idea does not enjoy valid coverage (even empirically)

- A naive idea:
  - Get $\hat{A}$ by training the algorithm $\mathcal{A}$ on $\{(X_1, Y_1), \ldots, (X_n, Y_n)\}$.
  - compute the empirical quantile $q_{1-\alpha}(\mathcal{S})$ of the set of scores

$$\mathcal{S} = \left\{ \mathrm{s}\left(\hat{A}\left(X_i\right), Y_i\right)\right\}_{i=1}^{n} \cup \{\infty\}.$$

The naive idea does not enjoy valid coverage (even empirically)

- A naive idea:
  - Get $\hat{A}$ by training the algorithm $\mathcal{A}$ on $\{(X_1, Y_1), \ldots, (X_n, Y_n)\}$.
  - compute the empirical quantile $q_{1-\alpha}(\mathcal{S})$ of the set of scores

$$\mathcal{S} = \left\{ \mathsf{s}\left(\hat{A}\left(X_i\right), Y_i\right)\right\}_{i=1}^{n} \cup \{\infty\}.$$

  - output the set $\left\{y \text{ such that } \mathsf{s}\left(\hat{A}\left(X_{n+1}\right), y\right) \leq q_{1-\alpha}(\mathcal{S})\right\}$.

The naive idea does not enjoy valid coverage (even empirically)

- A naive idea:
  - Get $\hat{A}$ by training the algorithm $\mathcal{A}$ on $\{(X_1, Y_1), \ldots, (X_n, Y_n)\}$.
  - compute the empirical quantile $q_{1-\alpha}(\mathcal{S})$ of the set of scores

$$\mathcal{S} = \left\{ s\left(\hat{A}\left(X_i\right), Y_i\right) \right\}_{i=1}^{n} \cup \{\infty\}.$$

  - output the set $\left\{ y \text{ such that } s\left(\hat{A}\left(X_{n+1}\right), y\right) \leq q_{1-\alpha}(\mathcal{S}) \right\}$.

✗ $\hat{A}$ obtained w. the training set $\{(X_1, Y_1), \ldots, (X_n, Y_n)\}$ but not X_{n+1}.

Example ("Naive Idea" sets with an interpolating algorithm)**.**

Assume $\mathcal{A}$ interpolates:

- $\hat{A} = \mathcal{A}\left((x_1, y_1), \ldots, (x_n, y_n)\right)$
- $\hat{A}(x_k) - y_k = 0$ for any $k \in [\![1, n]\!]$

$\Rightarrow$ Naive method above *(with MAE score functions)* outputs $\{\hat{A}\left(X_{n+1}\right)\}$ (a single point) for any new test point!

Full Conformal Prediction[9] does not discard training points!

- Full *(or transductive)* Conformal Prediction
  - avoids data splitting

[9] Vovk et al. (2005), *Algorithmic Learning in a Random World*

Full Conformal Prediction[9] does not discard training points!

- Full *(or transductive)* Conformal Prediction
  - avoids data splitting
  - at the cost of many more model fits

[9] Vovk et al. (2005), *Algorithmic Learning in a Random World*

Full Conformal Prediction[9] does not discard training points!

- Full *(or transductive)* Conformal Prediction
  - avoids data splitting
  - at the cost of many more model fits
- **Idea**: the most probable labels Y_{n+1} live in $\mathcal{Y}$, and have a low enough conformity score. By looping over all possible $y \in \mathcal{Y}$, the ones leading to the smallest conformity scores will be found.

[9] Vovk et al. (2005), *Algorithmic Learning in a Random World*

Full Conformal Prediction (CP): recovering exchangeability

For any candidate (X_{n+1}, y):

1. Get $\hat{A}_y$ by training $\mathcal{A}$ on $\{(X_1, Y_1), \ldots, (X_n, Y_n)\} \cup \{(X_{n+1}, y)\}$

Full Conformal Prediction (CP): recovering exchangeability

For any candidate (X_{n+1}, y):

1. Get $\hat{A}_y$ by training $\mathcal{A}$ on $\{(X_1, Y_1), \ldots, (X_n, Y_n)\} \cup \{(X_{n+1}, y)\}$
2. Obtain a set of training scores
$$\mathcal{S}_y^{(\text{train})} = \left\{ \mathsf{s}(\hat{A}_y(X_i), Y_i) \right\}_{i=1}^{n} \cup \{ \mathsf{s}(\hat{A}_y(X_{n+1}), y)\}$$
and compute their $1-\alpha$ empirical quantile $q_{1-\alpha}\left(\mathcal{S}_y^{(\text{train})}\right)$

Full Conformal Prediction (CP): recovering exchangeability

For any candidate (X_{n+1}, y):

1. Get $\hat{A}_y$ by training $\mathcal{A}$ on $\{(X_1, Y_1), \dots, (X_n, Y_n)\} \cup \{(X_{n+1}, y)\}$
2. Obtain a set of training scores
$$\mathcal{S}_y^{(\text{train})} = \left\{ \mathrm{s}(\hat{A}_y(X_i), Y_i) \right\}_{i=1}^{n} \cup \{ \mathrm{s}(\hat{A}_y(X_{n+1}), y)\}$$
and compute their $1 - \alpha$ empirical quantile $q_{1-\alpha}\left(\mathcal{S}_y^{(\text{train})}\right)$

Output the set $\left\{ y \text{ such that } \mathrm{s}\left(\hat{A}_y\left(X_{n+1}\right), y\right) \leq q_{1-\alpha}\left(\mathcal{S}_y^{(\text{train})}\right)\right\}$.

Full Conformal Prediction (CP): recovering exchangeability

For any candidate (X_{n+1}, y):

1. Get $\hat{A}_y$ by training $\mathcal{A}$ on $\{(X_1, Y_1), \ldots, (X_n, Y_n)\} \cup \{(X_{n+1}, y)\}$
2. Obtain a set of training scores
$$\mathcal{S}_y^{(\text{train})} = \left\{ \boxed{s}(\hat{A}_y(X_i), Y_i) \right\}_{i=1}^{n} \cup \{ \boxed{s}(\hat{A}_y(X_{n+1}), y)\}$$
and compute their $1 - \alpha$ empirical quantile $q_{1-\alpha}\left(\mathcal{S}_y^{(\text{train})}\right)$

Output the set $\left\{ y \text{ such that } \boxed{s}\left(\hat{A}_y\left(X_{n+1}\right), y\right) \leq q_{1-\alpha}\left(\mathcal{S}_y^{(\text{train})}\right)\right\}$.

✓ Test point treated in the same way than train points

Full Conformal Prediction (CP): recovering exchangeability

For any candidate (X_{n+1}, y):

1. Get $\hat{A}_y$ by training $\mathcal{A}$ on $\{(X_1, Y_1), \ldots, (X_n, Y_n)\} \cup \{(X_{n+1}, y)\}$
2. Obtain a set of training scores
$$\mathcal{S}_y^{(\text{train})} = \left\{ \mathrm{s}(\hat{A}_y(X_i), Y_i) \right\}_{i=1}^{n} \cup \{ \mathrm{s}(\hat{A}_y(X_{n+1}), y)\}$$
and compute their $1 - \alpha$ empirical quantile $q_{1-\alpha}\left(\mathcal{S}_y^{(\text{train})}\right)$

Output the set $\left\{ y \text{ such that } \mathrm{s}\left(\hat{A}_y\left(X_{n+1}\right), y\right) \leq q_{1-\alpha}\left(\mathcal{S}_y^{(\text{train})}\right)\right\}$.

✓ Test point treated in the same way than train points

✓ Any score works

Full Conformal Prediction (CP): recovering exchangeability

For any candidate (X_{n+1}, y):

1. Get $\hat{A}_y$ by training $\mathcal{A}$ on $\{(X_1, Y_1), \ldots, (X_n, Y_n)\} \cup \{(X_{n+1}, y)\}$
2. Obtain a set of training scores
$$\mathcal{S}_y^{(\text{train})} = \left\{ \boxed{\mathrm{s}}(\hat{A}_y(X_i), Y_i) \right\}_{i=1}^{n} \cup \{ \boxed{\mathrm{s}}(\hat{A}_y(X_{n+1}), y)\}$$
and compute their $1-\alpha$ empirical quantile $q_{1-\alpha}\left(\mathcal{S}_y^{(\text{train})}\right)$

Output the set $\left\{ y \text{ such that } \boxed{\mathrm{s}}\left(\hat{A}_y\left(X_{n+1}\right), y\right) \leq q_{1-\alpha}\left(\mathcal{S}_y^{(\text{train})}\right) \right\}$.

- ✓ Test point treated in the same way than train points
- ✓ Any score works
- ✗ Computationally costly

Definition (Symmetrical algorithm).

A deterministic algorithm $\mathcal{A} : (U_1, \ldots, U_n) \mapsto \hat{A}$ is symmetric if for any permutation σ of $[\![1, n]\!]$: $\mathcal{A}(U_1, \ldots, U_n) \overset{\text{a.s.}}{=} \mathcal{A}\left(U_{\sigma(1)}, \ldots, U_{\sigma(n)}\right)$.

Full CP: theoretical foundation

Definition (Symmetrical algorithm).

A deterministic algorithm $\mathcal{A} : (U_1, \ldots, U_n) \mapsto \hat{A}$ is symmetric if for any permutation σ of $[\![1, n]\!]$: $\mathcal{A}\,(U_1, \ldots, U_n) \overset{\text{a.s.}}{=} \mathcal{A}\left(U_{\sigma(1)}, \ldots, U_{\sigma(n)}\right)$.

Lemma (Exchangeable scores).

If the algorithm $\mathcal{A} : (U_1, \ldots, U_n) \mapsto \hat{A}$ is symmetric, and $(X_i, Y_i)_{i=1}^{n+1}$ are exchangeable, then $S_1, \ldots, S_{n+1}$ are exchangeable, with

$$S_i := \mathrm{s}(\hat{A}_{Y_{n+1}}(X_i), Y_i).$$

Definition (Symmetrical algorithm).

A deterministic algorithm $\mathcal{A} : (U_1, \ldots, U_n) \mapsto \hat{A}$ is symmetric if for any permutation σ of $[\![1, n]\!]$: $\mathcal{A}(U_1, \ldots, U_n) \overset{\text{a.s.}}{=} \mathcal{A}\left(U_{\sigma(1)}, \ldots, U_{\sigma(n)}\right)$.

Lemma (Exchangeable scores).

If the algorithm $\mathcal{A} : (U_1, \ldots, U_n) \mapsto \hat{A}$ is symmetric, and $(X_i, Y_i)_{i=1}^{n+1}$ are exchangeable, then $S_1, \ldots, S_{n+1}$ are exchangeable, with

$$S_i := \boxed{\mathrm{s}}(\hat{A}_{Y_{n+1}}(X_i), Y_i).$$

Moreover

$$\begin{aligned}
Y_{n+1} \in \widehat{C}_\alpha^{\text{Full}}(X_{n+1}) &:= \left\{ y \text{ such that } \boxed{\mathrm{s}} \left(\hat{A}_y \left(X_{n+1} \right), y \right) \leq q_{1-\alpha} \left(\mathcal{S}_y^{(\text{train})} \right) \right\} \\
&\Leftrightarrow \boxed{\mathrm{s}} \left(\hat{A}_{Y_{n+1}} \left(X_{n+1} \right), Y_{n+1} \right) \leq q_{1-\alpha} \left(\mathcal{S}_{Y_{n+1}}^{(\text{train})} \right) \\
&\Leftrightarrow S_{n+1} \leq q_{1-\alpha}(S_1, \ldots, S_n, S_{n+1}) \; !
\end{aligned}$$

Full CP: theoretical guarantees

Full CP enjoys finite sample guarantees proved in Vovk et al. (2005).

Theorem (Marginal validity of Full CP Vovk et al. (2005)).

Suppose that

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

(ii) the algorithm $\mathcal{A}$ is symmetric.

Full CP applied on $(X_i, Y_i)_{i=1}^{n} \cup \{X_{n+1}\}$ outputs $\widehat{C}_\alpha(\cdot)$ such that:

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

Additionally, if the scores are a.s. distinct:

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

Full CP: theoretical guarantees

Full CP enjoys finite sample guarantees proved in Vovk et al. (2005).

Theorem (Marginal validity of Full CP Vovk et al. (2005))**.**

Suppose that

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

(ii) the algorithm $\mathcal{A}$ is symmetric.

Full CP applied on $(X_i, Y_i)_{i=1}^{n} \cup \{X_{n+1}\}$ outputs $\widehat{C}_\alpha(\cdot)$ such that:

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

Additionally, if the scores are a.s. distinct:

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

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

Interpolation regime

Example (FCP sets with an interpolating algorithm)**.**

Assume $\mathcal{A}$ interpolates:

- $\hat{A} = \mathcal{A}\left((x_1, y_1), \ldots, (x_{n+1}, y_{n+1})\right)$
- $\hat{A}(x_k) - y_k = 0$ for any $k \in [\![1, n+1]\!]$

Interpolation regime

Example (FCP sets with an interpolating algorithm).

Assume $\mathcal{A}$ interpolates:

- $\hat{A} = \mathcal{A}\left((x_1, y_1), \ldots, (x_{n+1}, y_{n+1})\right)$
- $\hat{A}(x_k) - y_k = 0$ for any $k \in [\![1, n+1]\!]$

$\Rightarrow$ Full Conformal Prediction *(with standard score functions)* outputs $\mathcal{Y}$ (the whole label space) for any new test point!

Split Conformal Prediction is a special case of Full Conformal Prediction

- Set $\hat{A}_y \equiv \hat{A}$, constant, independent of $\{(X_1, Y_1), \ldots, (X_n, Y_n)\} \cup \{(X_{n+1}, y)\}$
- Then, running Full Conformal Prediction corresponds to Split Conformal Prediction with $\#\mathrm{Cal} = n$.

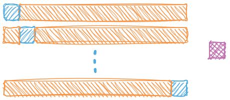
Avoiding data splitting: full conformal and out-of-bags approaches

Full Conformal Prediction

Jackknife+

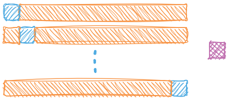
Handling missing data

Jackknife: the naive idea does not enjoy valid coverage



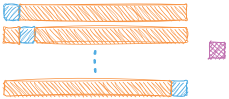
- Based on leave-one-out (LOO) residuals
- $\mathcal{D}_n = \{(X_1, Y_1), \ldots, (X_n, Y_n)\}$ training data
- Get $\hat{A}_{-i}$ by training $\mathcal{A}$ on $\mathcal{D}_n \setminus (X_i, Y_i)$

Jackknife: the naive idea does not enjoy valid coverage



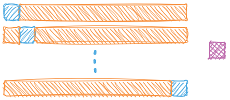
- Based on leave-one-out (LOO) residuals
- $\mathcal{D}_n = \{(X_1, Y_1), \dots, (X_n, Y_n)\}$ training data
- Get $\hat{A}_{-i}$ by training $\mathcal{A}$ on $\mathcal{D}_n \setminus (X_i, Y_i)$
- LOO scores $\mathcal{S} = \left\{ |\hat{A}_{-i}(X_i) - Y_i| \right\}_i \cup \{+\infty\}$ (in standard mean regression)

Jackknife: the naive idea does not enjoy valid coverage



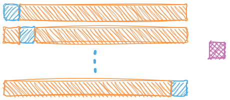
- Based on leave-one-out (LOO) residuals
- $\mathcal{D}_n = \{(X_1, Y_1), \ldots, (X_n, Y_n)\}$ training data
- Get $\hat{A}_{-i}$ by training $\mathcal{A}$ on $\mathcal{D}_n \setminus (X_i, Y_i)$
- LOO scores $\mathcal{S} = \left\{ |\hat{A}_{-i}(X_i) - Y_i| \right\}_i \cup \{+\infty\}$ (in standard mean regression)
- Get $\hat{A}$ by training $\mathcal{A}$ on $\mathcal{D}_n$

Jackknife: the naive idea does not enjoy valid coverage



- Based on leave-one-out (LOO) residuals
- $\mathcal{D}_n = \{(X_1, Y_1), \ldots, (X_n, Y_n)\}$ training data
- Get $\hat{A}_{-i}$ by training $\mathcal{A}$ 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 mean regression)
- Get $\hat{A}$ by training $\mathcal{A}$ on $\mathcal{D}_n$
- Build the predictive interval: $\left[\hat{A}(X_{n+1}) \pm q_{1-\alpha}(\mathcal{S}) \right]$

Jackknife: the naive idea does not enjoy valid coverage



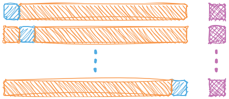
- Based on leave-one-out (LOO) residuals
- $\mathcal{D}_n = \{(X_1, Y_1), \ldots, (X_n, Y_n)\}$ training data
- Get $\hat{A}_{-i}$ by training $\mathcal{A}$ on $\mathcal{D}_n \setminus (X_i, Y_i)$
- LOO scores $\mathcal{S} = \left\{ |\hat{A}_{-i}(X_i) - Y_i| \right\}_i \cup \{+\infty\}$ (in standard mean regression)
- Get $\hat{A}$ by training $\mathcal{A}$ 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 $(\hat{A}_{-i})_i$, without assuming a form of **stability** on $\mathcal{A}$.

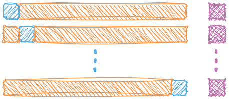
Jackknife+ (Barber et al., 2021)

Jackknife+ (Barber et al., 2021)



- Based on leave-one-out (LOO) residuals
- $\mathcal{D}_n = \{(X_1, Y_1), \ldots, (X_n, Y_n)\}$ training data
- Get $\hat{A}_{-i}$ by training $\mathcal{A}$ on $\mathcal{D}_n \setminus (X_i, Y_i)$

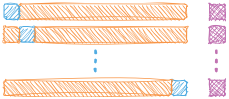
Jackknife+ (Barber et al., 2021)



- Based on leave-one-out (LOO) residuals
- $\mathcal{D}_n = \{(X_1, Y_1), \ldots, (X_n, Y_n)\}$ training data
- Get $\hat{A}_{-i}$ by training $\mathcal{A}$ on $\mathcal{D}_n \setminus (X_i, Y_i)$
- LOO predictions / predictive intervals
  $\mathcal{S}_{\text{up/down}} = \left\{ \hat{A}_{-i}(X_{n+1}) \pm |\hat{A}_{-i}(X_i) - Y_i| \right\}_i \cup \{\pm\infty\}$

(in standard mean regression)

Jackknife+ (Barber et al., 2021)

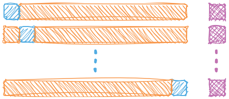


- Based on leave-one-out (LOO) residuals
- $\mathcal{D}_n = \{(X_1, Y_1), \ldots, (X_n, Y_n)\}$ training data
- Get $\hat{A}_{-i}$ by training $\mathcal{A}$ on $\mathcal{D}_n \setminus (X_i, Y_i)$
- LOO predictions / predictive intervals
  $\mathcal{S}_{\text{up/down}} = \left\{ \hat{A}_{-i}(X_{n+1}) \pm |\hat{A}_{-i}(X_i) - Y_i| \right\}_i \cup \{\pm\infty\}$

  (in standard mean regression)
- Build the predictive interval: $[q_{\alpha,\inf}(\mathcal{S}_{\text{down}}); q_{1-\alpha}(\mathcal{S}_{\text{up}})]$

Recall $q_{\beta,\inf}(X_1, \ldots, X_n) := \lfloor \beta \times n \rfloor$ smallest value of $(X_1, \ldots, X_n)$

Jackknife+ (Barber et al., 2021)



- Based on leave-one-out (LOO) residuals
- $\mathcal{D}_n = \{(X_1, Y_1), \ldots, (X_n, Y_n)\}$ training data
- Get $\hat{A}_{-i}$ by training $\mathcal{A}$ on $\mathcal{D}_n \setminus (X_i, Y_i)$
- LOO predictions / predictive intervals

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

(in standard mean regression)

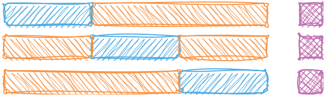
- Build the predictive interval: $[q_{\alpha,\inf}(\mathcal{S}_{\text{down}}); q_{1-\alpha}(\mathcal{S}_{\text{up}})]$

Theorem (Marginal validity of Jackknife+ Barber et al. (2021))**.**

If $\mathcal{D}_n \cup (X_{n+1}, Y_{n+1})$ are exchangeable and $\mathcal{A}$ is symmetric: $\mathbb{P}(Y_{n+1} \in \widehat{C}_\alpha(X_{n+1})) \geq 1 - 2\alpha$.

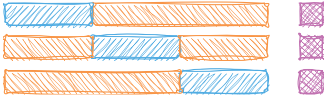
Recall $q_{\beta,\inf}(X_1, \ldots, X_n) := \lfloor \beta \times n \rfloor$ smallest value of $(X_1, \ldots, X_n)$

CV+ Barber et al. (2021) (see also cross-conformal predictors: Vovk, 2015)



- Based on cross-validation residuals
- $\mathcal{D}_n = \{(X_1, Y_1), \ldots, (X_n, Y_n)\}$ training data
- Split $\mathcal{D}_n$ into K folds $F_1, \ldots, F_K$
- Get $\hat{A}_{-F_k}$ by training $\mathcal{A}$ on $\mathcal{D}_n \setminus F_k$

CV+ Barber et al. (2021) (see also cross-conformal predictors: Vovk, 2015)

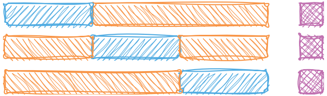


- Based on cross-validation residuals
- $\mathcal{D}_n = \{(X_1, Y_1), \dots, (X_n, Y_n)\}$ training data
- Split $\mathcal{D}_n$ into K folds $F_1, \dots, F_K$
- Get $\hat{A}_{-F_k}$ by training $\mathcal{A}$ on $\mathcal{D}_n \setminus F_k$
- Cross-val predictions / predictive intervals

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

(in standard mean regression)

CV+ Barber et al. (2021) (see also cross-conformal predictors: Vovk, 2015)



- Based on cross-validation residuals
- $\mathcal{D}_n = \{(X_1, Y_1), \dots, (X_n, Y_n)\}$ training data
- Split $\mathcal{D}_n$ into K folds $F_1, \dots, F_K$
- Get $\hat{A}_{-F_k}$ by training $\mathcal{A}$ on $\mathcal{D}_n \setminus F_k$
- Cross-val predictions / predictive intervals

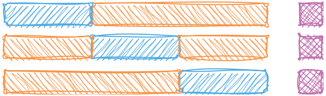
$$\mathcal{S}_{\text{up/down}} = \left\{ \left\{ \hat{A}_{-F_k}(X_{n+1}) \pm \left| \hat{A}_{-F_k}(X_i) - Y_i \right| \right\}_{i \in F_k} \right\}_k \cup \{\pm\infty\}$$

(in standard mean regression)

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

Recall $q_{\beta,\inf}(X_1, \dots, X_n) := \lfloor \beta \times n \rfloor$ smallest value of $(X_1, \dots, X_n)$

CV+ Barber et al. (2021) (see also cross-conformal predictors: Vovk, 2015)



- Based on cross-validation residuals
- $\mathcal{D}_n = \{(X_1, Y_1), \ldots, (X_n, Y_n)\}$ training data
- Split $\mathcal{D}_n$ into K folds $F_1, \ldots, F_K$
- Get $\hat{A}_{-F_k}$ by training $\mathcal{A}$ on $\mathcal{D}_n \setminus F_k$
- Cross-val predictions / predictive intervals

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

(in standard mean regression)

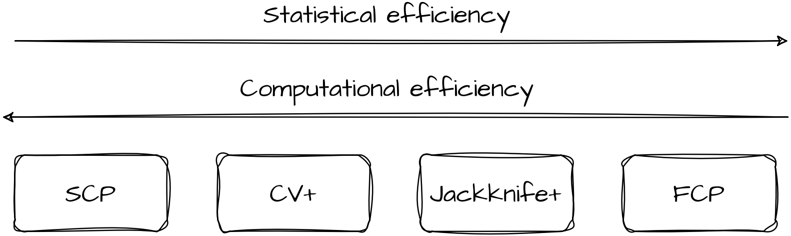
- Build the predictive interval: $[q_{\alpha,\inf}(\mathcal{S}_{\text{down}}); q_{1-\alpha}(\mathcal{S}_{\text{up}})]$

Theorem (Marginal validity of CV+ Barber et al. (2021)).

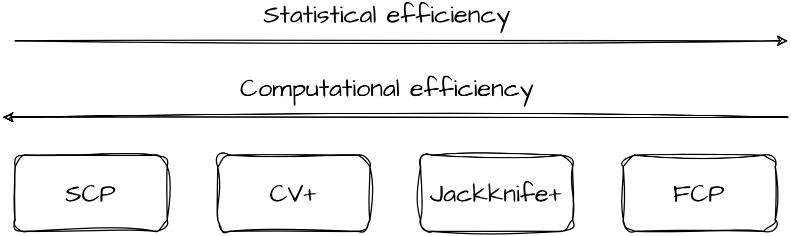
If $\mathcal{D}_n \cup (X_{n+1}, Y_{n+1})$ are exchangeable and $\mathcal{A}$ is symmetric: $\mathbb{P}(Y_{n+1} \in \widehat{C}_\alpha(X_{n+1})) \geq 1 - 2\alpha - \min\left(\frac{2(1-1/K)}{n/K+1}, \frac{1-K/n}{K+1}\right) \geq 1 - 2\alpha - \sqrt{2/n}$.

Recall $q_{\beta,\inf}(X_1, \ldots, X_n) := \lfloor \beta \times n \rfloor$ smallest value of $(X_1, \ldots, X_n)$

General overview

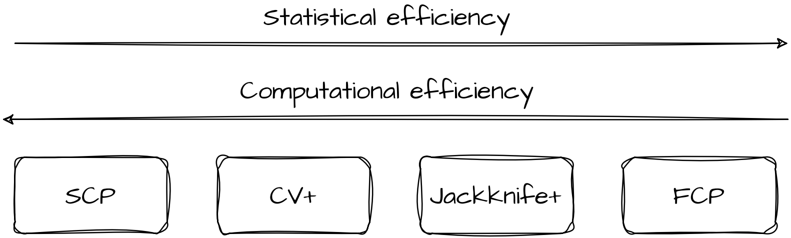


General overview



- Generalized framework encapsulating out-of-sample methods: Nested CP (Gupta et al., 2022) $\rightarrow$ extends *JK+/CV+ for any score.*

General overview



- Generalized framework encapsulating out-of-sample methods: Nested CP (Gupta et al., 2022) $\rightarrow$ extends *JK+/CV+ for any score.*
- Accelerating FCP: Nouretdinov et al. (2001); Lei (2019); Ndiaye and Takeuchi (2019); Cherubin et al. (2021); Ndiaye and Takeuchi (2022); Ndiaye (2022)

Non exhaustive references.

Avoiding data splitting: full conformal and out-of-bags approaches

Handling missing data

- Supervised learning setting with missing covariates
- Goals and challenges for predictive uncertainty quantification
- Is MCV a too lofty goal?!
- Achieving MCV under $M \perp\!\!\!\perp X$ and $Y \perp\!\!\!\perp M \mid X$
- Experimental results

A collaboration

Yaniv Romano

Technion - Israel Institute of Technology

Julie Josse

PreMeDICaL

INRIA

Aymeric Dieuleveut

École Polytechnique

- *Predictive Uncertainty Quantification with Missing Covariates*, 2024
- *Conformal Prediction with Missing Values*, ICML 2023

TraumaBase®: decision support for trauma patients

- 30 hospitals
- More than 30 000 trauma patients
- 4 000 new patients per year
- 250 continuous and categorical variables
  $\hookrightarrow$ Many useful statistical tasks

TraumaBase®: decision support for trauma patients

- 30 hospitals
- More than 30 000 trauma patients
- 4 000 new patients per year
- 250 continuous and categorical variables
  $\hookrightarrow$ Many useful statistical tasks

Predict the level of blood platelets upon arrival at hospital, given 7 pre-hospital features.

TraumaBase®: decision support for trauma patients

- 30 hospitals
- More than 30 000 trauma patients
- 4 000 new patients per year
- 250 continuous and categorical variables
  $\hookrightarrow$ Many useful statistical tasks

Predict the level of blood platelets upon arrival at hospital, given 7 pre-hospital features.

These covariates are not always observed.

Missing values are ubiquitous and challenging

Data: $\left(X^{(k)}, \qquad Y^{(k)}\right)_{k=1}^{n}$

Y	X_1	X_2	X_3
22	5	6	3
19	6	8	NA
19	5	3	6
7	NA	9	NA
13	4	9	0
20	NA	NA	1
9	8	NA	4

Missing values are ubiquitous and challenging

Data: $\left(X^{(k)}, M^{(k)}, Y^{(k)}\right)_{k=1}^{n}$

Y	X_1	X_2	X_3
22	5	6	3
19	6	8	NA
19	5	3	6
7	NA	9	NA
13	4	9	0
20	NA	NA	1
9	8	NA	4

Mask $M =$

$(M_1$	M_2	$M_3)$
0	0	0
0	0	1
0	0	0
1	0	1
0	0	0
1	1	0
0	1	0

Missing values are ubiquitous and challenging

Data: $\left(X^{(k)}, M^{(k)}, Y^{(k)}\right)_{k=1}^{n}$

Y	X_1	X_2	X_3
22	5	6	3
19	6	8	NA
19	5	3	6
7	NA	9	NA
13	4	9	0
20	NA	NA	1
9	8	NA	4

Mask $M = (M_1 \; M_2 \; M_3)$

M_1	M_2	M_3
0	0	0
0	0	1
0	0	0
1	0	1
0	0	0
1	1	0
0	1	0

If each entry has a probability 0.01 of being missing:

$$d = 6 \rightarrow \approx 94\% \text{ of rows kept}$$

$$d = 300 \rightarrow \approx 5\% \text{ of rows kept}$$

Missing values are ubiquitous and challenging

Data: $\left(X^{(k)}, M^{(k)}, Y^{(k)}\right)_{k=1}^{n}$

Y	X_1	X_2	X_3
22	5	6	3
19	6	8	NA
19	5	3	6
7	NA	9	NA
13	4	9	0
20	NA	NA	1
9	8	NA	4

Mask $M =$

$(M_1$	M_2	$M_3)$
0	0	0
0	0	1
0	0	0
1	0	1
0	0	0
1	1	0
0	1	0

If each entry has a probability 0.01 of being missing:

$$d = 6 \rightarrow \; \approx 94\% \text{ of rows kept}$$

$$d = 300 \rightarrow \; \approx 5\% \text{ of rows kept}$$

One of the **ironies of Big Data** *is that missing data play an ever more significant role.*[1]

Missing values are ubiquitous and challenging

Data: $\left(X^{(k)}, M^{(k)}, Y^{(k)}\right)_{k=1}^{n}$

Y	X_1	X_2	X_3
22	5	6	3
19	6	8	NA
19	5	3	6
7	NA	9	NA
13	4	9	0
20	NA	NA	1
9	8	NA	4

Mask $M =$

$(M_1$	M_2	$M_3)$
0	0	0
0	0	1
0	0	0
1	0	1
0	0	0
1	1	0
0	1	0

$\hookrightarrow$ 2^d potential masks.

Missing values are ubiquitous and challenging

Data: $\left(X^{(k)}, M^{(k)}, Y^{(k)}\right)_{k=1}^{n}$

Y	X_1	X_2	X_3
22	5	6	3
19	6	8	NA
19	5	3	6
7	NA	9	NA
13	4	9	0
20	NA	NA	1
9	8	NA	4

Mask $M =$

$(M_1$	M_2	$M_3)$
0	0	0
0	0	1
0	0	0
1	0	1
0	0	0
1	1	0
0	1	0

$\hookrightarrow$ 2^d potential masks.

$\hookrightarrow$ M can depend on X or Y (depending on the missing mechanism[1]).

[1] Three mechanisms connecting X and M from **?**, *Inference and missing data*, Biometrika

Missing values are ubiquitous and challenging

Data: $\left(X^{(k)}, M^{(k)}, Y^{(k)}\right)_{k=1}^{n}$

Y	X_1	X_2	X_3
22	5	6	3
19	6	8	NA
19	5	3	6
7	NA	9	NA
13	4	9	0
20	NA	NA	1
9	8	NA	4

Mask $M =$

$(M_1$	M_2	$M_3)$
0	0	0
0	0	1
0	0	0
1	0	1
0	0	0
1	1	0
0	1	0

$\hookrightarrow$ 2^d potential masks.

$\hookrightarrow$ M can depend on X or Y (depending on the missing mechanism[1]).

- **Missing Completely At Random** (MCAR): $M \perp\!\!\!\perp X$

[1] Three mechanisms connecting X and M from **?**, *Inference and missing data*, Biometrika

Missing values are ubiquitous and challenging

Data: $\left(X^{(k)}, M^{(k)}, Y^{(k)}\right)_{k=1}^{n}$

Y	X_1	X_2	X_3
22	5	6	3
19	6	8	NA
19	5	3	6
7	NA	9	NA
13	4	9	0
20	NA	NA	1
9	8	NA	4

Mask $M =$

$(M_1$	M_2	$M_3)$
0	0	0
0	0	1
0	0	0
1	0	1
0	0	0
1	1	0
0	1	0

$\hookrightarrow$ 2^d potential masks.

$\hookrightarrow$ M can depend on X or Y (depending on the missing mechanism[1]).

- Missing Completely At Random (MCAR): $M \perp\!\!\!\perp X$
- **Missing At Random** (MAR): missingness depends on the observed variables

[1] Three mechanisms connecting X and M from **?**, *Inference and missing data*, Biometrika

Missing values are ubiquitous and challenging

Data: $\left(X^{(k)}, M^{(k)}, Y^{(k)}\right)_{k=1}^{n}$

Y	X_1	X_2	X_3
22	5	6	3
19	6	8	NA
19	5	3	6
7	NA	9	NA
13	4	9	0
20	NA	NA	1
9	8	NA	4

Mask $M =$

$(M_1$	M_2	$M_3)$
0	0	0
0	0	1
0	0	0
1	0	1
0	0	0
1	1	0
0	1	0

$\hookrightarrow$ 2^d potential masks.

$\hookrightarrow$ M can depend on X or Y (depending on the missing mechanism[1]).

- Missing Completely At Random (MCAR): $M \perp\!\!\!\perp X$
- Missing At Random (MAR): missingness depends on the observed variables
- **Missing Non At Random** (MNAR)

[1] Three mechanisms connecting X and M from **?**, *Inference and missing data*, Biometrika

Missing values are ubiquitous and challenging

Data: $\left(X^{(k)}, M^{(k)}, Y^{(k)}\right)_{k=1}^{n}$

Y	X_1	X_2	X_3
22	5	6	3
19	6	8	NA
19	5	3	6
7	NA	9	NA
13	4	9	0
20	NA	NA	1
9	8	NA	4

Mask $M =$

$(M_1$	M_2	$M_3)$
0	0	0
0	0	1
0	0	0
1	0	1
0	0	0
1	1	0
0	1	0

$\hookrightarrow$ 2^d potential masks.

$\hookrightarrow$ M can depend on X or Y (depending on the missing mechanism[1]).

- $Y \perp\!\!\!\perp M \,|X$

[1] Three mechanisms connecting X and M from **?**, *Inference and missing data*, Biometrika

Missing values are ubiquitous and challenging

Data: $\left(X^{(k)}, M^{(k)}, Y^{(k)}\right)_{k=1}^{n}$

Y	X_1	X_2	X_3
22	5	6	3
19	6	8	NA
19	5	3	6
7	NA	9	NA
13	4	9	0
20	NA	NA	1
9	8	NA	4

Mask $M =$

$(M_1$	M_2	$M_3)$
0	0	0
0	0	1
0	0	0
1	0	1
0	0	0
1	1	0
0	1	0

$\hookrightarrow$ 2^d potential masks.

$\hookrightarrow$ M can depend on X or Y (depending on the missing mechanism[1]).

- $Y \not\perp\!\!\!\perp M \mid X$

[1] Three mechanisms connecting X and M from **?**, *Inference and missing data*, Biometrika

Missing values are ubiquitous and challenging

Data: $\left(X^{(k)}, M^{(k)}, Y^{(k)}\right)_{k=1}^{n}$

Y	X_1	X_2	X_3
22	5	6	3
19	6	8	NA
19	5	3	6
7	NA	9	NA
13	4	9	0
20	NA	NA	1
9	8	NA	4

Mask $M =$

$(M_1$	M_2	$M_3)$
0	0	0
0	0	1
0	0	0
1	0	1
0	0	0
1	1	0
0	1	0

$\hookrightarrow$ 2^d potential masks.

$\hookrightarrow$ M can depend on X or Y (depending on the missing mechanism[1]).

$\Rightarrow$ Statistical and computational challenges.

[1] Three mechanisms connecting X and M from **?**, *Inference and missing data*, Biometrika

Supervised learning with missing values: impute-then-predict

Impute-then-predict procedures are widely used.

Supervised learning with missing values: impute-then-predict

Impute-then-predict procedures are widely used.

1. Replace `NA` using an imputation function (e.g. the mean), noted ϕ.

$$
\begin{array}{c}
x^{(1)} \\ x^{(2)} \\ x^{(3)} \\ x^{(4)}
\end{array}
\begin{bmatrix}
-1 & -10 & 6 & 0 \\
4 & \texttt{NA} & -2 & 2 \\
5 & 1 & 2 & \texttt{NA} \\
0 & \texttt{NA} & \texttt{NA} & 1
\end{bmatrix}
\xrightarrow{\quad \phi \quad}
\begin{array}{c}
u^{(1)} \\ u^{(2)} \\ u^{(3)} \\ u^{(4)}
\end{array}
\begin{bmatrix}
-1 & -10 & 6 & 0 \\
4 & -4.5 & -2 & 2 \\
5 & 1 & 2 & 1 \\
0 & -4.5 & 3 & 1
\end{bmatrix}
$$

Supervised learning with missing values: impute-then-predict

Impute-then-predict procedures are widely used.

1. Replace NA using an imputation function (e.g. the mean), noted ϕ.

$$
\begin{array}{c}
x^{(1)} \\ x^{(2)} \\ x^{(3)} \\ x^{(4)}
\end{array}
\begin{array}{|c|c|c|c|}
\hline
-1 & -10 & 6 & 0 \\ \hline
4 & \texttt{NA} & -2 & 2 \\ \hline
5 & 1 & 2 & \texttt{NA} \\ \hline
0 & \texttt{NA} & \texttt{NA} & 1 \\ \hline
\end{array}
\xrightarrow{\quad\phi\quad}
\begin{array}{c}
u^{(1)} \\ u^{(2)} \\ u^{(3)} \\ u^{(4)}
\end{array}
\begin{array}{|c|c|c|c|}
\hline
-1 & -10 & 6 & 0 \\ \hline
4 & -4.5 & -2 & 2 \\ \hline
5 & 1 & 2 & 1 \\ \hline
0 & -4.5 & 3 & 1 \\ \hline
\end{array}
$$

2. Train your algorithm (Random Forest, Neural Nets, etc.) on the imputed data:

$$\left\{ \underbrace{\phi\left(X^{(k)}_{\mathrm{obs}(M^{(k)})}, M^{(k)}\right)}_{U^{(k)}=\text{imputed } X^{(k)}}, Y^{(k)} \right\}_{k=1}^{n}.$$

Supervised learning with missing values: impute-then-predict

Impute-then-predict procedures are widely used.

1. Replace `NA` using an imputation function (e.g. the mean), noted ϕ.

$$\begin{array}{c} x^{(1)} \\ x^{(2)} \\ x^{(3)} \\ x^{(4)} \end{array} \begin{bmatrix} -1 & -10 & 6 & 0 \\ 4 & \mathtt{NA} & -2 & 2 \\ 5 & 1 & 2 & \mathtt{NA} \\ 0 & \mathtt{NA} & \mathtt{NA} & 1 \end{bmatrix} \xrightarrow{\phi} \begin{array}{c} u^{(1)} \\ u^{(2)} \\ u^{(3)} \\ u^{(4)} \end{array} \begin{bmatrix} -1 & -10 & 6 & 0 \\ 4 & -4.5 & -2 & 2 \\ 5 & 1 & 2 & 1 \\ 0 & -4.5 & 3 & 1 \end{bmatrix}$$

2. Train your algorithm (Random Forest, Neural Nets, etc.) on the imputed data:

$$\left\{ \underbrace{\phi\left(X^{(k)}_{\mathrm{obs}(M^{(k)})}, M^{(k)}\right)}_{U^{(k)} = \text{imputed } X^{(k)}}, Y^{(k)} \right\}_{k=1}^{n}.$$

$\hookrightarrow$ we consider an impute-then-predict pipeline in this work.

Supervised learning with missing values: impute-then-predict, cont'd

✓ Le Morvan et al. (2021)[2] show that for **any deterministic imputation** and **universal learner** this procedure is **Bayes-consistent**.

[2] Le Morvan, Josse, Scornet & Varoquaux (2021), *What's a good imputation to predict with missing values?*, NeurIPS

Supervised learning with missing values: impute-then-predict, cont'd

- ✓ Le Morvan et al. (2021)[2] show that for any deterministic imputation and universal learner this procedure is Bayes-consistent.
- ✗ Ayme et al. (2022)[3] show that even for very **simple distributions** (linear model, Gaussian noise), this rate of convergence may suffer from **curse of dimensionality**.

[2] Le Morvan, Josse, Scornet & Varoquaux (2021), *What's a good imputation to predict with missing values?*, NeurIPS

[3] Ayme, Boyer, Dieuleveut & Scornet (2022), *Near-optimal rate of consistency for linear models with missing values*, ICML

Avoiding data splitting: full conformal and out-of-bags approaches

Handling missing data

- Supervised learning setting with missing covariates
- Goals and challenges for predictive uncertainty quantification
- Is MCV a too lofty goal?!
- Achieving MCV under $M \perp\!\!\!\perp X$ and $Y \perp\!\!\!\perp M \mid X$
- Experimental results

Goals of predictive uncertainty quantification with missing values

Goal: predict $Y^{(n+1)}$ with **confidence** $1-\alpha$, i.e. build the smallest $\mathcal{C}_\alpha$ such that:

Goals of predictive uncertainty quantification with missing values

Goal: predict $Y^{(n+1)}$ with **confidence** $1-\alpha$, i.e. build the smallest $\mathcal{C}_\alpha$ such that:

Definition (1. Marginal Validity (MV)).

$$\mathbb{P}\left\{Y^{(n+1)} \in \mathcal{C}_\alpha\left(X^{(n+1)}, M^{(n+1)}\right)\right\} \geq 1-\alpha. \qquad \text{(MV)}$$

Goals of predictive uncertainty quantification with missing values

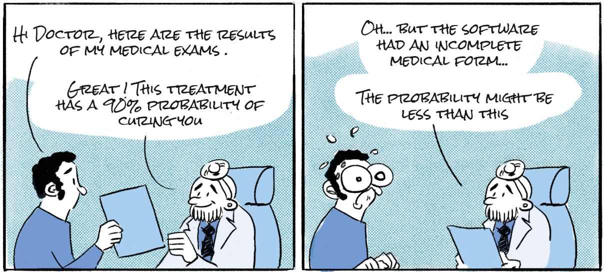
Goal: predict $Y^{(n+1)}$ with **confidence** $1-\alpha$, i.e. build the smallest $\mathcal{C}_\alpha$ such that:

Definition (1. Marginal Validity (MV)).

$$\mathbb{P}\left\{Y^{(n+1)} \in \mathcal{C}_\alpha\left(X^{(n+1)}, M^{(n+1)}\right)\right\} \geq 1-\alpha. \tag{MV}$$

Definition (2. Mask-Conditional-Validity (MCV)).

$$\mathbb{P}\left\{Y^{(n+1)} \in \mathcal{C}_\alpha\left(X^{(n+1)}, M^{(n+1)}\right) | M^{(n+1)}\right\} \overset{a.s.}{\geq} 1-\alpha. \tag{MCV}$$



Illustrations @theoremlinger

CP is marginally valid (MV) after imputation

Lemma (Exchangeability after imputation (Z., Dieuleveut, Josse and Romano, 2023)).

Assume $\left(X^{(k)}, M^{(k)}, Y^{(k)}\right)_{k=1}^{n}$ are i.i.d. (or exchangeable).
Then, for any missing mechanism, for almost all imputation function ϕ:
$\left(\phi\left(X^{(k)}_{\text{obs}(M^{(k)})}, M^{(k)}\right), Y^{(k)}\right)_{k=1}^{n}$ are **exchangeable**.

CP is marginally valid (MV) after imputation

Lemma (Exchangeability after imputation (Z., Dieuleveut, Josse and Romano, 2023)).

Assume $\left(X^{(k)}, M^{(k)}, Y^{(k)}\right)_{k=1}^{n}$ are i.i.d. (or exchangeable).
Then, for any missing mechanism, for almost all imputation function ϕ:
$\left(\phi\left(X^{(k)}_{\mathrm{obs}(M^{(k)})}, M^{(k)}\right), Y^{(k)}\right)_{k=1}^{n}$ are **exchangeable**.

$\Rightarrow$ Conformal Prediction (CP), applied on an imputed data set still enjoys marginal guarantees:

$$\mathbb{P}\left\{Y^{(n+1)} \in \widehat{C}_\alpha\left(X^{(n+1)}, M^{(n+1)}\right)\right\} \geq 1 - \alpha.$$

CP is marginally valid (MV) after imputation

Lemma (Exchangeability after imputation (Z., Dieuleveut, Josse and Romano, 2023)).

Assume $\left(X^{(k)}, M^{(k)}, Y^{(k)}\right)_{k=1}^{n}$ are i.i.d. (or exchangeable).
Then, for any missing mechanism, for almost all imputation function ϕ:
$\left(\phi\left(X^{(k)}_{\text{obs}(M^{(k)})}, M^{(k)}\right), Y^{(k)}\right)_{k=1}^{n}$ are **exchangeable**.

$\Rightarrow$ Conformal Prediction (CP), applied on an imputed data set still enjoys marginal guarantees:

$$\mathbb{P}\left\{Y^{(n+1)} \in \widehat{C}_\alpha\left(X^{(n+1)}, M^{(n+1)}\right)\right\} \geq 1 - \alpha.$$

CP is marginally valid (MV) after imputation

Lemma (Exchangeability after imputation (Z., Dieuleveut, Josse and Romano, 2023)).

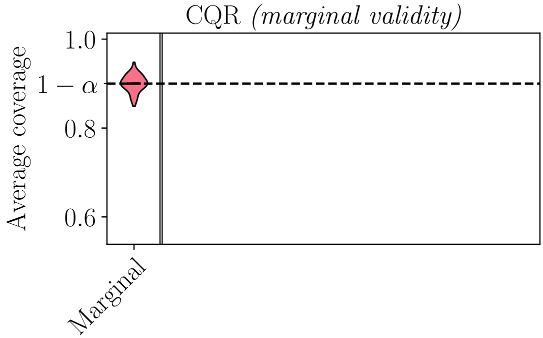
Assume $\left(X^{(k)}, M^{(k)}, Y^{(k)}\right)_{k=1}^{n}$ are i.i.d. (or exchangeable).
Then, for any missing mechanism, for almost all imputation function ϕ:
$\left(\phi\left(X^{(k)}_{\text{obs}(M^{(k)})}, M^{(k)}\right), Y^{(k)}\right)_{k=1}^{n}$ are **exchangeable**.

$\Rightarrow$ Conformal Prediction (CP), applied on an imputed data set still enjoys marginal guarantees:

$$\mathbb{P}\left\{Y^{(n+1)} \in \widehat{C}_\alpha\left(X^{(n+1)}, M^{(n+1)}\right)\right\} \geq 1 - \alpha.$$

CP is marginally valid on imputed data sets

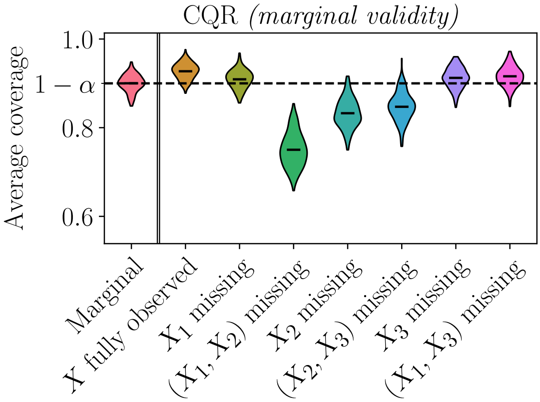
$Y = \beta^T X + \varepsilon$, $\beta = (1, 2, -1)^T$, X and ε Gaussian.



✓ Marginal (i.e. average) coverage (MV) is indeed recovered!

CP is marginally valid on imputed data sets

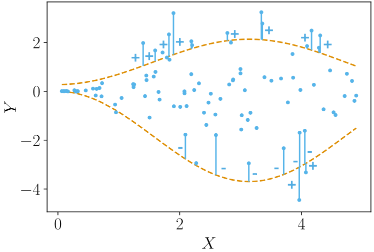
$$Y = \beta^T X + \varepsilon,\ \beta = (1, 2, -1)^T,\ X \text{ and } \varepsilon \text{ Gaussian.}$$



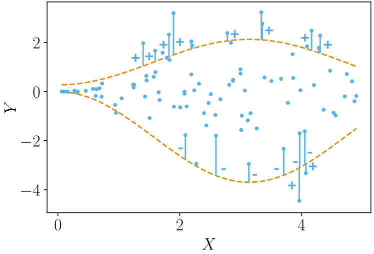
- ✓ Marginal (i.e. average) coverage (MV) is indeed recovered!
- ✗ Mask-conditional-validity (MCV) is not attained
  - ↪ Missing values induce heteroskedasticity
    (supported by theory under (non-)parametric assumptions)

Conformalization step is independent of the important variable: the mask!

Observation: the α-correction term is computed among all the data points, regardless of their mask!

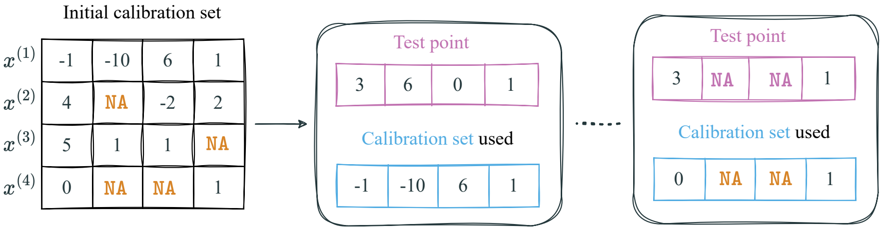


Observation: the α-correction term is computed among all the data points, regardless of their mask!



Warning: 2^d possible masks

$\Rightarrow$ Splitting the calibration set by mask is infeasible (lack of data)!



Conceptually: a structured distribution shift situation

1. For each pattern m, we have a different distribution for $(X_{\mathrm{obs(M)}}, Y)|M = m$.

Conceptually: a structured distribution shift situation

1. For each pattern m, we have a different distribution for $(X_{\mathrm{obs(M)}}, Y)|M = m$.
2. Those distributions are connected.

Conceptually: a structured distribution shift situation

1. For each pattern m, we have a different distribution for $(X_{\mathrm{obs(M)}}, Y)|M = m$.
2. Those distributions are connected.
3. Reasonable model of the link between pattern and the uncertainty.

Fully distribution-free MCV is necessarily uninformative

Theorem (General MCV hardness result (Z., Josse, Romano and Dieuleveut, 2024)[4]).

If any $\widehat{C}_\alpha$ is distribution-free MCV then for any distribution P, for any mask m such that $P_M(m) > 0$, it holds:

$$\mathbb{P}_{P^{\otimes(n+1)}}\left(\text{mes}\left(\widehat{C}_\alpha\left(X_{n+1}, m\right)\right) = \infty\right) \geq 1 - \alpha - \Delta_{m,n} \geq 1 - \alpha - P_M(m)\sqrt{n+1}.$$

[4] An analogous statement is also available for the classification framework.

Fully distribution-free MCV is necessarily uninformative

Theorem (General MCV hardness result (Z., Josse, Romano and Dieuleveut, 2024)[4]).

If any $\widehat{C}_\alpha$ is distribution-free MCV then for any distribution P, for any mask m such that $P_M(m) > 0$, it holds:

$$\mathbb{P}_{P^{\otimes(n+1)}}\left(\mathrm{mes}\left(\widehat{C}_\alpha\left(X_{n+1}, m\right)\right) = \infty\right) \geq 1 - \alpha - \Delta_{m,n} \geq 1 - \alpha - P_M(m)\sqrt{n+1}.$$

[4] An analogous statement is also available for the classification framework.

Fully distribution-free MCV is necessarily uninformative

Theorem (General MCV hardness result (Z., Josse, Romano and Dieuleveut, 2024)[4]).

If any $\widehat{C}_\alpha$ is distribution-free MCV then **for any distribution** P, for any mask m such that $P_M(m) > 0$, it holds:

$$\mathbb{P}_{P^{\otimes(n+1)}}\left(\text{mes}\left(\widehat{C}_\alpha\left(X_{n+1}, m\right)\right) = \infty\right) \geq 1 - \alpha - \Delta_{m,n} \geq 1 - \alpha - P_M(m)\sqrt{n+1}.$$

[4] An analogous statement is also available for the classification framework.

Fully distribution-free MCV is necessarily uninformative

Theorem (General MCV hardness result (Z., Josse, Romano and Dieuleveut, 2024)[4]).

If any $\widehat{C}_\alpha$ is distribution-free MCV then for any distribution P, for any mask m such that $P_M(m) > 0$, it holds:

$$\mathbb{P}_{P^{\otimes(n+1)}}\left(\text{mes}\left(\widehat{C}_\alpha\left(X_{n+1}, m\right)\right) = \infty\right) \geq 1 - \alpha - \Delta_{m,n} \geq 1 - \alpha - P_M(m)\sqrt{n+1}.$$

Irreducible term: consider $\widehat{C}_\alpha$ outputting $\mathcal{Y}$ with probability $1 - \alpha$ and $\emptyset$ otherwise.

[4] An analogous statement is also available for the classification framework.

Fully distribution-free MCV is necessarily uninformative

Theorem (General MCV hardness result (Z., Josse, Romano and Dieuleveut, 2024)[4]).

If any $\widehat{C}_\alpha$ is distribution-free MCV then for any distribution P, for any mask m such that $P_M(m) > 0$, it holds:

$$\mathbb{P}_{P^{\otimes(n+1)}}\left(\text{mes}\left(\widehat{C}_\alpha\left(X_{n+1}, m\right)\right) = \infty\right) \geq 1 - \alpha - \Delta_{m,n} \geq 1 - \alpha - P_M(m)\sqrt{n+1}.$$

Irreducible term: consider $\widehat{C}_\alpha$ outputting $\mathcal{Y}$ with probability $1-\alpha$ and $\emptyset$ otherwise.

$\Delta_{m,n}$ term: smaller than $P_M(m)\sqrt{n+1}$

[4] An analogous statement is also available for the classification framework.

Fully distribution-free MCV is necessarily uninformative

Theorem (General MCV hardness result (Z., Josse, Romano and Dieuleveut, 2024)[4]).

If any $\widehat{C}_\alpha$ is distribution-free MCV then for any distribution P, for any mask m such that $P_M(m) > 0$, it holds:

$$\mathbb{P}_{P^{\otimes(n+1)}}\left(\text{mes}\left(\widehat{C}_\alpha\left(X_{n+1}, m\right)\right) = \infty\right) \geq 1 - \alpha - \Delta_{m,n} \geq 1 - \alpha - P_M(m)\sqrt{n+1}.$$

Irreducible term: consider $\widehat{C}_\alpha$ outputting $\mathcal{Y}$ with probability $1 - \alpha$ and $\emptyset$ otherwise.

$\Delta_{m,n}$ term: smaller than $P_M(m)\sqrt{n+1}$

$\hookrightarrow$ gets negligible (making the lower bound nearly $1 - \alpha$) for low probability masks compared to n;

[4] An analogous statement is also available for the classification framework.

Fully distribution-free MCV is necessarily uninformative

Theorem (General MCV hardness result (Z., Josse, Romano and Dieuleveut, 2024)[4]).

If any $\widehat{C}_\alpha$ is distribution-free MCV then for any distribution P, for any mask m such that $P_M(m) > 0$, it holds:

$$\mathbb{P}_{P^{\otimes(n+1)}}\left(\text{mes}\left(\widehat{C}_\alpha\left(X_{n+1}, m\right)\right) = \infty\right) \geq 1 - \alpha - \Delta_{m,n} \geq 1 - \alpha - P_M(m)\sqrt{n+1}.$$

Irreducible term: consider $\widehat{C}_\alpha$ outputting $\mathcal{Y}$ with probability $1-\alpha$ and $\emptyset$ otherwise.

$\Delta_{m,n}$ term: smaller than $P_M(m)\sqrt{n+1}$

- $\hookrightarrow$ gets negligible (making the lower bound nearly $1-\alpha$) for low probability masks compared to n;
- $\hookrightarrow$ gets large (making the lower bound trivial because negative) for high probability masks compared to n.

[4] An analogous statement is also available for the classification framework.

Restricting the link between M and $(X \text{or} Y)$ does not allow informative MCV

Analogous statements are also available for the classification framework.

Restricting the link between M and $(X \text{or} Y)$ does not allow informative MCV

Theorem ($M \perp\!\!\!\perp X$ hardness result (Z., Josse, Romano and Dieuleveut, 2024)).

If any $\widehat{C}_\alpha$ is MCV under $M \perp\!\!\!\perp X$, then for any distribution P such that $M \perp\!\!\!\perp X$, for any mask m such that $P_M(m) > 0$, it holds:

$$\mathbb{P}_{P^{\otimes(n+1)}}\left(\text{mes}\left(\widehat{C}_\alpha\left(X_{n+1}, m\right)\right) = \infty\right) \geq 1 - \alpha - \Delta_{m,n} \geq 1 - \alpha - P_M(m)\sqrt{n+1}.$$

Analogous statements are also available for the classification framework.

Restricting the link between M and $(X \text{or} Y)$ does not allow informative MCV

Theorem ($M \perp\!\!\!\perp X$ hardness result (Z., Josse, Romano and Dieuleveut, 2024)).

If any $\widehat{C}_\alpha$ is MCV under $M \perp\!\!\!\perp X$, then for any distribution P such that $M \perp\!\!\!\perp X$, for any mask m such that $P_M(m) > 0$, it holds:

$$\mathbb{P}_{P^{\otimes(n+1)}}\left(\text{mes}\left(\widehat{C}_\alpha\left(X_{n+1}, m\right)\right) = \infty\right) \geq 1 - \alpha - \Delta_{m,n} \geq 1 - \alpha - P_M(m)\sqrt{n+1}.$$

Theorem ($Y \perp\!\!\!\perp M \mid X$ hardness result (Z., Josse, Romano and Dieuleveut, 2024)).

If any $\widehat{C}_\alpha$ is MCV under $Y \perp\!\!\!\perp M \mid X$, then for any distribution P such that $Y \perp\!\!\!\perp M \mid X$, for any mask m such that $\frac{1}{\sqrt{2}} \geq P_M(m) > 0$, it holds:

$$\mathbb{P}_{P^{\otimes(n+1)}}\left(\text{mes}\left(\widehat{C}_\alpha\left(X_{n+1}, m\right)\right) = \infty\right) \geq 1 - \alpha - \Delta_{m,n} \geq 1 - \alpha - 2P_M(m)\sqrt{n+1}.$$

Analogous statements are also available for the classification framework.

Restricting the link between M and $(X \text{or} Y)$ does not allow informative MCV

Theorem ($M \perp\!\!\!\perp X$ hardness result (Z., Josse, Romano and Dieuleveut, 2024)).

If any $\widehat{C}_\alpha$ is MCV under $M \perp\!\!\!\perp X$, then for any distribution P such that $M \perp\!\!\!\perp X$, for any mask m such that $P_M(m) > 0$, it holds:

$$\mathbb{P}_{P^{\otimes(n+1)}}\left(\text{mes}\left(\widehat{C}_\alpha\left(X_{n+1}, m\right)\right) = \infty\right) \geq 1-\alpha-\Delta_{m,n} \geq 1-\alpha-P_M(m)\sqrt{n+1}.$$

Theorem ($Y \perp\!\!\!\perp M \mid X$ hardness result (Z., Josse, Romano and Dieuleveut, 2024)).

If any $\widehat{C}_\alpha$ is MCV under $Y \perp\!\!\!\perp M \mid X$, then for any distribution P such that $Y \perp\!\!\!\perp M \mid X$, for any mask m such that $\frac{1}{\sqrt{2}} \geq P_M(m) > 0$, it holds:

$$\mathbb{P}_{P^{\otimes(n+1)}}\left(\text{mes}\left(\widehat{C}_\alpha\left(X_{n+1}, m\right)\right) = \infty\right) \geq 1-\alpha-\Delta_{m,n} \geq 1-\alpha-2P_M(m)\sqrt{n+1}.$$

$\Rightarrow$ need to restrict both the link between M and X, as well as between M and Y.

Analogous statements are also available for the classification framework.

Missing Data Augmentation (MDA) of the calibration set

Idea: for each test point, modify the calibration points to mimic the test mask

Test point

3	NA	NA	1

Initial calibration set

$x^{(1)}$	-1	-10	6	1
$x^{(2)}$	4	NA	-2	2
$x^{(3)}$	5	1	1	NA
$x^{(4)}$	0	NA	NA	1

Calibration set used

$\tilde{x}^{(1)}$	-1	NA	NA	1
$\tilde{x}^{(2)}$	4	NA	NA	2
$\tilde{x}^{(3)}$	5	NA	NA	NA
$\tilde{x}^{(4)}$	0	NA	NA	1

CP-MDA with Exact masking

Test point

3	NA	NA	1

Initial calibration set

$x^{(1)}$	-1	-10	6	1
$x^{(2)}$	4	NA	-2	2
$x^{(3)}$	5	1	1	NA
$x^{(4)}$	0	NA	NA	1

Calibration set used

$\tilde{x}^{(1)}$	-1	NA	NA	1
$\tilde{x}^{(2)}$	4	NA	NA	2
$\tilde{x}^{(3)}$	5	NA	NA	NA
$\tilde{x}^{(4)}$	0	NA	NA	1

CP-MDA with Exact masking

Test point

3	NA	NA	1

Initial calibration set

$x^{(1)}$	-1	-10	6	1
$x^{(2)}$	4	NA	-2	2
$x^{(3)}$	5	1	1	NA
$x^{(4)}$	0	NA	NA	1

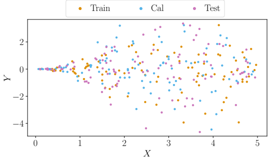
Calibration set used

$\tilde{x}^{(1)}$	-1	NA	NA	1
$\tilde{x}^{(2)}$	4	NA	NA	2
$\tilde{x}^{(3)}$				
$\tilde{x}^{(4)}$	0	NA	NA	1

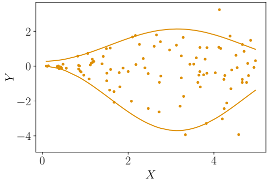
$\#\mathrm{Cal}^{\mathrm{M}^{(\text{test})}}$ observations

CQR-MDA with exact masking in words

1. Split the training set into a `proper training set` and `calibration set`



2. Train the imputation function on the `proper training set`
3. Impute the `proper training set`
4. Train the quantile regressors on the imputed `proper training set`



CQR-MDA with exact masking in words

1. Split the training set into a `proper training set` and `calibration set`
2. Train the imputation function on the `proper training set`
3. Impute the `proper training set`
4. Train the quantile regressors on the `imputed proper training set`
5. For a test point $\left(X^{(n+1)}, M^{(n+1)}\right)$:

3	NA	NA	1

CQR-MDA with exact masking in words

1. Split the training set into a `proper training set` and `calibration set`
2. Train the imputation function on the `proper training set`
3. Impute the `proper training set`
4. Train the quantile regressors on the `imputed proper training set`
5. For a test point $\left(X^{(n+1)}, M^{(n+1)}\right)$:

3	NA	NA	1

$\tilde{x}^{(1)}$	-1	NA	NA	1
$\tilde{x}^{(2)}$	4	NA	NA	2
$\tilde{x}^{(3)}$				
$\tilde{x}^{(4)}$	0	NA	NA	1

5.1 For each $j \in [\![1, d]\!]$ s.t. $M_j^{(n+1)} = 1$, set $\tilde{M}_j^{(k)} = 1$ for k in `Cal` s.t. $M^{(k)} \subset M^{(n+1)}$

CQR-MDA with exact masking in words

1. Split the training set into a `proper training set` and `calibration set`
2. Train the imputation function on the `proper training set`
3. Impute the `proper training set`
4. Train the quantile regressors on the `imputed proper training set`
5. For a test point $\left(X^{(n+1)}, M^{(n+1)}\right)$:

   5.1 For each $j \in [\![1, d]\!]$ s.t. $M_j^{(n+1)} = 1$, set $\tilde{M}_j^{(k)} = 1$ for k in `Cal` s.t. $M^{(k)} \subset M^{(n+1)}$

   5.2 Impute the new `calibration set`

3	NA	NA	1

$\tilde{x}^{(1)}$	-1	NA	NA	1
$\tilde{x}^{(2)}$	4	NA	NA	2
$\tilde{x}^{(3)}$				
$\tilde{x}^{(4)}$	0	NA	NA	1

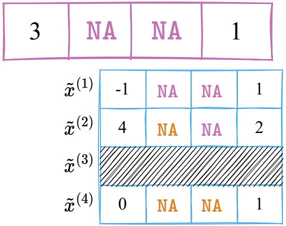
CQR-MDA with exact masking in words

1. Split the training set into a proper training set and calibration set
2. Train the imputation function on the proper training set
3. Impute the proper training set
4. Train the quantile regressors on the imputed proper training set
5. For a test point $\left(X^{(n+1)}, M^{(n+1)}\right)$:

   5.1 For each $j \in [\![1, d]\!]$ s.t. $M_j^{(n+1)} = 1$, set $\tilde{M}_j^{(k)} = 1$ for k in Cal s.t. $M^{(k)} \subset M^{(n+1)}$

   5.2 Impute the new calibration set

   5.3 Compute the calibration correction, i.e. $q_{1-\alpha}(\mathcal{S})$



1. Split the training set into a `proper training set` and `calibration set`
2. Train the imputation function on the `proper training set`
3. Impute the `proper training set`
4. Train the quantile regressors on the `imputed proper training set`
5. For a test point $\left(X^{(n+1)}, M^{(n+1)}\right)$:

   5.1 For each $j \in [\![1, d]\!]$ s.t. $M_j^{(n+1)} = 1$, set $\tilde{M}_j^{(k)} = 1$ for k in `Cal` s.t. $M^{(k)} \subset M^{(n+1)}$

   5.2 Impute the new `calibration set`

   5.3 Compute the calibration correction, i.e. $q_{1-\alpha}(\mathcal{S})$

   5.4 Impute the test point

3	NA	NA	1

$\tilde{x}^{(1)}$	-1	NA	NA	1
$\tilde{x}^{(2)}$	4	NA	NA	2
$\tilde{x}^{(3)}$				
$\tilde{x}^{(4)}$	0	NA	NA	1

CQR-MDA with exact masking in words

1. Split the training set into a proper training set and calibration set
2. Train the imputation function on the proper training set
3. Impute the proper training set
4. Train the quantile regressors on the imputed proper training set
5. For a test point $\left(X^{(n+1)}, M^{(n+1)}\right)$:

3	NA	NA	1

$\tilde{x}^{(1)}$	-1	NA	NA	1
$\tilde{x}^{(2)}$	4	NA	NA	2
$\tilde{x}^{(3)}$				
$\tilde{x}^{(4)}$	0	NA	NA	1

5.1 For each $j \in [\![1, d]\!]$ s.t. $M_j^{(n+1)} = 1$, set $\tilde{M}_j^{(k)} = 1$ for k in Cal s.t. $M^{(k)} \subset M^{(n+1)}$

5.2 Impute the new calibration set

5.3 Compute the calibration correction, i.e. $q_{1-\alpha}(\mathcal{S})$

5.4 Impute the test point

5.5 Predict with the quantile regressors and the correction previously obtained, $q_{1-\alpha}(\mathcal{S})$

MDA-Exact achieves Mask-Conditional-Validity (MCV)

Theorem (CP-MDA-Exact achieves MCV).

If: i) the data is exchangeable, ii) $M \perp\!\!\!\perp X$, iii) $(Y \perp\!\!\!\perp M)|X$, then for almost all imputation function CP-MDA-Exact is such that for any $m \in \{0,1\}^d$:

$$\mathbb{P}\left(Y \in \widehat{C}_\alpha\left(X, m\right) | M = m\right) \geq 1 - \alpha,$$

and if additionally the scores are almost surely distinct:

$$\mathbb{P}\left(Y \in \widehat{C}_\alpha\left(X, m\right) | M = m\right) \leq 1 - \alpha + \frac{1}{\#\mathrm{Cal}^{\mathrm{m}} + 1}.$$

What if we kept all observations?

Test point

3	NA	NA	1

Initial calibration set

$x^{(1)}$	-1	-10	6	1
$x^{(2)}$	4	NA	-2	2
$x^{(3)}$	5	1	1	NA
$x^{(4)}$	0	NA	NA	1

Calibration set used

$\tilde{x}^{(1)}$	-1	NA	NA	1
$\tilde{x}^{(2)}$	4	NA	NA	2
$\tilde{x}^{(3)}$	5	NA	NA	NA
$\tilde{x}^{(4)}$	0	NA	NA	1

Idea: modify the test point accordingly

Test point

3	NA	NA	1

Initial calibration set

$x^{(1)}$	-1	-10	6	1
$x^{(2)}$	4	NA	-2	2
$x^{(3)}$	5	1	1	NA
$x^{(4)}$	0	NA	NA	1

Calibration set used

$\tilde{x}^{(1)}$	-1	NA	NA	1
$\tilde{x}^{(2)}$	4	NA	NA	2
$\tilde{x}^{(3)}$	5	NA	NA	NA
$\tilde{x}^{(4)}$	0	NA	NA	1

and

Temporary test points

3	NA	NA	1
3	NA	NA	1
3	NA	NA	NA
3	NA	NA	1

$\rightsquigarrow$ similar motivation than Barber et al. (2021)[5] and Gupta et al. (2022)[6].

[5] *Predictive inference with the jackknife+*, The Annals of Statistics

[6] *Nested conformal prediction and quantile out-of-bag ensemble methods*, Pattern Recognition

CQR-MDA with nested masking in words

5. For a test point $\left(X^{(n+1)}, M^{(n+1)}\right)$:

    5.1 Set $\tilde{M}^{(k)} = \max(M^{(k)}, M^{(n+1)})$ for k in the `calibration set`

3	NA	NA	1

$\tilde{x}^{(1)}$	-1	NA	NA	1
$\tilde{x}^{(2)}$	4	NA	NA	2
$\tilde{x}^{(3)}$	5	NA	NA	NA
$\tilde{x}^{(4)}$	0	NA	NA	1

CQR-MDA with nested masking in words

5. For a test point $\left(X^{(n+1)}, M^{(n+1)}\right)$:

   5.1 Set $\tilde{M}^{(k)} = \max(M^{(k)}, M^{(n+1)})$ for k in the calibration set

   5.2 Impute the new calibration set

   5.3 For each augmented calibration point k:

3	NA	NA	1

$\tilde{x}^{(1)}$	-1	NA	NA	1
$\tilde{x}^{(2)}$	4	NA	NA	2
$\tilde{x}^{(3)}$	5	NA	NA	NA
$\tilde{x}^{(4)}$	0	NA	NA	1

CQR-MDA with nested masking in words

5. For a test point $\left(X^{(n+1)}, M^{(n+1)}\right)$:

    5.1 Set $\tilde{M}^{(k)} = \max(M^{(k)}, M^{(n+1)})$ for k in the calibration set

    5.2 Impute the new calibration set

    5.3 For each augmented calibration point k:

        5.3.1 Get its score $S^{(k)}$

3	NA	NA	1

$\tilde{x}^{(1)}$	-1	NA	NA	1
$\tilde{x}^{(2)}$	4	NA	NA	2
$\tilde{x}^{(3)}$	5	NA	NA	NA
$\tilde{x}^{(4)}$	0	NA	NA	1

5. For a test point $\left(X^{(n+1)}, M^{(n+1)}\right)$:

3	NA	NA	1

   5.1 Set $\tilde{M}^{(k)} = \max(M^{(k)}, M^{(n+1)})$ for k in the calibration set

$\tilde{x}^{(1)}$	-1	NA	NA	1
$\tilde{x}^{(2)}$	4	NA	NA	2
$\tilde{x}^{(3)}$	5	NA	NA	NA
$\tilde{x}^{(4)}$	0	NA	NA	1

   5.2 Impute the new calibration set

   5.3 For each augmented calibration point k:

      5.3.1 Get its score $S^{(k)}$

      5.3.2 Impute-then-predict on the augmented test point $(X^{(n+1)}, \tilde{M}^{(k)})$, giving: $\widehat{QR}_{\alpha/2}(\tilde{X}^{(n+1),k})$ and $\widehat{QR}_{1-\alpha/2}(\tilde{X}^{(n+1),k})$

3	NA	NA	1
3	NA	NA	1
3	NA	NA	NA
3	NA	NA	1

CQR-MDA with nested masking in words

5. For a test point $\left(X^{(n+1)}, M^{(n+1)}\right)$:

3	NA	NA	1

   5.1 Set $\tilde{M}^{(k)} = \max(M^{(k)}, M^{(n+1)})$ for k in the `calibration set`

$\tilde{x}^{(1)}$	-1	NA	NA	1
$\tilde{x}^{(2)}$	4	NA	NA	2
$\tilde{x}^{(3)}$	5	NA	NA	NA
$\tilde{x}^{(4)}$	0	NA	NA	1

   5.2 Impute the new `calibration set`

   5.3 For each augmented calibration point k:

      5.3.1 Get its score $S^{(k)}$

      5.3.2 Impute-then-predict on the augmented test point $(X^{(n+1)}, \tilde{M}^{(k)})$, giving: $\widehat{QR}_{\alpha/2}(\tilde{X}^{(n+1),k})$ and $\widehat{QR}_{1-\alpha/2}(\tilde{X}^{(n+1),k})$

3	NA	NA	1
3	NA	NA	1
3	NA	NA	NA
3	NA	NA	1

      5.3.3 Compute the corrected prediction interval:

$$[\widehat{QR}_{\alpha/2}(\tilde{X}^{(n+1),k}) - S^{(k)}; \widehat{QR}_{1-\alpha/2}(\tilde{X}^{(n+1),k}) + S^{(k)}] := \left[Z^{(k)}_{\text{lower}}; Z^{(k)}_{\text{upper}}\right]$$

CQR-MDA with nested masking in words

5. For a test point $\left(X^{(n+1)}, M^{(n+1)}\right)$:

3	NA	NA	1

$\tilde{x}^{(1)}$	-1	NA	NA	1
$\tilde{x}^{(2)}$	4	NA	NA	2
$\tilde{x}^{(3)}$	5	NA	NA	NA
$\tilde{x}^{(4)}$	0	NA	NA	1

5.1 Set $\tilde{M}^{(k)} = \max(M^{(k)}, M^{(n+1)})$ for k in the calibration set

5.2 Impute the new calibration set

5.3 For each augmented calibration point k:

5.3.1 Get its score $S^{(k)}$

3	NA	NA	1
3	NA	NA	1
3	NA	NA	NA
3	NA	NA	1

5.3.2 Impute-then-predict on the augmented test point $(X^{(n+1)}, \tilde{M}^{(k)})$, giving: $\widehat{QR}_{\alpha/2}(\tilde{X}^{(n+1),k})$ and $\widehat{QR}_{1-\alpha/2}(\tilde{X}^{(n+1),k})$

5.3.3 Compute the corrected prediction interval:
$$[\widehat{QR}_{\alpha/2}(\tilde{X}^{(n+1),k}) - S^{(k)}; \widehat{QR}_{1-\alpha/2}(\tilde{X}^{(n+1),k}) + S^{(k)}] := \left[Z^{(k)}_{\text{lower}}; Z^{(k)}_{\text{upper}}\right]$$

5.4 Compute the quantiles $q_\alpha(\{Z^{(k)}_{\text{lower}}\}_{k\in\mathrm{Cal}})$ and $q_{1-\alpha}(\{Z^{(k)}_{\text{upper}}\}_{k\in\mathrm{Cal}})$

CQR-MDA with nested masking in words

5. For a test point $\left(X^{(n+1)}, M^{(n+1)}\right)$:

3	NA	NA	1

5.1 Set $\tilde{M}^{(k)} = \max(M^{(k)}, M^{(n+1)})$ for k in the calibration set

$\tilde{x}^{(1)}$	-1	NA	NA	1
$\tilde{x}^{(2)}$	4	NA	NA	2
$\tilde{x}^{(3)}$	5	NA	NA	NA
$\tilde{x}^{(4)}$	0	NA	NA	1

5.2 Impute the new calibration set

5.3 For each augmented calibration point k:

5.3.1 Get its score $S^{(k)}$

5.3.2 Impute-then-predict on the augmented test point $(X^{(n+1)}, \tilde{M}^{(k)})$, giving: $\widehat{QR}_{\alpha/2}(\tilde{X}^{(n+1),k})$ and $\widehat{QR}_{1-\alpha/2}(\tilde{X}^{(n+1),k})$

3	NA	NA	1
3	NA	NA	1
3	NA	NA	NA
3	NA	NA	1

5.3.3 Compute the corrected prediction interval:
$[\widehat{QR}_{\alpha/2}(\tilde{X}^{(n+1),k}) - S^{(k)}; \widehat{QR}_{1-\alpha/2}(\tilde{X}^{(n+1),k}) + S^{(k)}] := \left[Z^{(k)}_{\text{lower}}; Z^{(k)}_{\text{upper}}\right]$

5.4 Compute the quantiles $q_\alpha(\{Z^{(k)}_{\text{lower}}\}_{k\in\mathrm{Cal}})$ and $q_{1-\alpha}(\{Z^{(k)}_{\text{upper}}\}_{k\in\mathrm{Cal}})$

5.5 Predict $[q_\alpha(\{Z^{(k)}_{\text{lower}}\}_{k\in\mathrm{Cal}}); q_{1-\alpha}(\{Z^{(k)}_{\text{upper}}\}_{k\in\mathrm{Cal}})]$

MDA-Nested is Marginally Valid (MV)

Theorem (CP-MDA-Nested marginal validity)**.**

If the data is exchangeable, then for almost all imputation function CP-MDA-Nested is such that:

$$\mathbb{P}\left(Y \in \widehat{C}_\alpha\left(X, M\right)\right) \geq 1 - 2\alpha.$$

MDA-Nested is Marginally Valid (MV)

Theorem (CP-MDA-Nested marginal validity)**.**

If the data is exchangeable, then for almost all imputation function CP-MDA-Nested is such that:

$$\mathbb{P}\left(Y \in \widehat{C}_\alpha\left(X, M\right)\right) \geq 1 - 2\alpha.$$

- ✓ Any missing mechanism (no need to assume $M \perp\!\!\!\perp X$)
- ✓ Does not require $\left(Y \perp\!\!\!\perp M\right)|X$
- ✗ Marginal guarantee

MDA-Nested is Marginally Valid (MV)

Theorem (CP-MDA-Nested marginal validity).

If the data is exchangeable, then for almost all imputation function CP-MDA-Nested is such that:

$$\mathbb{P}\left(Y \in \widehat{C}_{\alpha}\left(X, M\right)\right) \geq 1 - 2\alpha.$$

- ✓ Any missing mechanism (no need to assume $M \perp\!\!\!\perp X$)
- ✓ Does not require $(Y \perp\!\!\!\perp M)\,|X$
- ✗ Marginal guarantee

Proof element: based on Jackknife+ ideas (Barber et al., 2021).

Leaving-out the k-th data point to predict on the l-th data point

$\leftrightarrow$

Apply the mask of the k-th data point to the l-th data point on which you predict

CP-MDA-Nested* (Missing Data Augmentation)

Idea: for each test point, modify the calibration points to mimic the test mask

Idea: for each test point, modify the calibration points to mimic the test mask

Overmasked calibration set

Test point

3	NA	NA	1	2

Initial calibration set

$x^{(1)}$	-1	-10	6	1	0
$x^{(2)}$	4	NA	-2	2	1
$x^{(3)}$	5	1	1	NA	3
$x^{(4)}$	0	NA	NA	1	-2
$x^{(5)}$	NA	-3	0	NA	NA

$\tilde{x}^{(1)}$	-1	NA	NA	1	0
$\tilde{x}^{(2)}$	4	NA	NA	2	1
$\tilde{x}^{(3)}$	5	NA	NA	NA	3
$\tilde{x}^{(4)}$	0	NA	NA	1	-2
$\tilde{x}^{(5)}$	NA	NA	NA	NA	NA

CP-MDA-Nested* (Missing Data Augmentation)

Idea: for each test point, modify the calibration points to mimic the test mask

Test point

3	NA	NA	1	2

Initial calibration set

$x^{(1)}$	-1	-10	6	1	0
$x^{(2)}$	4	NA	-2	2	1
$x^{(3)}$	5	1	1	NA	3
$x^{(4)}$	0	NA	NA	1	-2
$x^{(5)}$	NA	-3	0	NA	NA

$i)$ keep same mask

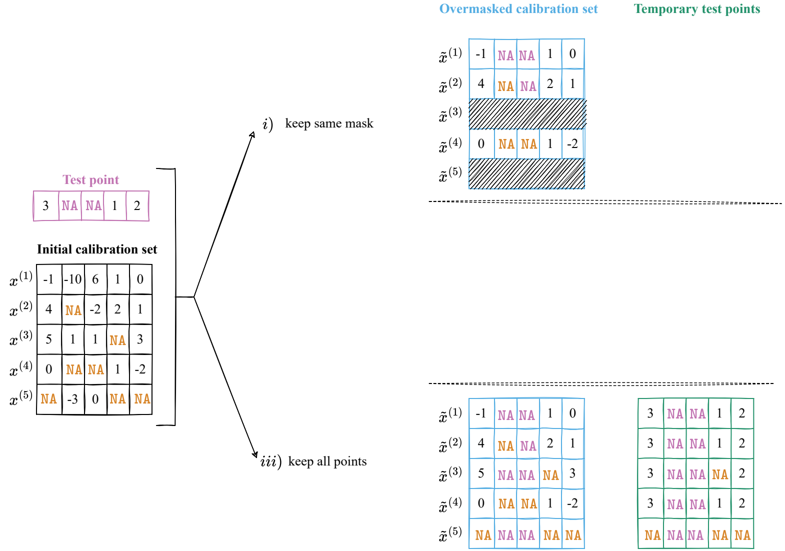
Overmasked calibration set

$\tilde{x}^{(1)}$	-1	NA	NA	1	0
$\tilde{x}^{(2)}$	4	NA	NA	2	1
$\tilde{x}^{(3)}$					
$\tilde{x}^{(4)}$	0	NA	NA	1	-2
$\tilde{x}^{(5)}$					

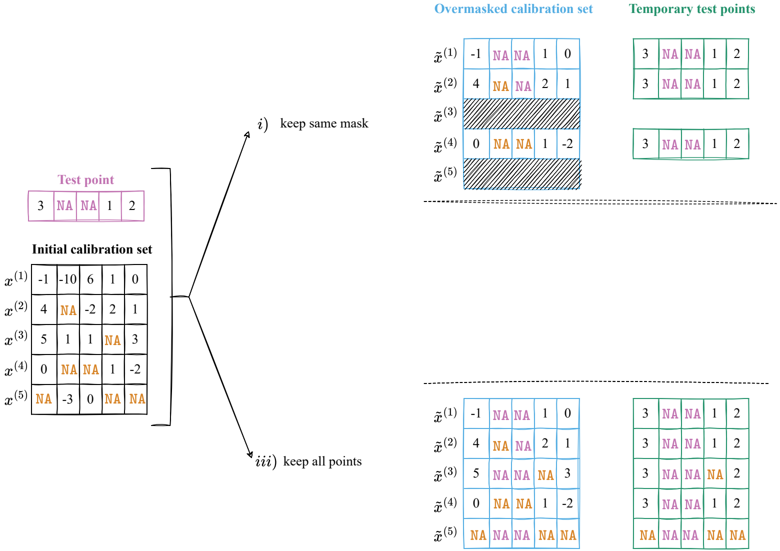
$\tilde{x}^{(1)}$	-1	NA	NA	1	0
$\tilde{x}^{(2)}$	4	NA	NA	2	1
$\tilde{x}^{(3)}$	5	NA	NA	NA	3
$\tilde{x}^{(4)}$	0	NA	NA	1	-2
$\tilde{x}^{(5)}$	NA	NA	NA	NA	NA

CP-MDA-Nested* (Missing Data Augmentation)

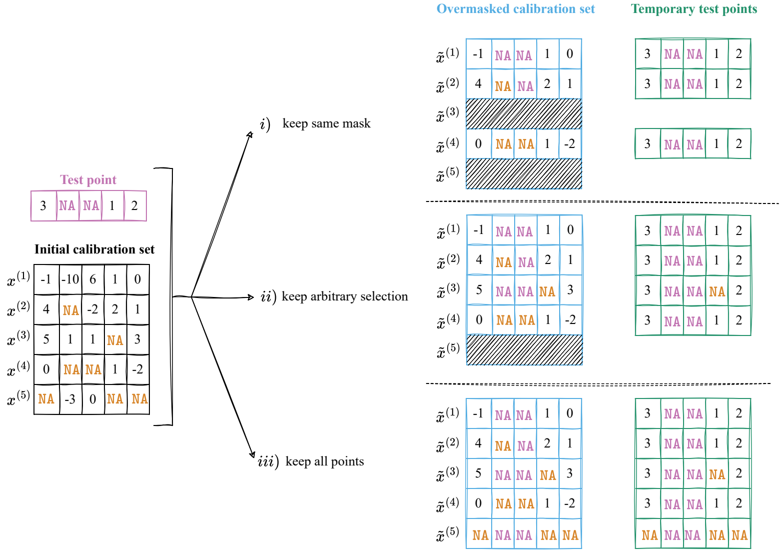
Idea: for each test point, modify the calibration points to mimic the test mask



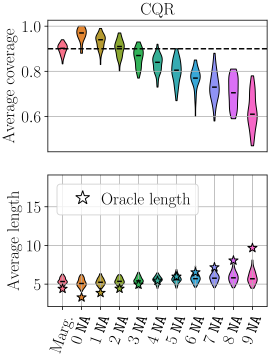
Idea: for each test point, modify the calibration points to mimic the test mask



Idea: for each test point, modify the calibration points to mimic the test mask

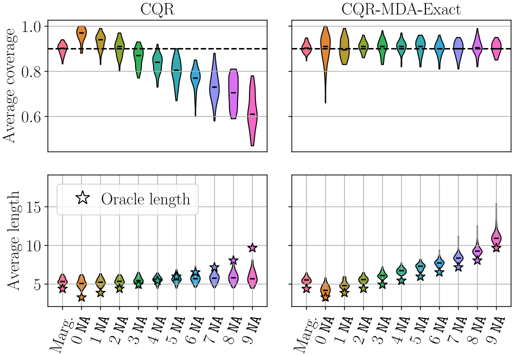


Experiments on $M \perp\!\!\!\perp X$ and $Y \perp\!\!\!\perp M \mid X$ Gaussian linear data in dimension 10



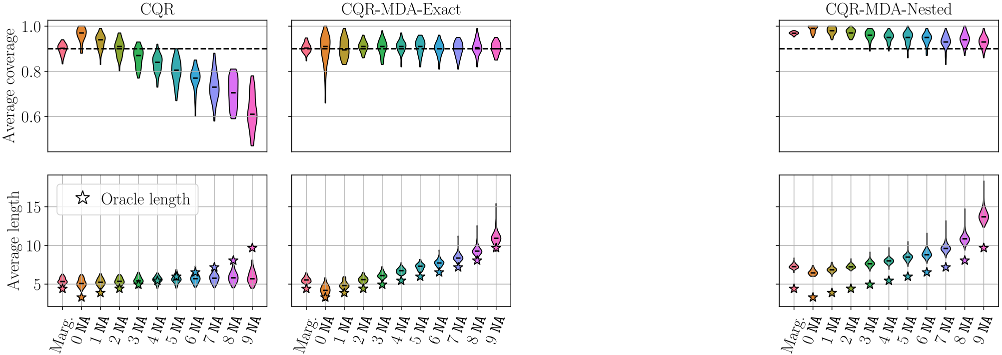
20% of missing values

Experiments on $M \perp\!\!\!\perp X$ and $Y \perp\!\!\!\perp M \mid X$ Gaussian linear data in dimension 10



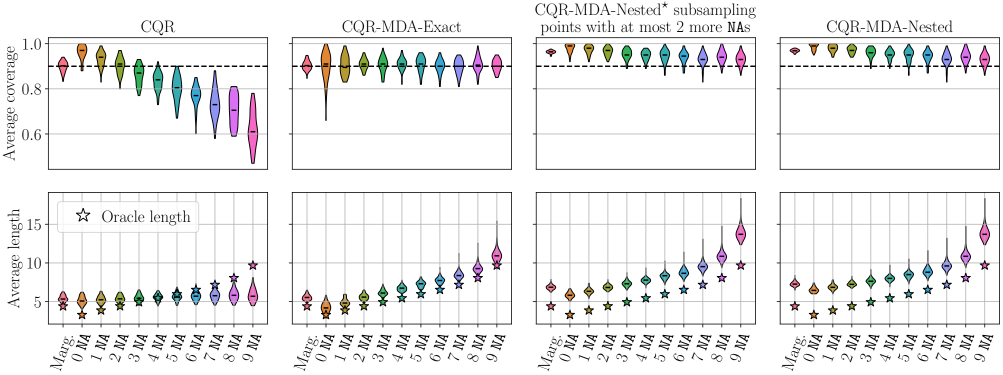
20% of missing values

Experiments on $M \perp\!\!\!\perp X$ and $Y \perp\!\!\!\perp M \,|X$ Gaussian linear data in dimension 10



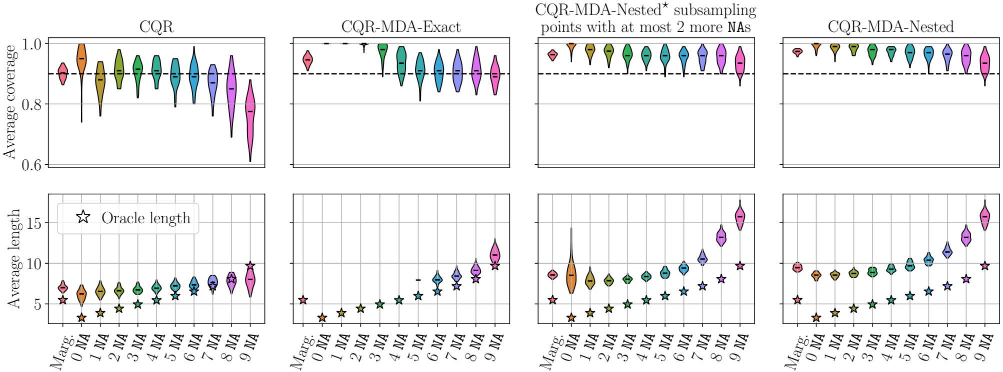
20% of missing values

Experiments on $M \perp\!\!\!\perp X$ and $Y \perp\!\!\!\perp M \mid X$ Gaussian linear data in dimension 10



20% of missing values

Experiments on $M \perp\!\!\!\perp X$ and $Y \perp\!\!\!\perp M \mid X$ Gaussian linear data in dimension 10



40% of missing values

Theorem (Mask-conditional-validity of CP-MDA-Nested* (Z., Josse, Romano and Dieuleveut, 2024)).

Under the assumptions that:

- $M \perp\!\!\!\perp (X, Y)$,
- $\left(X^{(k)}, M^{(k)}, Y^{(k)}\right)_{k=1}^{n+1}$ are i.i.d.,

then, for almost all imputation function, CP-MDA-Nested* reaches (MCV) at the level $1 - 2\alpha$, that is:

$$\mathbb{P}\left\{Y^{(n+1)} \in \widehat{C}_\alpha\left(X^{(n+1)}, M^{(n+1)}\right) | M^{(n+1)}\right\} \overset{a.s.}{\geq} 1 - 2\alpha.$$

Theorem (Mask-conditional-validity of CP-MDA-Nested* (Z., Josse, Romano and Dieuleveut, 2024)).

Under the assumptions that:

- $M \perp\!\!\!\perp (X, Y)$,
- $\left(X^{(k)}, M^{(k)}, Y^{(k)}\right)_{k=1}^{n+1}$ are i.i.d.,

then, for almost all imputation function, CP-MDA-Nested* reaches (MCV) at the level $1 - 2\alpha$, that is:

$$\mathbb{P}\left\{Y^{(n+1)} \in \widehat{C}_\alpha\left(X^{(n+1)}, M^{(n+1)}\right) | M^{(n+1)}\right\} \overset{a.s.}{\geq} 1 - 2\alpha.$$

Proof elements:

1. Crop the data sets to hide the missing entries of the test point
2. Applying the mask of the calibration point corresponds to a predictor that draws a predictions randomly

$\Rightarrow$ Use the same proof arguments than (Barber et al., 2021) on random predictors

Valididities of predictive uncertainty quantification with missing values

Goal: predict $Y^{(n+1)}$ with **confidence** $1-\alpha$, i.e. build the smallest $\mathcal{C}_\alpha$ such that:

Definition (1. Marginal Validity (MV)).

$$\mathbb{P}\left\{Y^{(n+1)} \in \mathcal{C}_\alpha\left(X^{(n+1)}, M^{(n+1)}\right)\right\} \geq 1-\alpha. \qquad \text{(MV)}$$

Definition (2. Mask-Conditional-Validity (MCV)).

$$\mathbb{P}\left\{Y^{(n+1)} \in \mathcal{C}_\alpha\left(X^{(n+1)}, M^{(n+1)}\right) | M^{(n+1)}\right\} \overset{a.s.}{\geq} 1-\alpha. \qquad \text{(MCV)}$$

	Exisiting approaches	
(MV)	✓ (Z., Dieuleveut, Josse, and Romano, 2023)	
(MCV)		

Validities of predictive uncertainty quantification with missing values

Goal: predict $Y^{(n+1)}$ with **confidence** $1-\alpha$, i.e. build the smallest $\mathcal{C}_\alpha$ such that:

Definition (1. Marginal Validity (MV)).

$$\mathbb{P}\left\{Y^{(n+1)} \in \mathcal{C}_\alpha\left(X^{(n+1)}, M^{(n+1)}\right)\right\} \geq 1-\alpha. \qquad \text{(MV)}$$

Definition (2. Mask-Conditional-Validity (MCV)).

$$\mathbb{P}\left\{Y^{(n+1)} \in \mathcal{C}_\alpha\left(X^{(n+1)}, M^{(n+1)}\right) | M^{(n+1)}\right\} \overset{a.s.}{\geq} 1-\alpha. \qquad \text{(MCV)}$$

	Exisiting approaches	
(MV)	✓	
	(Z., Dieuleveut, Josse, and Romano, 2023)	
(MCV)	✗	

Validities of predictive uncertainty quantification with missing values

Goal: predict $Y^{(n+1)}$ with **confidence** $1 - \alpha$, i.e. build the smallest $\mathcal{C}_\alpha$ such that:

Definition (1. Marginal Validity (MV)).

$$\mathbb{P}\left\{Y^{(n+1)} \in \mathcal{C}_\alpha\left(X^{(n+1)}, M^{(n+1)}\right)\right\} \geq 1 - \alpha. \tag{MV}$$

Definition (2. Mask-Conditional-Validity (MCV)).

$$\mathbb{P}\left\{Y^{(n+1)} \in \mathcal{C}_\alpha\left(X^{(n+1)}, M^{(n+1)}\right) | M^{(n+1)}\right\} \overset{a.s.}{\geq} 1 - \alpha. \tag{MCV}$$

	Exisiting approaches	CP-MDA-Nested* (Z., Josse, Romano and Dieuleveut, 2024)
(MV)	✓ (Z., Dieuleveut, Josse, and Romano, 2023)	✓
(MCV)	✗	✓

Valididities of predictive uncertainty quantification with missing values

Goal: predict $Y^{(n+1)}$ with **confidence** $1-\alpha$, i.e. build the smallest $\mathcal{C}_\alpha$ such that:

Definition (1. Marginal Validity (MV)).

$$\mathbb{P}\left\{Y^{(n+1)} \in \mathcal{C}_\alpha\left(X^{(n+1)}, M^{(n+1)}\right)\right\} \geq 1-\alpha. \qquad \text{(MV)}$$

Definition (2. Mask-Conditional-Validity (MCV)).

$$\mathbb{P}\left\{Y^{(n+1)} \in \mathcal{C}_\alpha\left(X^{(n+1)}, M^{(n+1)}\right) | M^{(n+1)}\right\} \overset{a.s.}{\geq} 1-\alpha. \qquad \text{(MCV)}$$

	Exisiting approaches	`CP-MDA-Nested`* (Z., Josse, Romano and Dieuleveut, 2024)
(MV)	✓ (Z., Dieuleveut, Josse, and Romano, 2023)	✓
(MCV)	✗	✓ under $M \perp\!\!\!\perp (X, Y)$

Some settings

- Imputation by iterative ridge ($\sim$ conditional expectation)

Some settings

- Imputation by iterative ridge ($\sim$ conditional expectation)
- **Concatenate the mask in the features**

Some settings

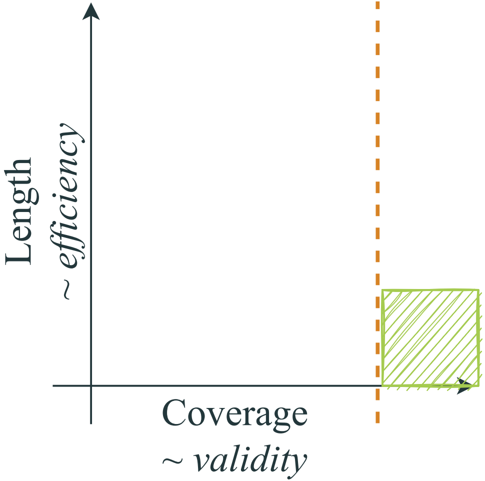
- Imputation by iterative ridge ($\sim$ conditional expectation)
- **Concatenate the mask in the features**
- Neural network, fitted to minimize the pinball loss

Some settings

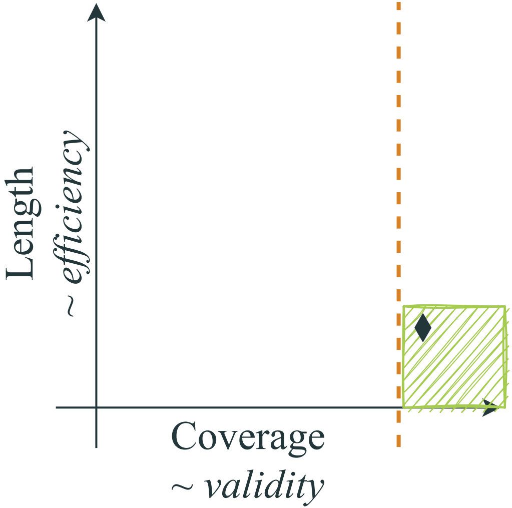
- Imputation by iterative ridge ($\sim$ conditional expectation)
- **Concatenate the mask in the features**
- Neural network, fitted to minimize the pinball loss
- (Semi)-synthetic experiments:

Some settings

- Imputation by iterative ridge ($\sim$ conditional expectation)
- **Concatenate the mask in the features**
- Neural network, fitted to minimize the pinball loss
- (Semi)-synthetic experiments:
  - Uniform MCAR missing values, with probability 20%
  - 100 repetitions


Length
$\sim$ *efficiency*
Coverage
$\sim$ *validity*

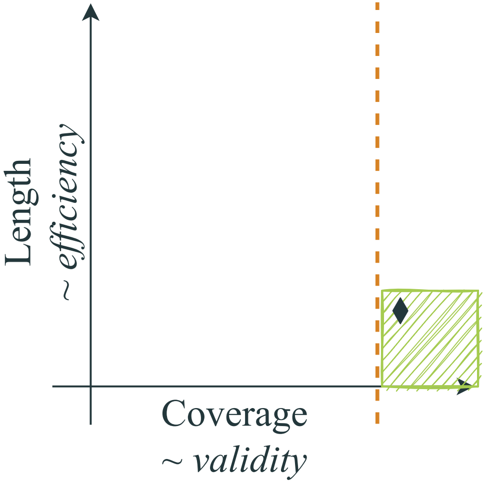
Before more experiments, visualisation



♦ : marginal coverage, i.e.

$$\mathbb{P}(Y \in \hat{C}_\alpha(X, M))$$

Before more experiments, visualisation



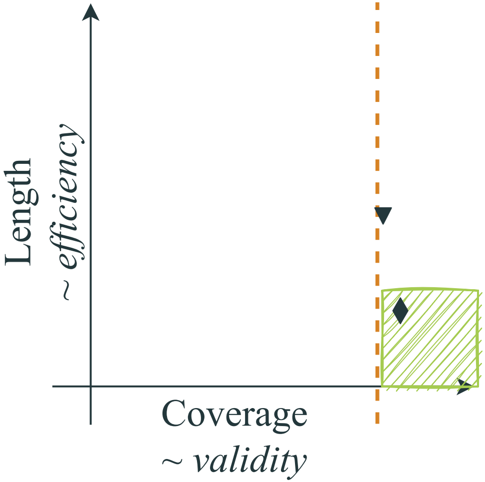
♦ : marginal coverage, i.e.

$$\mathbb{P}(Y \in \hat{C}_\alpha(X, M))$$

▼ : lowest coverage, i.e.

$$\min_{m \in \mathcal{M}} \mathbb{P}(Y \in \hat{C}_\alpha(X, m) | M = m)$$

Before more experiments, visualisation



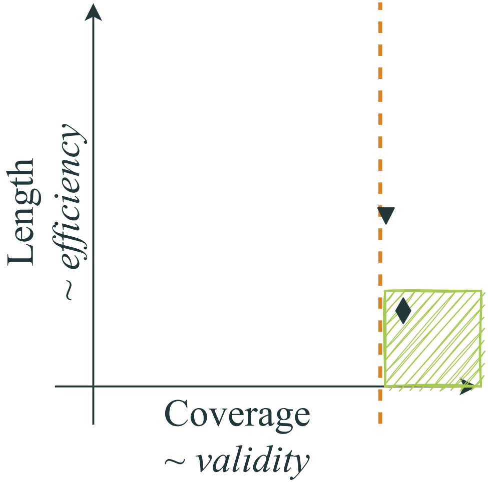
♦ : marginal coverage, i.e.

$$\mathbb{P}(Y \in \hat{C}_\alpha(X, M))$$

▼ : lowest coverage, i.e.

$$\min_{m \in \mathcal{M}} \mathbb{P}(Y \in \hat{C}_\alpha(X, m) | M = m)$$

Before more experiments, visualisation



♦ : marginal coverage, i.e.

$$\mathbb{P}(Y \in \hat{C}_\alpha(X, M))$$

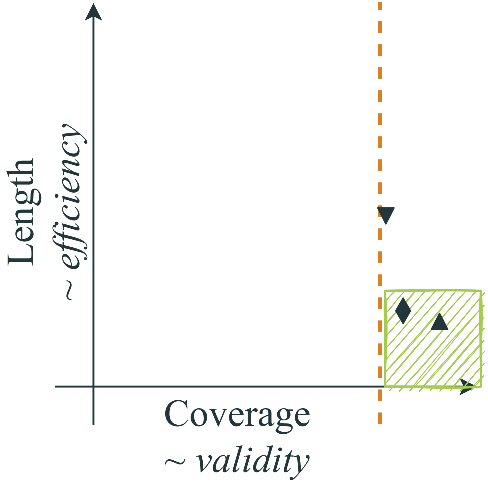
▼ : lowest coverage, i.e.

$$\min_{m \in \mathcal{M}} \mathbb{P}(Y \in \hat{C}_\alpha(X, m) | M = m)$$

▲ : highest coverage, i.e.

$$\max_{m \in \mathcal{M}} \mathbb{P}(Y \in \hat{C}_\alpha(X, m) | M = m)$$

Before more experiments, visualisation



◆ : marginal coverage, i.e.

$$\mathbb{P}(Y \in \hat{C}_\alpha(X, M))$$

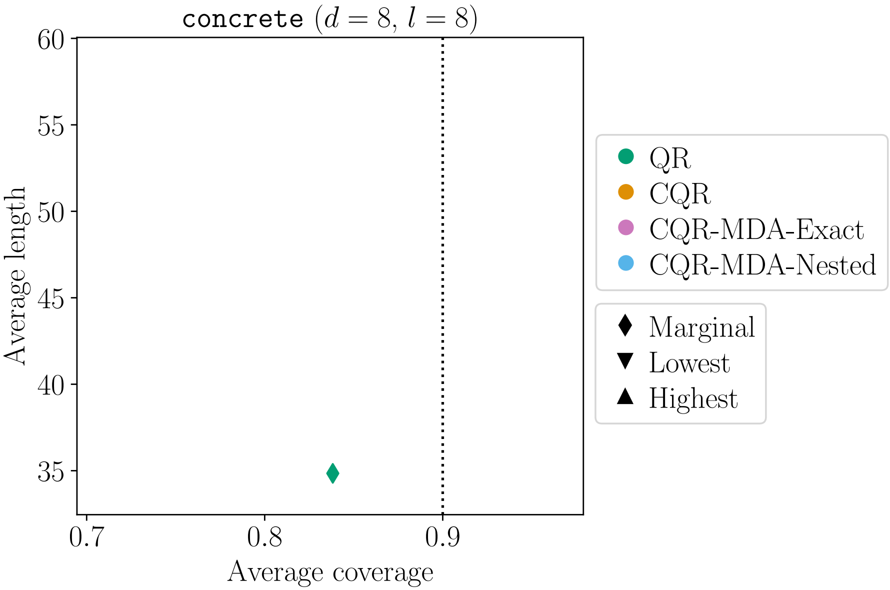
▼ : lowest coverage, i.e.

$$\min_{m \in \mathcal{M}} \mathbb{P}(Y \in \hat{C}_\alpha(X, m) | M = m)$$

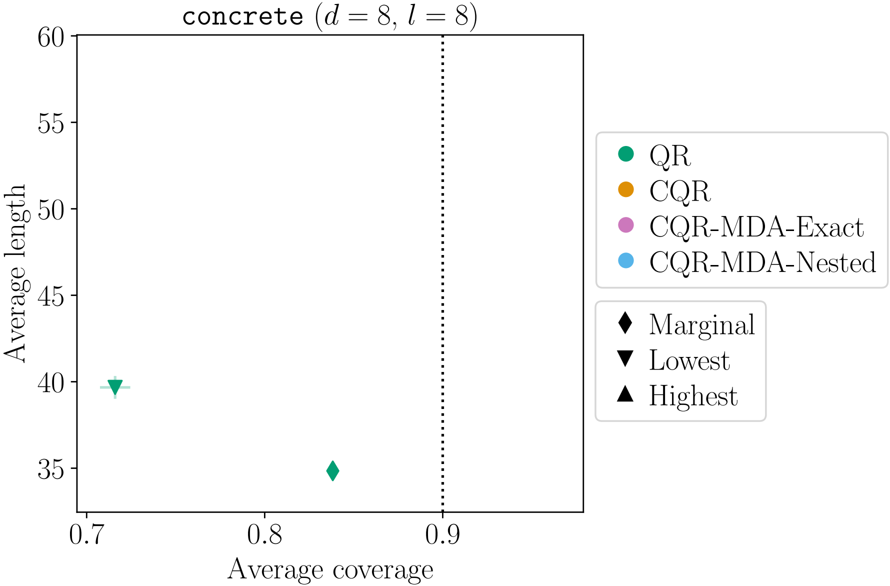
▲ : highest coverage, i.e.

$$\max_{m \in \mathcal{M}} \mathbb{P}(Y \in \hat{C}_\alpha(X, m) | M = m)$$

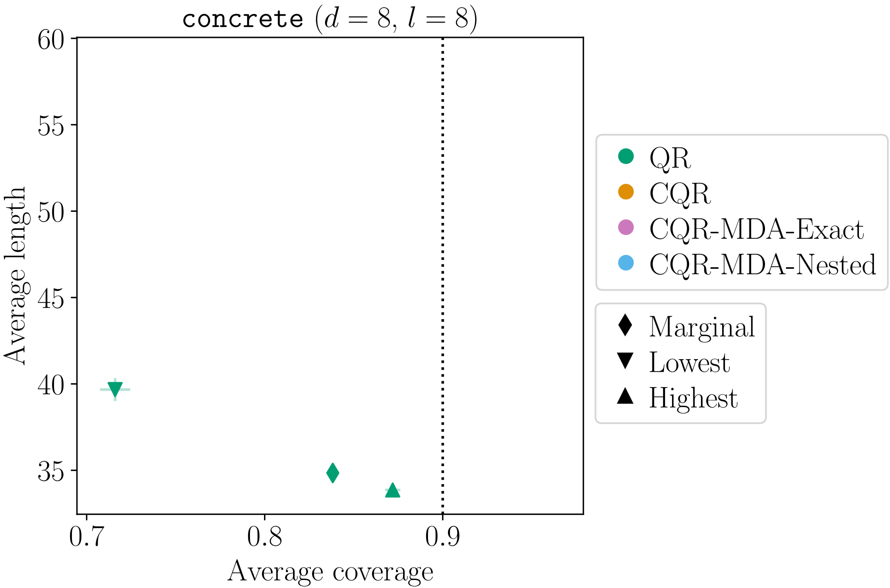
Semi-synthetic experiments



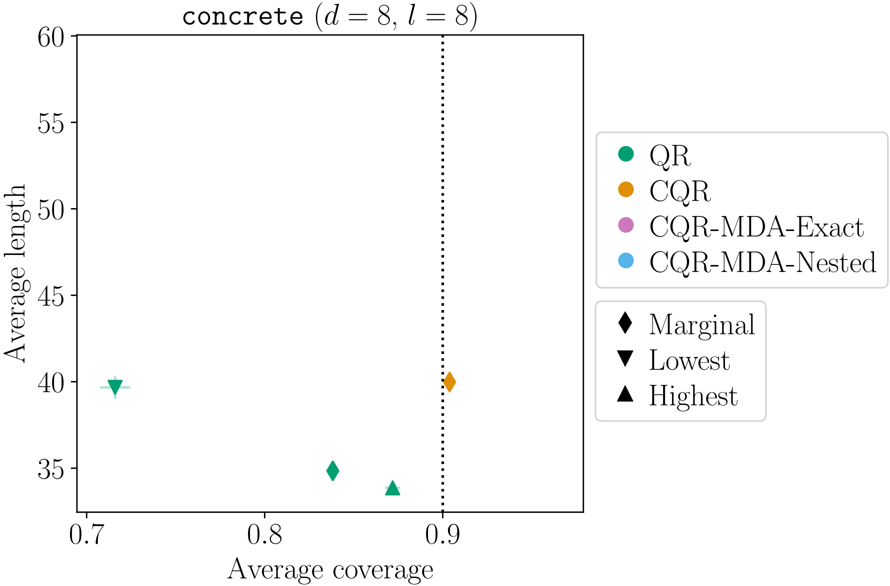
Semi-synthetic experiments



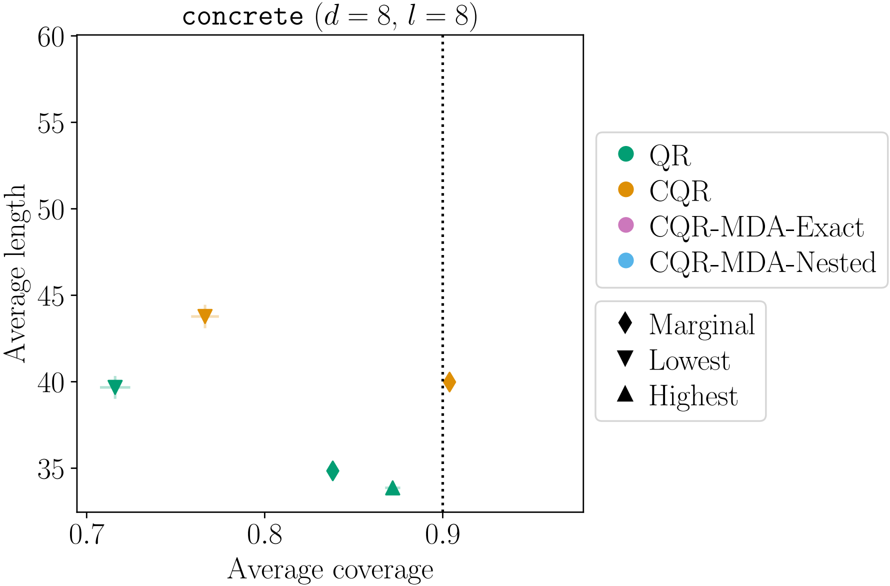
Semi-synthetic experiments



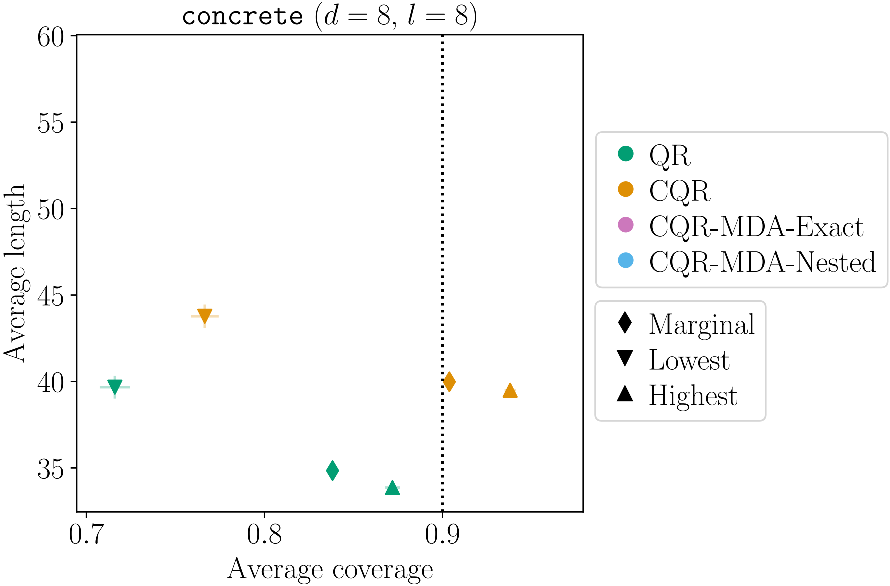
Semi-synthetic experiments



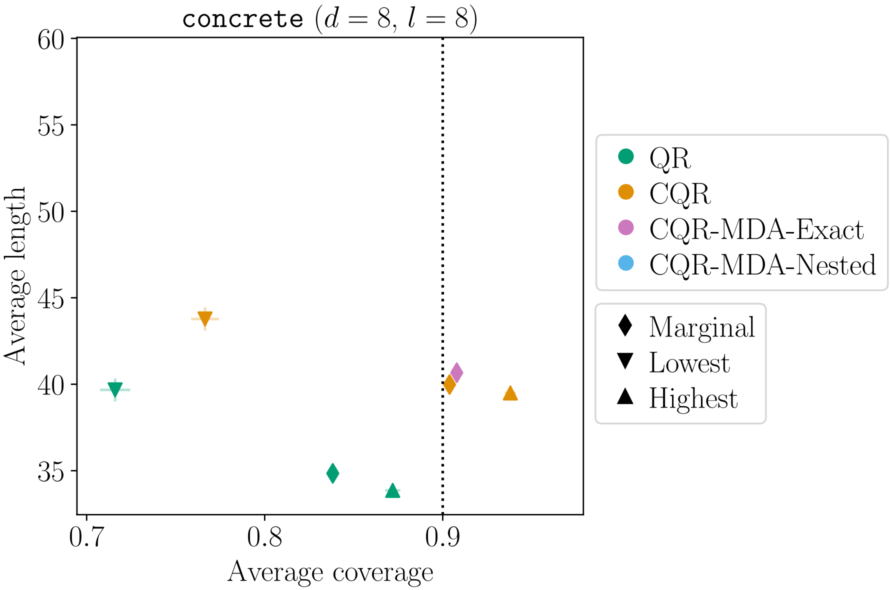
Semi-synthetic experiments

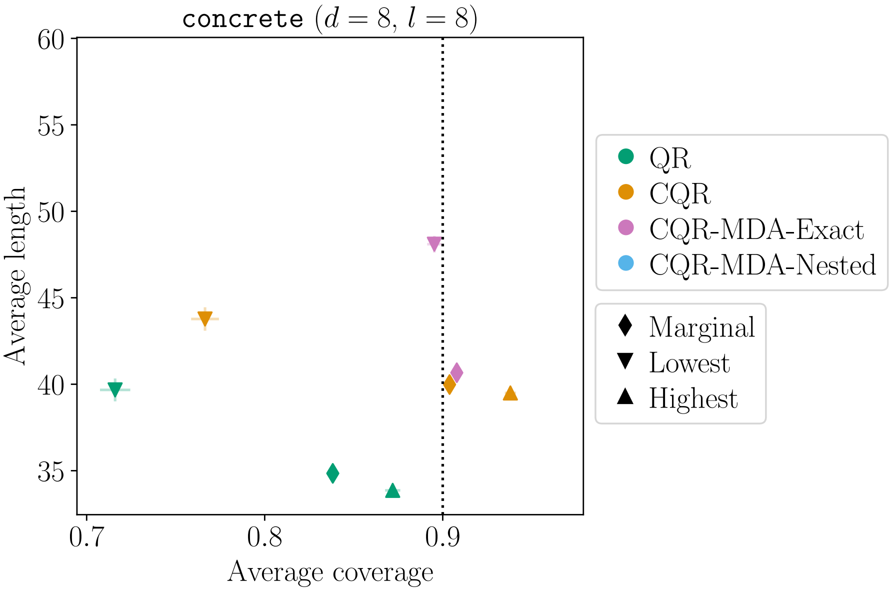


Semi-synthetic experiments



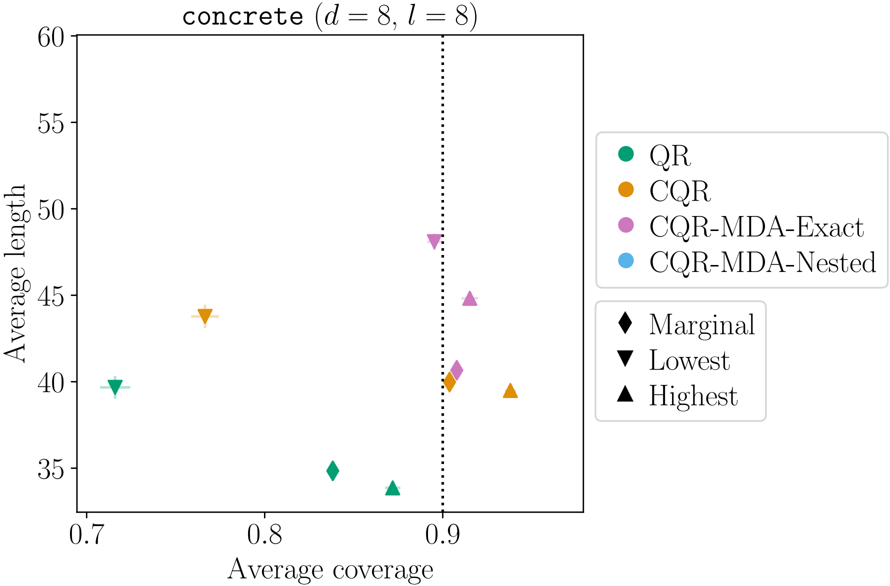
Semi-synthetic experiments



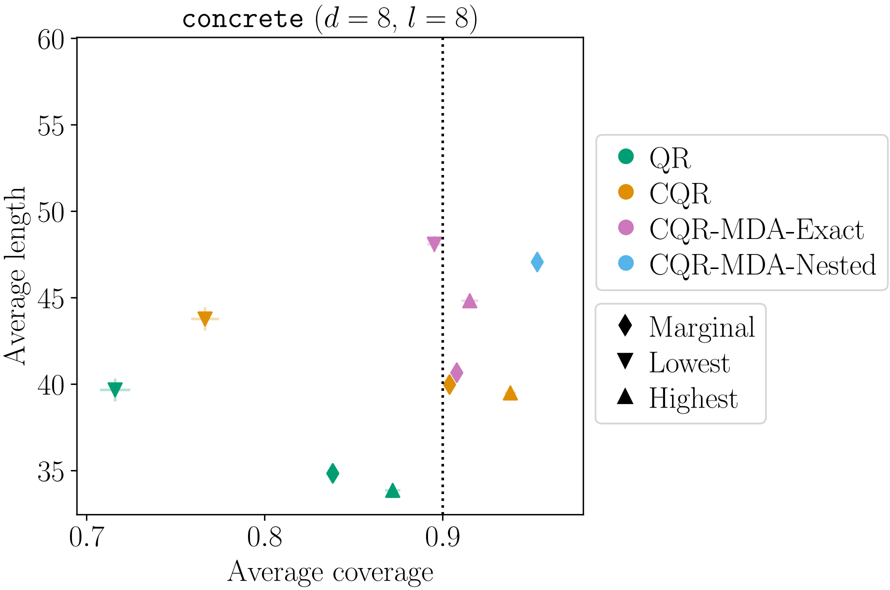


concrete ($d = 8$, $l = 8$)
60
55
50
45
40
35
Average length
0.7
0.8
0.9
Average coverage
QR
CQR
CQR-MDA-Exact
CQR-MDA-Nested
Marginal
Lowest
Highest

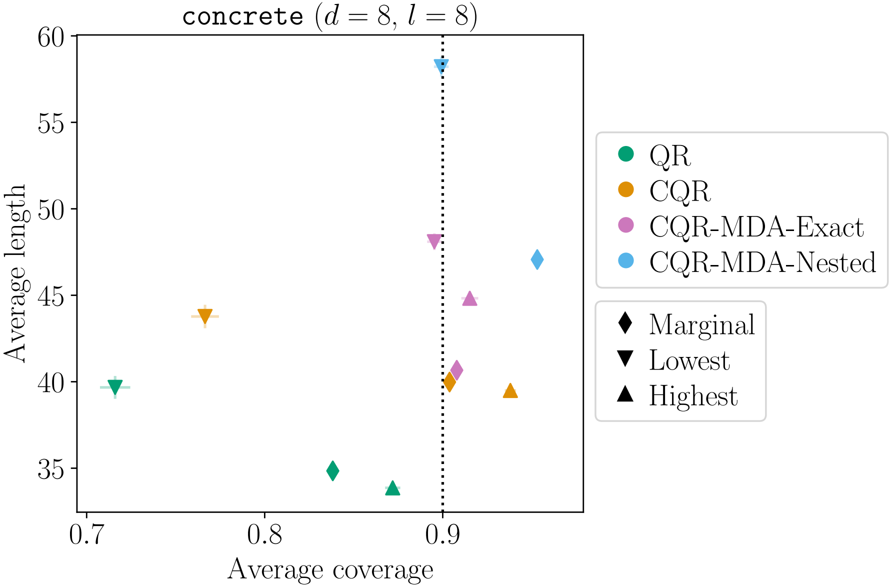
Semi-synthetic experiments



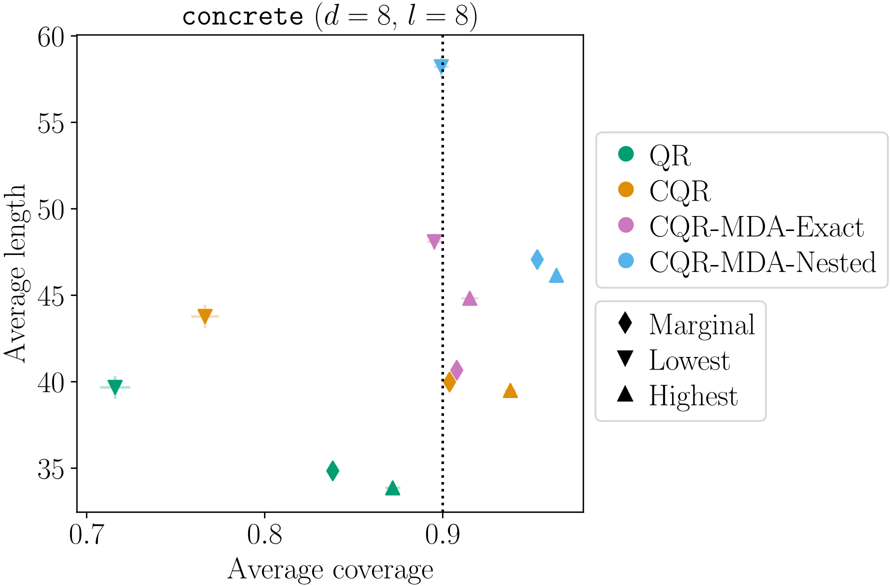
Semi-synthetic experiments



Semi-synthetic experiments



Semi-synthetic experiments

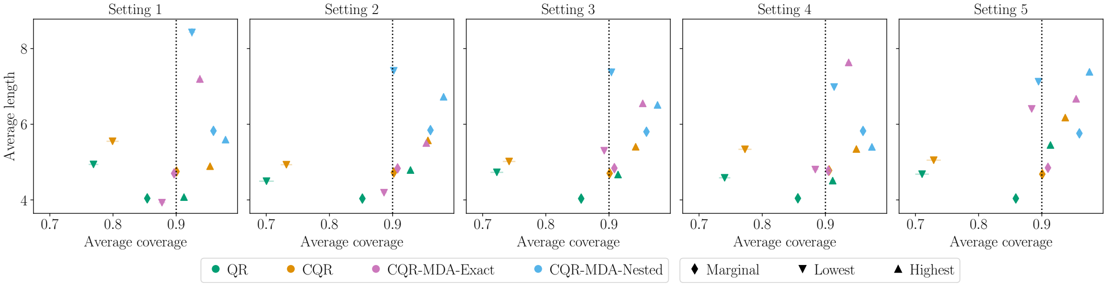


Simulation settings: beyond MCAR

- 6 variables (denote this set X_{missing}) out of 10 can be missing (the 4 others form the set X_{observed})
  $\rightarrow$ $X_{\text{missing}} = \{X_1, X_2, X_3, X_5, X_8, X_9\}$;
- Proportion of missing entries fixed to be 20%.

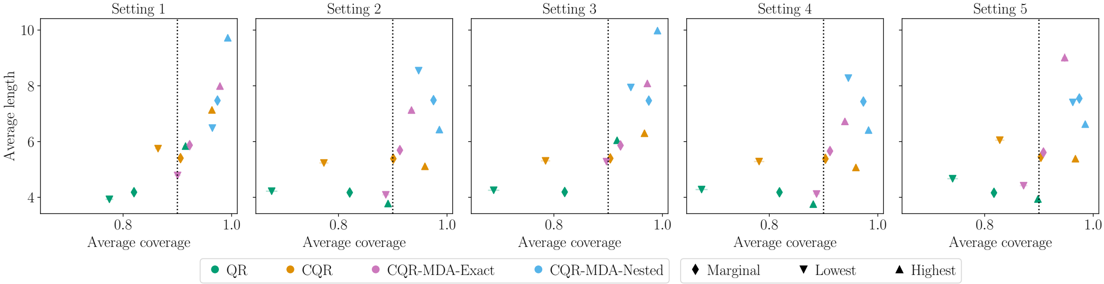
MAR missingness

- Probability of the variables in X_{missing} to be missing given by a logistic model of arguments X_{observed}.
- This setting is declined 5 times, with different weights for the logistic model.



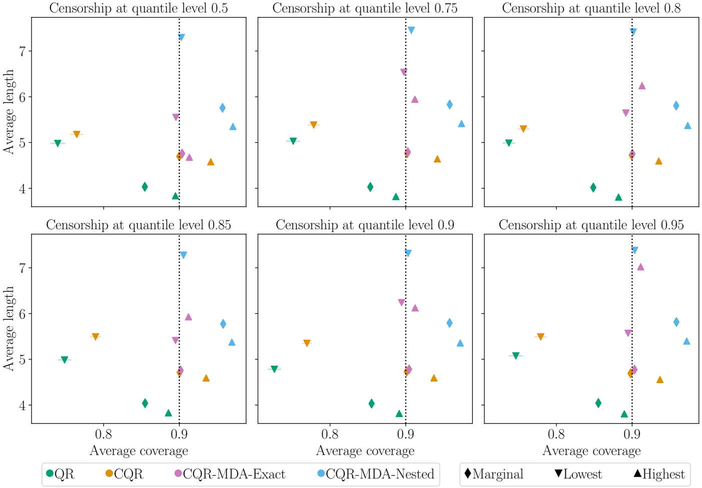
MNAR self masked missingness

- Probability of each variable in X_{missing} to be missing given by a logistic model of argument the same variable of X_{missing}.
- This setting is declined 5 times, with different weights for the logistic model.



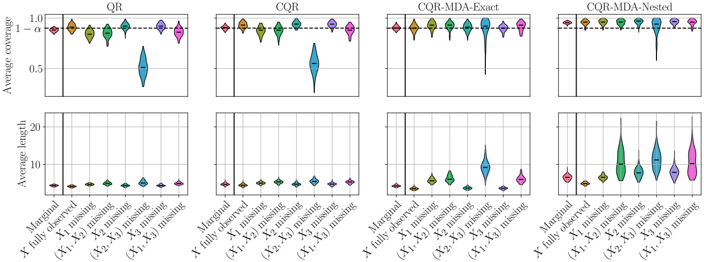
MNAR quantile censorship missingness

- Missing values are introduced at random in each q-quantile of the variables in X_{missing}.
- 6 different settings: q varies between 0.5, 0.75, 0.8, 0.85, 0.9 and 0.95.

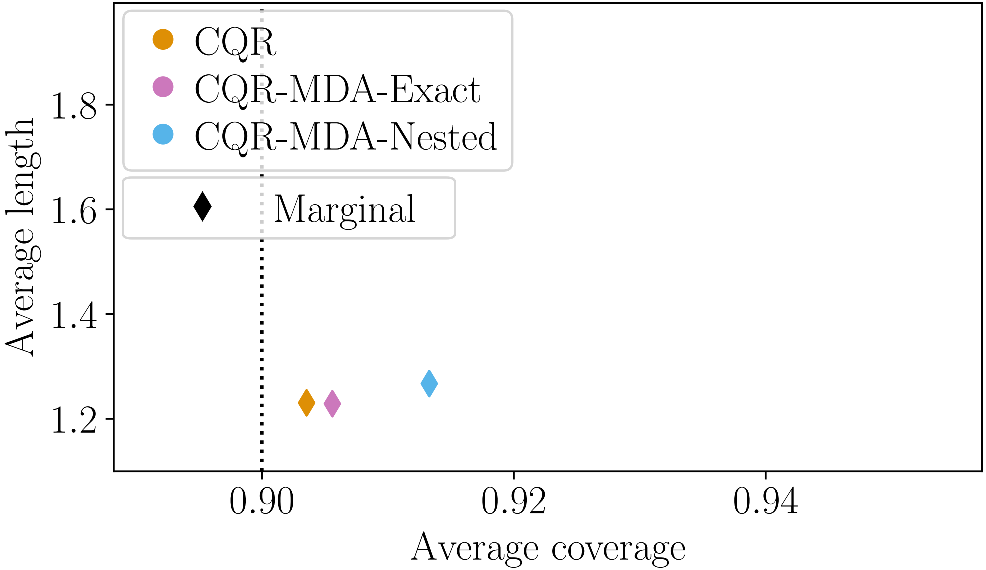


Experiments under $Y \not\perp\!\!\!\perp M \mid X$

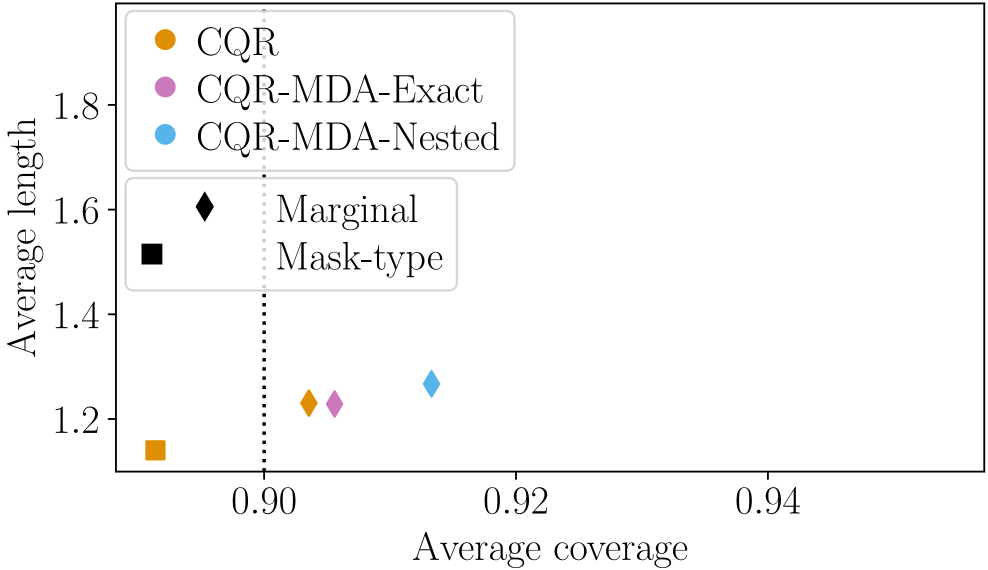
- $M_i \sim \mathcal{B}(0.2)$ for any $i \in [\![1, 3]\!]$, independently from X and ε
- $Y = X_1 \mathbb{1}\{M_1 = 0\} + 2X_1 \mathbb{1}\{M_1 = 1\} + 3X_2 \mathbb{1}\{M_2 = 1, M_3 = 1\} + \varepsilon$



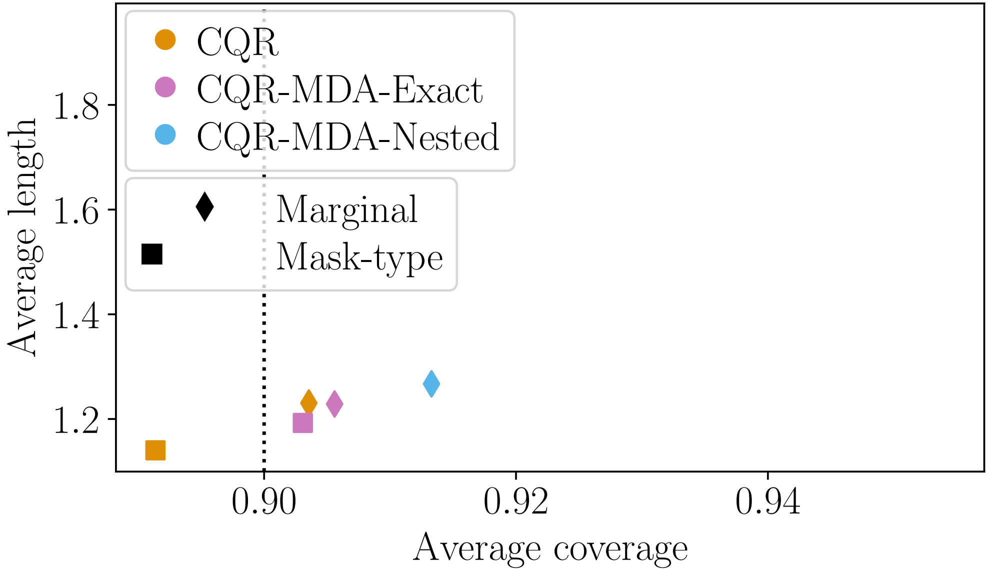
Real data experiment: TraumaBase®, critical care medicine



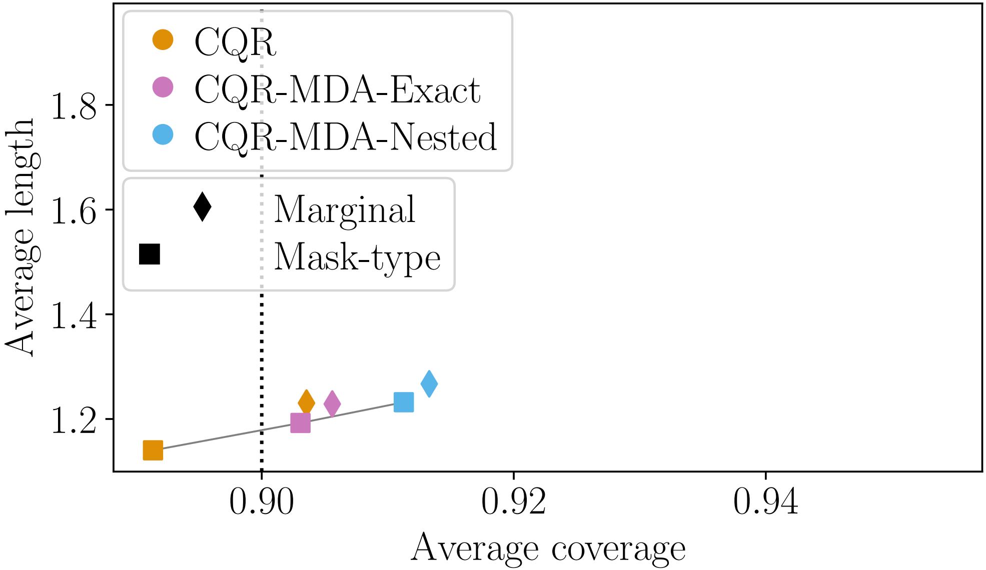
Real data experiment: TraumaBase®, critical care medicine



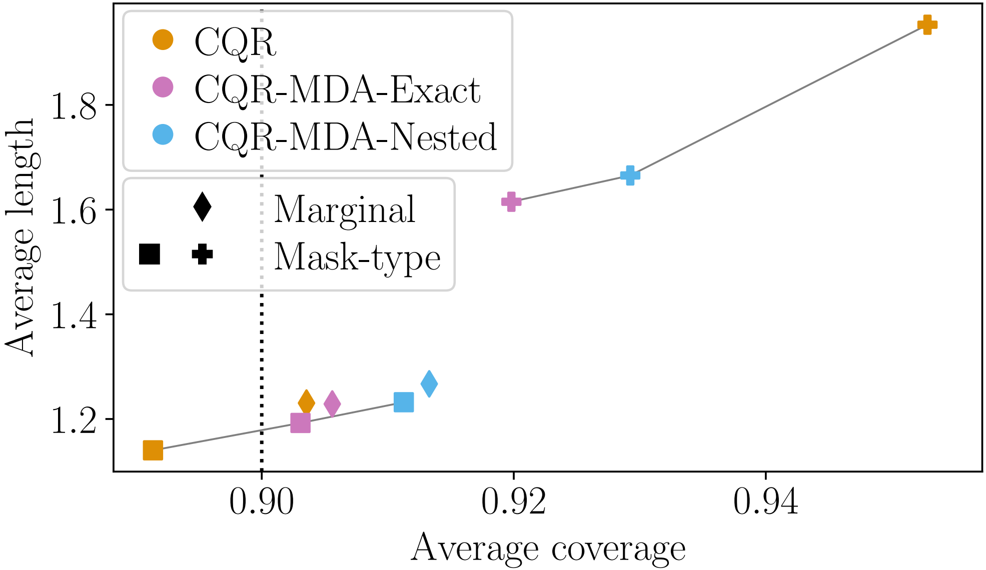
Real data experiment: TraumaBase®, critical care medicine

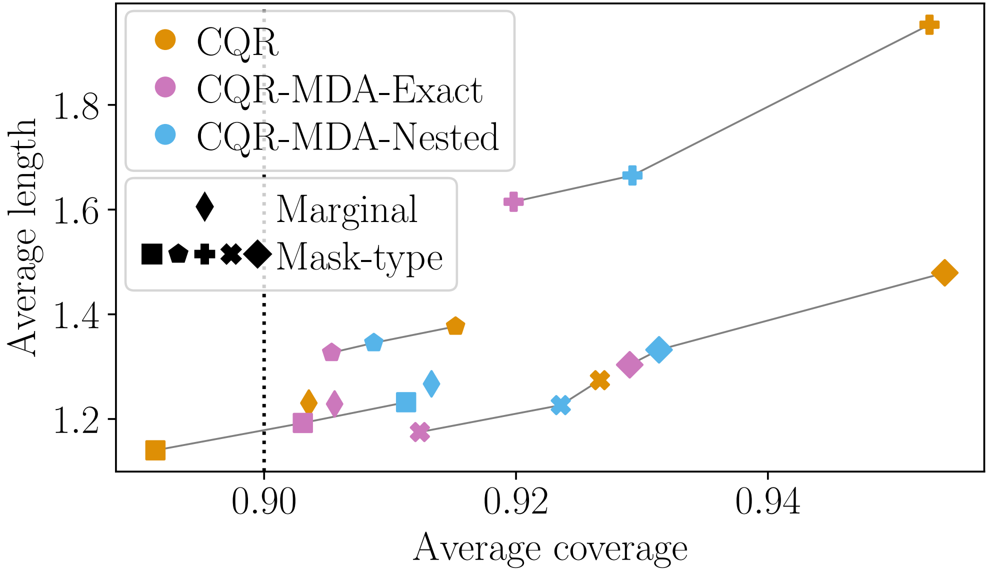


Real data experiment: TraumaBase®, critical care medicine



Real data experiment: TraumaBase®, critical care medicine




CQR
CQR-MDA-Exact
CQR-MDA-Nested
Marginal
Mask-type
Average length
1.2
1.4
1.6
1.8
Average coverage
0.90
0.92
0.94

Take-home-messages

- CP marginal guarantees hold on the imputed data set.

Take-home-messages

- CP marginal guarantees hold on the imputed data set.
- Missingness introduces additional heteroskedasticity and is a fun case to study shifts.

Take-home-messages

- CP marginal guarantees hold on the imputed data set.
- Missingness introduces additional heteroskedasticity and is a fun case to study shifts.
- CQR (and more generally CP) fails to attain coverage conditional on the missing pattern, i.e. MCV.

Take-home-messages

- CP marginal guarantees hold on the imputed data set.
- Missingness introduces additional heteroskedasticity and is a fun case to study shifts.
- CQR (and more generally CP) fails to attain coverage conditional on the missing pattern, i.e. MCV.
- Missingness introduces additional heteroskedasticity.

Take-home-messages

- CP marginal guarantees hold on the imputed data set.
- Missingness introduces additional heteroskedasticity and is a fun case to study shifts.
- CQR (and more generally CP) fails to attain coverage conditional on the missing pattern, i.e. MCV.
- Missingness introduces additional heteroskedasticity.
- MCV is impossible to ensure in an informative way without restricting both the dependence between M and X, and between M and Y.

Take-home-messages

- CP marginal guarantees hold on the imputed data set.
- Missingness introduces additional heteroskedasticity and is a fun case to study shifts.
- CQR (and more generally CP) fails to attain coverage conditional on the missing pattern, i.e. MCV.
- Missingness introduces additional heteroskedasticity.
- MCV is impossible to ensure in an informative way without restricting both the dependence between M and X, and between M and Y.
- `CP-MDA-Nested`* (Missing Data Augmentation) is the first method to output predictive intervals with missing values.

Take-home-messages

- CP marginal guarantees hold on the imputed data set.
- Missingness introduces additional heteroskedasticity and is a fun case to study shifts.
- CQR (and more generally CP) fails to attain coverage conditional on the missing pattern, i.e. MCV.
- Missingness introduces additional heteroskedasticity.
- MCV is impossible to ensure in an informative way without restricting both the dependence between M and X, and between M and Y.
- `CP-MDA-Nested`* (Missing Data Augmentation) is the first method to output predictive intervals with missing values.
- `CP-MDA-Nested`* attains conditional coverage with respect to the missing pattern (in MCAR and $Y \perp\!\!\!\perp M \mid X$ setting).

Take-home-messages

- CP marginal guarantees hold on the imputed data set.
- Missingness introduces additional heteroskedasticity and is a fun case to study shifts.
- CQR (and more generally CP) fails to attain coverage conditional on the missing pattern, i.e. MCV.
- Missingness introduces additional heteroskedasticity.
- MCV is impossible to ensure in an informative way without restricting both the dependence between M and X, and between M and Y.
- `CP-MDA-Nested`* (Missing Data Augmentation) is the first method to output predictive intervals with missing values.
- `CP-MDA-Nested`* attains conditional coverage with respect to the missing pattern (in MCAR and $Y \perp\!\!\!\perp M \,|X$ setting).
- `CP-MDA-Nested`* is empirically robust to non-MCAR scenarii.

References i

Ayme, A., Boyer, C., Dieuleveut, A., and Scornet, E. (2022). Near-optimal rate of consistency for linear models with missing values. *ICML*.

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

Cherubin, G., Chatzikokolakis, K., and Jaggi, M. (2021). Exact optimization of conformal predictors via incremental and decremental learning. In *Proceedings of the 38th International Conference on Machine Learning*. PMLR.

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

Le Morvan, M., Josse, J., Scornet, E., and Varoquaux, G. (2021). What's a good imputation to predict with missing values? *NeurIPS*.

Lei, J. (2019). Fast exact conformalization of the lasso using piecewise linear homotopy. *Biometrika*, 106(4).

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*.

Ndiaye, E. (2022). Stable conformal prediction sets. In *Proceedings of the 39th International Conference on Machine Learning*. PMLR.

Ndiaye, E. and Takeuchi, I. (2019). Computing full conformal prediction set with approximate homotopy. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc.

Ndiaye, E. and Takeuchi, I. (2022). Root-finding approaches for computing conformal prediction set. *Machine Learning*, 112(1).

Nouretdinov, I., Melluish, T., and Vovk, V. (2001). Ridge regression confidence machine. In *Proceedings of the 18th International Conference on Machine Learning*.

References iii

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

Vovk, V. (2015). Cross-conformal predictors. *Annals of Mathematics and Artificial Intelligence*, 74(1-2).

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

Zhu, Z., Wang, T., and Samworth, R. J. (2019). High-dimensional principal component analysis with heterogeneous missingness. arXiv.