A Complete Suite for Conformal Prediction of Simple and Complex Data in R, with some theoretical extensions

Course: Mathematical Engineering Academic Year: 2020-2021

Author: Paolo Vergottini, 946834
Advisor: Simone Vantini
Co-adivisors: Jacopo Diquigiovanni, Matteo Fontana, Aldo Solari

Outline

Conformal Prediction for Regression

Hp: training data $z_1 := (x_1, y_1), z_2 := (x_2, y_2), ..., z_n := (x_n, y_n) \sim i.i.d.$
Th: prediction set $C(x_{n+1}) : \mathbb{P}(y_{n+1} \in C(x_{n+1})) \geq 1 - \alpha$

1. Univariate response: $y \in \mathbb{R}$
2. Multivariate response: $y \in \mathbb{R}^q$
3. Multivariate functional response: $y \in \prod_{j=1}^{q} L^{\infty}(\tau_j)$, where τ_j is a closed and bounded subset of $\mathbb{R}^{d_j}$, $d_j \in \mathbb{N}_{>0}$

[9]

Study Aim

Python **libconform** (classification) and **nonconformist** (univariate)

R **conformalInference** (univariate)

Goal

1. Improve **conformalInference**
2. Extend conformal prediction theory to multivariate and functional response cases
3. build R packages for complex frameworks

Non-conformity Measure

Given a new point z_{n+1}, one can score how unusual it is w.r.t. $\{z_1, ..., z_n\}$ with

$$\mathcal{A}(\{z_1, ..., z_n\}, z_{n+1}) \in \bar{\mathbb{R}} \text{ where } \mathcal{A} \text{ measurable function}$$

For each regression framework we choose a suitable NCM

Conformal Prediction Methods

1. Full conformal
2. Split conformal
3. Jackknife+
4. Multi Split conformal
5. Conformalised Quantile Regression

Jackknife+

$$C_{jack+}(x_{n+1}) = [q_{\alpha}\{\hat{\mu}_{-i}(x_{n+1}) - R_i^{LOO}\}, q_{1-\alpha}\{\hat{\mu}_{-i}(x_{n+1}) + R_i^{LOO}\}]$$

$$\mathbb{P}(y_{n+1} \in C_{jack+}(x_{n+1})) \geq 1 - 2\alpha$$

How to translate the concept of quantile in multivariate and functional cases?

I need an order $\rightarrow$ non-conformity measure

[1]

Extended quantile

$$\mathcal{A}_{max}(x, y) = \sup_{j \in \{1,\dots,q\}} \left| \frac{y_j - [\hat{\mu}^j(x)]}{s^j} \right| \quad \text{(multivariate)}$$

$$= \sup_{j \in \{1,\dots,q\}} \left(\operatorname*{ess\,sup}_{t \in \tau_j} \left| \frac{y_j(t) - [\hat{\mu}^j(x_j)](t)}{s^j(t)} \right| \right) \quad \text{(functional)}$$

Extended quantile $q_\alpha^{\mathcal{A}}$ is the level set induced by the non-conformity measure $\mathcal{A}$

$$q_\alpha^{\mathcal{A}}(u_1, .., u_n) := \{u \in \mathcal{U} : \mathcal{A}_{max}(u) \leq q_{1-\alpha}\{\mathcal{A}_{max}(u_1), ..., \mathcal{A}_{max}(u_n)\}\}$$

Extended Jackknife+

$$C_{jack+}^{multi} = \{y \in \mathbb{R}^q : y \in [q_\alpha^{\mathcal{A}}(\{\hat{\mu}_{-i}(x_{n+1}) \pm R_i^{LOO} : i = 1, ..., n\})]\}$$

$$C_{jack+}^{fun} = \{y \in \prod_{j=1}^{q} L^\infty(\tau_j) : y(t) \in [q_\alpha^{\mathcal{A}}(\{\hat{\mu}_{-i}(x_{n+1}) \pm R_i^{LOO} : i = 1, ..., n\})(t)] \; \forall t \in \prod_{j=1}^{q} \tau_j\}$$

Finally, project on axes with Axes-Aligned Bounding Box

Multi Split Conformal

Input: split proportion vector *prop*, level $\alpha \in (0,1)$, and a regression algorithm $\mathcal{G}$, number of replications B, smoothing parameter λ, joining parameter τ

1. Repeat Split Conformal B times, with $\alpha_{split} = \alpha(1 - \tau + \lambda/B)$, obtaining $C^{[b]} \;\; b = 1, ..., B$
2. $\Pi^y = \frac{1}{B} \sum_{b=1}^{B} \mathbb{1}\{y \in C^{[b]}\} \; \forall y \in \mathbb{R}$
3. $C_{msplit}(x_{n+1}) = \{y \in \mathbb{R} \; : \; \Pi^y > \tau\}$

$$\mathbb{P}(y_{n+1} \in C_{msplit}(x_{n+1})) \geq 1 - \alpha$$

[6]

Multi Split Extension

How to join multiple prediction regions?

Extended quantile $q^{\mathcal{A}}_{\alpha_m}$, with $\alpha_m := 2\tau B$

2. $L = \{lo^{[b]}, up^{[b]} \; b = 1, ..., B\}$
3. $L_q = q^{\mathcal{A}}_{2\tau B}(L)$
4. $C_{msplit}(x_{n+1}) = BoundingBox(L_q)$

Packages

1. **conformalInference**
2. **conformalInference.multi**
3. **conformalInference.fd**

Package Structure

Structure:

- Regression methods
- Prediction methods
- Plot functions

Package Design

Regression methods not included into the prediction methods

```
conformal.multidim.full = function(x, y, x0, train.fun,
predict.fun,alpha = 0.1, mad.train.fun = NULL,
mad.predict.fun = NULL, score='l2', s.type = "st-dev",
num.grid.pts.dim=100, grid.factor=1.25, verbose=FALSE)
```

Function	Description
`conformal.pred.jack`	Jackknife+ prediction intervals
`conformal.pred.msplit`	Multi Split Conformal prediction intervals
`conformal.quant`	Full CQR prediction intervals
`conformal.quant.split`	Split CQR prediction intervals

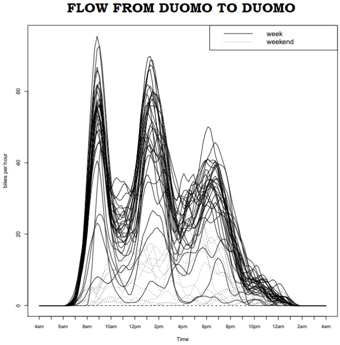
conformalInference.multi

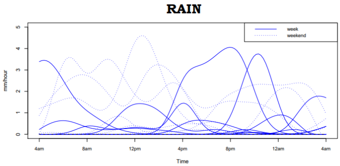
Function	Description
conformal.multidim.full	Full Conformal prediction regions
conformal.multidim.jackplus	Jackknife+ prediction regions
conformal.multidim.split	Split Conformal prediction regions
conformal.multidim.msplit	Multi Split Conformal prediction regions
elastic.funs	Build elastic net regression
lasso.funs	Build lasso regression
lm_multi	Build linear regression
mean_multi	Build regression functions with mean
plot_multidim	Plot the output of prediction methods
ridge.funs	Build elastic net regression

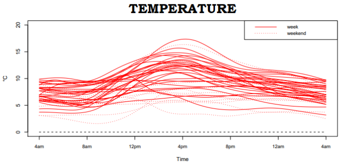
conformalInference.fd

Function	Description
concurrent	Build concurrent regression model
conformal.fun.jackplus	Jackknife+ prediction sets
conformal.fun.split	Split Conformal prediction sets
conformal.fun.msplit	Multi Split Conformal prediction sets
mean_lists	Build regression method with mean
plot_fun	Plot the output prediction methods

Example - BikeMi



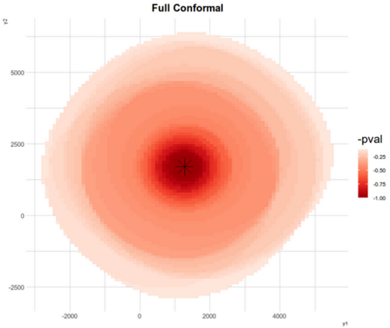




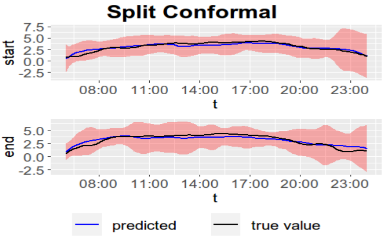
$$log(y_i^k(t)) = \beta_0^k(t) + \beta_{we}^k(t)x_{we,i}(t) + \beta_{rain}^k(t)x_{rain,i}(t) + \beta_{temp}^k(t)x_{dtemp,i}(t) + \beta_{we\_rain}^k(t)x_{we,i}(t)x_{rain,i}(t) + \epsilon_i^k(t) \quad k = 1,2 \quad i = 1,...,41$$

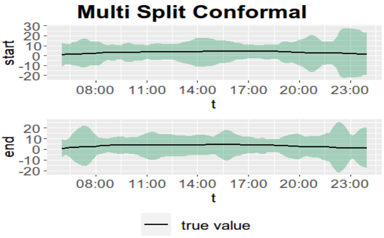
Example - Multivariate

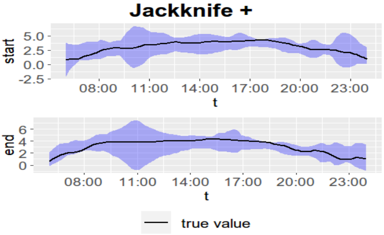
type	coverage	avg area	avg time (s)
full	0.90	$5.68 * 10^7$	35.3
split	0.98	$3.46 * 10^8$	0.01
msplit	0.90	$4.95 * 10^7$	1.80
jack	0.93	$6.56 * 10^7$	1.96



Example - Functional







type	coverage	average length	average time (s)
split	0.85	12.09	0.31
msplit	0.85	79.80	8.90
jack	0.63	3.22	4.66

Conclusions

- **conformalInference.multi** and **conformalInference.fd** on CRAN
- Increased the pool of conformal methods for R
- Extended Multi Split and Jackknife +

Next Steps

- **conformalInference** on CRAN
- Conformal tools for time-series analysis, as in [2]

References I

[1] Rina Barber et al. "Predictive inference with the jackknife+". In: *Annals of Statistics* 49.1 (2021), pp. 486–507.

[2] Jacopo Diquigiovanni, Matteo Fontana, and Simone Vantini. *Distribution-Free Prediction Bands for Multivariate Functional Time Series: an Application to the Italian Gas Market*. July 2021. URL: https://doi.org/10.48550/arXiv.2107.00527.

[3] Jonas Fassbender. *libconform v0.1.0: a Python library for conformal prediction*. 2019. URL: https://doi.org/10.48550/arxiv.1907.02015.

[4] Henrik Linusson. *nonconformist*. Python package version 2.1.0 (on GitHub). 2017. URL: https://github.com/donlnz/nonconformist.

References II

[5] Yaniv Romano, Evan Patterson, and Emmanuel J. Candès. "Conformalized Quantile Regression". In: *Advances in Neural Information Processing Systems*. Vol. 32. 2019. URL: https://doi.org/10.48550/arXiv.1905.03222.

[6] Aldo Solari and Vera Djordjilović. "Multi split conformal prediction". In: *Statistics Probability Letters* 184 (2022), p. 109395. ISSN: 0167-7152. URL: https://doi.org/10.1016/j.spl.2022.109395.

[7] Ryan Tibshirani. *conformalInference: Tools for conformal inference in regression*. R package version 1.1.0 (on GitHub). 2019. URL: https://github.com/ryantibs/conformal.

References III

[8] Agostino Torti, Alessia Pini, and Simone Vantini. "Modelling time-varying mobility flows using function-on-function regression: Analysis of a bike sharing system in the city of Milan". In: *Journal of the Royal Statistical Society.Series C: Applied Statistics* 70.1 (Nov. 2020), pp. 226–247. URL: https://doi.org/10.1111/rssc.12456.

[9] Gianluca Zeni, Matteo Fontana, and Simone Vantini. "Conformal Prediction: a Unified Review of Theory and New Challenges". In: *ArXiv* abs/2005.07972 (2020).