universität wien

MAGISTERARBEIT / MASTER'S THESIS

Titel der Magisterarbeit / Title of the Master's Thesis

„Distribution-Free Predictive Intervals for Quantile Forests“

verfasst von / submitted by

Christian Url

angestrebter akademischer Grad / in partial fulfilment of the requirements for the degree of

Magister der Sozial- und Wirtschaftswissenschaften (Mag. rer. soc. oec)

Wien, 2020 / Vienna 2020

Studienkennzahl lt. Studienblatt / degree programme code as it appears on the student record sheet:	UA 066 951
Studienrichtung lt. Studienblatt / degree programme as it appears on the student record sheet:	Magisterstudium Statistik
Betreut von / Supervisor:	ao. Univ.-Prof. Dipl.-Ing. Dr. Erhard Reschenhofer

Abstract

We propose a method that yields distribution-free prediction intervals for Quantile Regression Forests (pi-QF) that overcomes the deficiencies of these forests. To do so, we apply a split-conformal procedure to calibrate QRF estimation of the Inter Quartile Range (IQR). The proposed method stretches the upper and lower bounds separately. We prove the result and apply the algorithm on different artificial and real data sets. We thus provide empirical proof that our method improves the results from a standard QRF in terms of width, coverage and stability of the predictive intervals.

Contents

1 Introduction

Introduced by Meinshausen [2006], Quantile Regression Forests (QRF) can be seen as an extension to Random Forests (RF) because they give information about the spread of the response variable. By definition the former keep the value of all observations in a node for each node in each tree and evaluates the conditional distribution based on this information. In contrast, the latter only keep the mean of the observations and does not deploy this information. Furthermore, Random Forests have been shown to be accurate regressors that approximate the conditional mean of a response variable [Breiman, 2001]. Generalizing this approach, Quatile Regression Forests yield information about the conditional distribution function and thus conditional quantiles can be inferred accurately for high-dimensional predictor variables.

To understand the properties of the estimated intervals, it is helpful to take a closer look at the underlying procedure. The Random Forests algorithm grows an ensemble of trees, utilizing random split point and node selection [Amit and Geman, 1997]. The prediction of Random Forests could be referred to as an adaptive neighborhood classification and regression procedure [Lin and Jeon, 2006]. This means that for every $X = x$, a set of weights $w_i(x)$, $i = 1, \dots, n$ for the original n observations is obtained. The prediction of Random Forestsis equivalent to the weighted mean of the observed response variables. Applying this to Quantile Regression Forests, the conditional distribution function is estimated by the weighted distribution of response variables [Meinshausen, 2006]. However, no distiction is made between noise variables and variables that contain signal, as proposed in Breiman (2004). In accordance with the above discussion, Hothorn and Zeileis [2017] argue that for a quantile regression forest estimating the conditional 10% and 90% quantiles in a uniform setting the estimations vary from the true value and instabilities of these estimates can infer the intervals strongly.

To stabilize the quantile estimations we first fit a QRF estimating the IQR and second stretch these intervals using hold-out data to gain valid, distribution-free prediction intervals at level α. For the second step to be applicable we utilize conformal prediction as introduced by Vovk et al. [2005, 2009], the split conformal method by Papadopoulos et al. [2002], Vovk et al. [2005], Lei et al. [2015] and extensions to the previous by [Lei et al., 2017] who derive methods for distribution-free predictive inference in regression. The extensions and hence the split-conformal by Lei et al. [2015, 2017] are proven valid for any arbitrary regression algorithm.

Generally speaking the split-conformal method divides the underlying dataset D into two disjoint sets D_1 and D_2. Then one estimates the model on one part of the data and with the help of the second subset we evaluate how large a stretching factor or parameter must be such that we can then determine prediction intervals of level α. In Lei et al. [2017] the authors propose an additive constant C to return an interval for a prediction $\widehat{\mu}(x)$ that has the form $[\widehat{\mu}(x) - C, \widehat{\mu}(x) + C]$ and can be derived without any assumptions on the distribution of the data. The split-conformal method uses quantile regression to obtain $\widehat{\mu}$ and C. It is clear that this alone will not deliver stable and meaningful prediction intervals when the underlying model or algorithm is instable or inconstant, although conformal inference ensures that prediction intervals are valid to the α-level.

Instead of using an additive constant, we propose using multiplicative stretching factors for both the upper and lower bound. It is obvious that this procedure is also valid for only one expanding parameter due to conformal inference but we expect a higher stability in the case of asymmetrically distributed data which cannot be intercepted by the QRF. Using a multiplicative constant means that we expand the resulting IQR $[l(X), u(X)]$ using the estimated median $m(X)$ and factors c_l, c_u such that $\widehat{l}(X) = m(X) - \widehat{c}_l\,(m(X) - l(X))$ and $\widehat{u}(X) = m(X) + \widehat{c}_u\,(u(X) - m(X))$. These then deliver prediction bands $[\widehat{l}(X), \widehat{u}(X)]$ at level α. We call our proposed method *predictive invervals for Quantile Forests*, shorthand pi-QF, and denote it by $\widehat{\text{QF}}$.

2 From Linear Regression Towards Quantile Regression Forests

Linear Regression. Assume we have a response variable $Y \in \mathbb{R}^n$ and a predictor $X \in \mathbb{R}^{n \times p}$, $n, p \in \mathbb{N}$. In a linear regression setting $Y = X\tilde{\beta} + \varepsilon$, where $\tilde{\beta} \in \mathbb{R}^p$ and $\varepsilon \in \mathbb{R}^n$, the standard approach is to find an estimate $\widehat{\mu}(x) = \mathbb{E}(Y|X = x)$ minimizing the expected squared error loss

$$\mathbb{E}(Y|X = x) = \arg\min_{\tilde{\beta}} \mathbb{E}\left[\left(Y - X\tilde{\beta}\right)^2 \mid X = x\right].$$

Regression Tree. According to Hastie et al. [2009] and James et al. [2013]: Suppose we only observe independent samples $Z_i = (X_i, Y_i)$, with $Y \in \mathbb{R}^n$, $X \in \mathbb{R}^{n \times p}$, $n, p \in \mathbb{N}$ fixed, and want to build a CART regression tree. Recursively splitting the feature space leads to a set of leaves L, each of which only contains a few training samples. That means we divide the predictor space, i.e. the set of possible values for X_i, $i = 1, \ldots, p$ into J distinct and non-overlapping regions, $R_1, R_2, \ldots, R_J$. In other words, each leaf corresponds to a rectangular subspace of the image space of X with the goal to find rectangles that minimize the RSS. As this is computationally infeasible, a possible solution builds on recursive binary splitting. Therefore, we begin at the top of the tree where all observations are in one single leaf and then successively split the image space. At each splitting point, we perform a binary split that leads to the greatest possible reduction in RSS. Let θ be the random parameter determining how the tree is grown. Given a test point x, we evaluate the prediction $\widehat{\mu}(x)$ by identifying the leaf $L(x, \theta)$ which contains x and setting

$$\widehat{\mu}(x) = \frac{1}{i : X_i \in L(x, \theta)} \sum_{|\{i : X_i \in L(x,\theta)\}|} Y_i \tag{1}$$

Furthermore, using a weight vector

$$w_i(x, \theta) = \frac{\mathbb{1}_{\{X_i \in L(x,\theta)\}}}{\#\{j : X_j \in L(x, \theta)\}} \tag{2}$$

we can rewrite (1) to get the prediction of a single tree

$$\widehat{\mu}(x) = \sum_{i=1}^{n} w_i(x, \theta) Y_i.$$

Averaging the weights vector over a collection of k single trees constructed using an i.i.d. vector θ_t, $t = 1, \ldots, k$ we can find

$$w_i(x) = k^{-1} \sum_{t=1}^{k} w_i(x, \theta_t). \tag{3}$$

Random Forest. Suppose we have training samples $Z_i = (X_i, Y_i)$ for $i = 1, \ldots, n$ and a test point x. Furthermore, we have regression trees T which can be used to get estimates of the conditional mean function at x of the form $T(x; \xi, Z_1, \ldots, Z_n)$, with $\xi \sim \Xi$ auxiliary noise. The randomness in how to pick the splitting features is contained in the auxiliary random variable ξ. In practice, such a random forest is computed using Monte Carlo averaging:

$$\mathrm{RF}(x; Z_1, \ldots, Z_n) \approx \frac{1}{B} \sum_{b=1}^{B} T(x; \xi_b^*, Z_{b1}^*, \ldots, Z_{bn}^*). \tag{4}$$

$\{Z_{b1}^*, \ldots, Z_{bn}^*\}$ is drawn randomly without replacement from $\{Z_1, \ldots, Z_n\}$, ξ_b^* is a random draw from Ξ and B is the number of Monte Carlo replicates. It is recommended to take B on the order of n. The $B \to \infty$ limit of (4) then is the definition of a *random forest* with base learner T and subsample size s:

$$\mathrm{RF}(x; Z_1, \ldots, Z_n) = \binom{n}{s}^{-1} \sum_{1 \le i_1 < i_2, \ldots, i_s \le n} \mathbb{E}_{\xi \sim \Xi}[T(x; \xi, Z_{i_1}, \ldots, Z_{i_s})]. \tag{5}$$

Finally, using (3) we can derive the predictions of a random forest as a approximation of the conditional mean of Y which is a weighted sum over all observations [Breiman, 2001]:

$$\widehat{\mu}(x) = \sum_{i=1}^{n} w_i(x) Y_i. \tag{6}$$

Quantile Regression. Quantile regression on the other hand utilizes the idea of finding information beyond the mean of Y, such as the prediction of quantiles in salary, or the evaluation of students' scores

on a test [Koenker and Hallock, 2001]. Therefore, we define the conditional distribution function for $y \in \mathbb{R}$ as

$$F(y|X = x) = P(Y \leq y|X = x)$$

and for a continuous distribution function, the α-quantile

$$Q_\alpha = \inf \{y : F(y|X = x) \geq \alpha\}. \tag{7}$$

In this set-up, the aim is to estimate the conditional quantiles from data. For the optimization problem in quantile regression, define the loss function L_α for $0 < \alpha < 1$ by

$$L_\alpha(y, q) = \mathbb{1}_{\{y>q\}}(\alpha|y - q|) + \mathbb{1}_{\{y\leq q\}}((1 - \alpha)|y - q|)$$

and minimize the expected loss resulting in conditional quantiles:

$$Q_\alpha(x) = \arg\min_q \mathbb{E}\left[L_\alpha(Y, q) \mid X = x\right]$$

Quantile Regression Forest. "For Quantile Regression Forests, trees are grown as in the standard Random Forests algorithm. The conditional distribution is then estimated by the weighted distribution of observed response variables, where the weights attached to observations are identical to the original Random Forests algorithm." [Meinshausen, 2006]

That being said, we find that compared to random forests, that keep only the means of the observations in every leaf, quantile regression forests additionally keep information about the conditional distribution of the data in the leafs. The conditional distribution function of Y is defined by

$$F(y|X = x) = \mathbb{P}(Y \leq y|X = x) = \mathbb{E}\left[\mathbb{1}_{\{Y\leq y\}}|X = x\right].$$

Using the above equation and the discussion about random forests we can define an approximation to $\mathbb{E}\left[\mathbb{1}_{\{Y\leq y\}}|X = x\right]$ by

$$\widehat{F}(y|X = x) = \sum_{i=1}^{n} w_i(x)\mathbb{1}_{\{Y_i\leq y\}},$$

where the weights $w_i(x)$ are the same as for random forests, defined in (3). Finally, we apply the algorithm from Meinshausen [2006] and obtain estimates $\widehat{Q}_\alpha(x)$ of the conditional quantiles $Q_\alpha(x)$ by plugging $\widehat{\mathrm{F}}(y|X = x)$ instead of $\mathrm{F}(y|X = x)$ into (7).

Prediction Intervals with Quantile Regression Forests. Obtaining (7) in quantile regression, we can find prediction intervals for QRFs. In literature [Meinshausen, 2006] a 95% prediction interval is defined as $I(x) = [Q_{.025}(x), Q_{.975}(x)]$. Hence, for a new data point Y and $X = x$ the estimated IQR is

$$\Gamma(x) = [Q_{.25}(x), Q_{.75}(x)] = [l(x), u(x)]. \tag{8}$$

3 Theory of Conformal Inference

3.1 Introduction

For this section, consider a regression problem. The paper of Lei et al. [2017] shows strong results for conformal prediction sets on regression data. We utilize these results to get conformal prediction intervals for quantile regression forests. Suppose we have n i.i.d. observations

$$(X_1, Y_1), \ldots, (X_n, Y_n) \sim P,$$

where $X \in \mathbb{R}^{n\times p}$, $Y \in \mathbb{R}^n$ are random variables and $n, p \in \mathbb{N}$ are fixed. Let

$$\mu(x) = \mathbb{E}\left(Y \mid X = x\right), \quad x \in \mathbb{R}^d$$

denote the regression function. The goal is to predict a new response Y_{n+1} given new values X_{n+1} without any assumptions on μ or P. For a miscoverage level $\alpha \in (0,1)$, we want to construct a conformal prediction interval $C \in \mathbb{R}^d \times \mathbb{R}$ based on $(X_1, Y_1), \ldots, (X_n, Y_n)$ with the property that

$$\mathbb{P}\left(Y_{n+1} \in C(X_{n+1})\right) \geq 1 - \alpha, \tag{9}$$

where the probability is taken over the $n+1$ i.i.d. draws $(X_1, Y_1), \ldots, (X_n, Y_n), (X_{n+1}, Y_{n+1}) \sim P$. Furthermore, for $x \in \mathbb{R}^p$ we denote $C(x) = \{y \in \mathbb{R} : (x, y) \in C\}$. Lei et al. [2013] constructed prediction bands like in (9) utilizing a result about sample quantiles [Vovk et al., 2005, Shafer and Vovk, 2007, Vovk et al., 2009].

Definition 1. Let $U_1, \ldots, U_n$ be i.i.d. samples of a scalar random variable. Given $\alpha \in (0,1)$, define the sample quantile $\widehat{q}_{1-\alpha}$ based on $U_1, \ldots, U_n$ by

$$\widehat{q}_{1-\alpha} = \begin{cases} U_{(\lceil (n+1)(1-\alpha) \rceil)} & \text{if } \lceil (n+1)(1-\alpha) \rceil \leq n \\ \infty & \text{otherwise,} \end{cases}$$

and $U_{(1)} \leq \ldots \leq U_{(n)}$ denote the order-statistics of $U_1, \ldots, U_n$.

Theorem 1. *For the setting given in Definition 1 and another i.i.d. sample U_{n+1} we have that*

$$\mathbb{P}\left(U_{n+1} \leq \widehat{q}_{1-\alpha}\right) \geq 1 - \alpha. \tag{10}$$

Proof. Independence means that the ranks are uniform on the set. Then the rank of of the new observation has the desired probability of being smaller than $(1-\alpha)(n+1)$, which implies the result. □

Transferring these properties to the regression problem, we might form a naive prediction interval by

$$C_{naive}(X_{n+1}) = \left(\widehat{\mu}(X_{n+1}) - \widehat{\mathrm{F}}_n^{-1}(1-\alpha),\ \widehat{\mu}(X_{n+1}) + \widehat{\mathrm{F}}_n^{-1}(1-\alpha)\right). \tag{11}$$

In this equation $\widehat{\mu}$ denotes an estimate for the underlying regression function and $\widehat{\mathrm{F}}_n$ the empirical distribution function fitted on the residuals $R_i = |Y_i - \widehat{\mu}(X_i)|$, $i = 1, \ldots, n$ and $\widehat{\mathrm{F}}_n^{-1}(1-\alpha)$ the $(1-\alpha)$-quantile of $\widehat{\mathrm{F}}_n$ [Lei et al., 2017]. The naive method looks similar to our desired result, but it is only valid for large samples. Furthermore, we have to assure that the estimated regression function $\widehat{\mu}$ is accurate, which may only be yielded by requiring additional and appropriate regularity conditions on P and $\widehat{\mu}$. In general, these deficiencies lead to miscoverage as the residual distribution can be biased.

There are four approaches to overcome these deficiencies. *Conformal Prediction Intervals* deliver finite-sample coverage without any assumptions on P and $\widehat{\mu}$ [Vovk et al., 2005, 2009, Lei and Wasserman, 2014, Lei et al., 2013, 2017]. But the algorithm is computationally intensive as for any X_{n+1} and Y_{n+1} the model has to be retrained on the augmented data set including the new point (X_{n+1}, Y_{n+1}) and then the absolute residuals have to be recomputed and reordered.

These problems motivate a method called *Split Conformal Prediction Sets* [Papadopoulos et al., 2002, Vovk et al., 2005, Lei et al., 2015, 2017] which is computationally far less intensive because we only have fit once on D_1 as the ranking step is separated. This method allows to perform full conformal inference in high-dimensional regression and is the core of our algorithm and therefore we will discuss it here in full length.

3.2 Split Conformal Prediction Sets

Algorithm 1 was developed by Lei et al. [2017] as an adaption from Lei et al. [2015, Algorithm 1] based on the idea of Vovk et al. [2005]. Theorem 2 states the key coverage properties of this method.

Theorem 2. *If (X_i, Y_i), $i = 1, \ldots, n$ are i.i.d., then for a new i.i.d. draw (X_{n+1}, Y_{n+1}),*

$$\mathbb{P}\left(Y_{n+1} \in C_{split}(X_{n+1})\right) \geq 1 - \alpha,$$

Algorithm 1: Split Conformal Prediction

Input: Data (X_i, Y_i), $i = 1, \ldots, n$, miscoverage level $\alpha \in (0, 1)$, regression algorithm $\mathcal{A}$
Output: Prediction band over $x \in \mathbb{R}^d$
Randomly split $\{1, \ldots, n\}$ into two equally sized subsets $\mathcal{I}_1, \mathcal{I}_2$
Set: $\widehat{\mu} = \mathcal{A}\left(\{(X_i, Y_i) : i \in \mathcal{I}_1\}\right)$
Set: $R_i = |Y_i - \widehat{\mu}(X_i)|$, $i \in \mathcal{I}_2$
Set: $d = R_{(k)}$, $k = \lceil (|D_2| + 1)(1 - \alpha) \rceil$ and $R_{(k)}$ the k-th order-statistic of $\{R_i : i \in \mathcal{I}_2\}$
Return: $C_{\text{split}}(x) = [\widehat{\mu}(x) - d,\ \widehat{\mu}(x) + d]$ for all $x \in \mathbb{R}^d$

for the split conformal prediction band C_{split} constructed in Algorithm 1. Moreover, if we assume additionally that the residuals R_i, $i \in \mathcal{I}_2$ have a continuous joint distribution, then

$$\mathbb{P}\left(Y_{n+1} \in C_{split}(X_{n+1})\right) \leq 1 - \alpha + \frac{1}{|D_2| + 1}.$$

Proof. We find that

$$\begin{aligned} \{Y_{n+1} \in C(X_{n+1})\} &\Leftrightarrow \left\{R_{(n+1)} \leq R_{(k)}\right\} \\ &\Leftrightarrow \{\text{Unif}(n+1) \leq k\}. \end{aligned}$$

Then,

$$\begin{aligned} \mathbb{P}\left(\text{Unif}(n+1) \leq k\right) &= \mathbb{P}\left(\text{Unif}(n+1) \leq \lceil (|D_2| + 1)(1 - \alpha) \rceil\right) \\ &\leq \frac{\lceil (|D_2| + 1)(1 - \alpha) \rceil}{|D_2| + 1} \\ &\leq 1 - \alpha + \frac{1}{|D_2| + 1}. \end{aligned}$$

□

In this thesis Algorithm 1 is adapted in several ways, yielding competitors for our proposed method. First, we extend a random forest using the split conformal to obtain a prediction band. In another step, we use a single stretch parameter to widen the predicted IQR from a quantile forest with the aim to get $1 - \alpha$ coverage. Our proposed method, the pi-forest $\widehat{\mathrm{F}}$, uses a split conformal algorithm that returns two stretching parameters, one for the upper and another lower bound, respectively. We state that these prediction bands are empirically more stable and tighter than the benchmark intervals. We define this procedure in Algorithm 2 and prove that we have $1 - \alpha$ average coverage for the prediction intervals.

4 Prediction Intervals for Quantile Forests

Let QF be a quantile forest. The data D is independently split into two parts, D_1 and D_2. We fit QF on D_1. The estimates are denoted by $\text{QF}_{D_1}(x) = (l(x), m(x), u(x))$ where l is the lower bound of the interval, m the median and u the upper bound. As indicated by the notation all depend on D_1. The estimated IQR is $\Gamma(x) = [l(x), u(x)]$.

Let Ω denote the set of all events for the new observations $(X, Y) \in D_2$. Furthermore, we note that for these random variables either $\{Y \leq m(X)|D_1\}$ or $\{Y > m(X)|D_1\}$ is true. For notational simplicity we drop the condition on D_1 statement in the formulation as it becomes clear that $m(X)|D_1$ is a fixed function whereas $m(X)$ is random. By definition, we have that $l(X) \leq m(X) \leq u(X)$ for all X and hence if $Y \in \Gamma(X)$ then $l(X) \leq Y \leq u(X)$.

By our assumption we have that D_1 is independent of D_2. We can partition D_2 into $D_2 = D_2^l \cup D_2^u$. Recall that $m(X)$ is random and depends on D_1. Fix D_1 and define

$$\begin{aligned} D_2^l &:= \{(X_i, Y_i) \in D_2 : Y_i \leq m(X_i)\} \overset{i.i.d.}{\sim} P_l \\ D_2^u &:= \{(X_i, Y_i) \in D_2 : Y_i > m(X_i)\} \overset{i.i.d.}{\sim} P_u, \end{aligned}$$

where P_l is the conditional distribution of $\{(X,Y) \in D_2 : Y \leq m(X)\}$ and P_u the conditional distribution of $\{(X,Y) \in D_2 : Y > m(X)\}$, respectively.

Definition 2. For all $(X_i, Y_i) \in D_2$, we calculate the stretching parameters

$$c_{l,s} = \left| \frac{m(X_i) - Y_i}{l(X_i) - m(X_i)} \right| \quad \text{if} \quad (X_i, Y_i) \in D_2^l, \quad \text{or}$$
$$c_{u,t} = \left| \frac{m(X_i) - Y_i}{u(X_i) - m(X_i)} \right| \quad \text{if} \quad (X_i, Y_i) \in D_2^u.$$

Definition 3. For the vectors of stretching parameters from Definition 2, let $\widehat{c}_l$ be the k-th order statistic of $c_{l,s}$, $s \in 1, \ldots, |D_2^l|$. Formally, set $\widehat{c}_l = c_{l_{(k)}}$, where $k = \lceil (1-\alpha) \times \left(\left|D_2^l\right| + 1\right) \rceil$. Set $\widehat{c}_u$ analogously using $t \in 1, \ldots, |D_2^u|$. Then we define the conformal bounds of the prediction interval for new observations $(X,Y) \sim P$ by

$$\widehat{l}(X) \coloneqq m(X) - \widehat{c}_l \left(m(X) - l(X)\right) \quad \text{and}$$
$$\widehat{u}(X) \coloneqq m(X) + \widehat{c}_u \left(u(X) - m(X)\right).$$

Algorithm 2: Split Conformal Intervals for Quantile Forests

Input: Holdout data D_2, QF $\mathrm{QF}_{D_1}(x) = (l(x), m(x), u(x))$, coverage probability $1-\alpha$.
Output: Scaled pi-QF $\widehat{\mathrm{QF}}$ using expansion parameters $\widehat{c}_l$, $\widehat{c}_u$.
Set: $D_2^l = \{(X_i, Y_i) \in D_2 : Y_i \leq m(X_i)\}$, $D_2^u = \{(X_i, Y_i) \in D_2 : Y_i > m(X_i)\}$.
Set: $c_{l,s} = \left|\frac{m(X_i)-Y_i}{l(X_i)-m(X_i)}\right|$ forall $(X_i, Y_i) \in D_2^l$ and $c_{u,t} = \left|\frac{m(X_i)-Y_i}{u(X_i)-m(X_i)}\right|$ forall $(X_i, Y_i) \in D_2^u$.
Set: $\widehat{c}_l = c_{l_{(k)}}$, $k = \lceil (1-\alpha) \times \left(\left|D_2^l\right| + 1\right) \rceil$ and $c_{l_{(k)}}$ the k-th order statistic and
Set: $\widehat{c}_u = c_{u_{(k)}}$, $k = \lceil (1-\alpha) \times (|D_2^u| + 1) \rceil$ and $c_{u_{(k)}}$ the k-th order statistic.
Set: $\widehat{l}(X) = m(X) - \widehat{c}_l \left(m(X) - l(X)\right)$ and $\widehat{u}(X) = m(X) + \widehat{c}_u \left(u(X) - m(X)\right)$ for new observations $(X,Y) \sim P$.
Return: $\widehat{\mathrm{QF}} = (\widehat{l}(X), m(X), \widehat{u}(X))$

Theorem 3. *Let* $\widehat{\mathrm{QF}} = \left(\widehat{l}(X), m(X), \widehat{u}(X)\right)$ *be the result of Algorithm 2 and let* $\widehat{\Gamma}(X) = \left[\widehat{l}(X),\, \widehat{u}(X)\right]$ *where* $\widehat{l}(X)$, $\widehat{u}(X)$ *are given in Definition 3. Then*

$$\mathbb{P}\left(Y \in \widehat{\Gamma}(X)\right) \geq 1 - \alpha.$$

Proof. We already know D_1 is independent of D_2 and that $c_{l,i}$ are i.i.d. conditional on D_1 as $\{c_{l,j} : (X_j, Y_j) \in D_2^l\}$. We argue for $c_{u,i}$ analogously. For new observations (X,Y) calculate

$$c_l' = \left| \frac{m(X) - Y_i}{l(X) - m(X)} \right|, \quad c_u' = \left| \frac{m(X) - Y_i}{u(X) - m(X)} \right|.$$

Conditional on D_1, c_l' is independent of $c_{l,i}$, has the same distribution as $c_{l,n}$ and the ranks of c_l' are uniform among $\{1, 2, \ldots, |D_2^l| + 1\}$. Analogously for c_u'. Note that $\left\{Y < \widehat{l}(X)\right\} \subseteq \{Y \leq m(X)\}$ and $\{Y > \widehat{u}(X)\} \subseteq \{Y > m(X)\}$ by Definition 3 and the fact that $\widehat{l}(X) \leq m(X) \leq \widehat{u}(X)$. Then we find that

$$\begin{aligned}
\mathbb{P}\left(Y \notin \widehat{\Gamma}(X) | Y \leq m(x)\right) &= \mathbb{P}\left(Y \leq \widehat{l}(X) | Y \leq m(x)\right) \\
&= \mathbb{E}\left[\mathbb{P}\left(Y \leq \widehat{l}(X) \,\middle|\, Y \leq m(x),\, D_1\right)\right] \\
&= \mathbb{E}\left[\mathbb{P}\left(c_l' > c_{l_{(k)}} \,\middle|\, D_1\right)\right] \\
&\overset{\text{Thrm 1}}{\leq} \mathbb{E}\left[\alpha\right] \\
&= \alpha.
\end{aligned}$$

In line 3 we have $Y \leq m(X)$, then compute c_l' and compare this to $c_{l_{(k)}}$. Hence,

$$\mathbb{P}\Big(Y \leq \widehat{l}(X) \,|\, Y \leq m(x)\Big) = \mathbb{P}\Big(c_l' > c_{l_{(k)}}\Big).$$

For non-containment on the upper bound analogously

$$\begin{aligned}
\mathbb{P}\Big(Y \notin \widehat{\Gamma}(X) | Y \geq m(x)\Big) &= \mathbb{P}\Big(Y \leq \widehat{u}(X) | Y \geq m(x)\Big) \\
&= \mathbb{E}\Big[\mathbb{P}\Big(Y \leq \widehat{u}(X) \,\Big|\, Y \geq m(x),\, D_1\Big)\Big] \\
&= \mathbb{E}\Big[\mathbb{P}\Big(c_u' > c_{u_{(k)}} \,\Big|\, D_1\Big)\Big] \\
&\overset{\text{Thrm 1}}{\leq} \mathbb{E}\left[\alpha\right] \\
&= \alpha.
\end{aligned}$$

Using the above we find

$$\mathbb{P}\left(Y \in \widehat{\Gamma}(X) \mid D_1\right) \geq 1 - \alpha$$

and integrating out D_1 gives the desired result. □

Remark 3.1. *We can simplify Algorithm 2 and Theorem 3 to a procedure using only one stretching parameter c^*. This reduces the proposed method to the result of Kivaranovic et al. [2019]. Then, according to Definition 2, we calculate*

$$c_n^* = \max\left(\left|\frac{m(X_i) - Y_i}{m(X_i) - l(X_i)}\right|, \left|\frac{m(X_i) - Y_i}{u(X_i) - m(X_i)}\right|\right) \quad \textit{for all} \quad (X_i, Y_i) \in D_2, \tag{12}$$

and by plugging Equation (12) into Definition 3, i.e. using c_n^* instead of $c_{u,n}$ and $c_{l,m}$, we yield $\widehat{\mathrm{QF}^*}(X) = \Big(\widehat{l^*}(X), m(X), \widehat{u^*}(X)\Big)$. It is clear that Theorem 3 also holds for $\widehat{\mathrm{QF}^*}(X)$.

5 Simulated Data

5.1 Data Generating Processes

For simulation, we use three different data generating processes, each with homoskedastic, heteroskedastic and asymmetric noise, respectively.

Additive Regression Data. First, we take a simple model $Y \sim X + \varepsilon$ where we are given the features X. In the default setting, we have 10 covariates each normally distributed with mean 0 and variance 1.

Polynomial Regression Data. Here we use a regression model $Y \sim X + \varepsilon$ but the true parameter space contains polynomials of the first 10 covariates.

Nonlinear Function after Selection. The data is simulated according to $Y = f(\beta' X) + \epsilon, \epsilon \sim N(0, 1 + (\beta' X)^2)$ where $f(x) = 2 * sin(\pi x) + \pi x$, each column in X follows a uniform distribution on $[0, 1]$ and only the first 4 components of $\beta \in \{0, 1\}^p$ are non-zero and equal to 1. Complexity rises with p, the number of variables, because the forest has to work in a very sparse setting [Kivaranovic et al., 2019]. According to the two previous settings, we use $p = 10$ in the following.

5.2 Results

We can compare the estimated prediction intervals to the true bands, because an oracle described the data-generating processes to us. To smooth the prediction intervals for the plots we used a local smoother `geom_smooth()` from the *tidyverse*-Package in R [Wickham, 2017]. The local smoother is a generalized additive model performing cubic regression spline based smooths with a penalized smoothing basis from

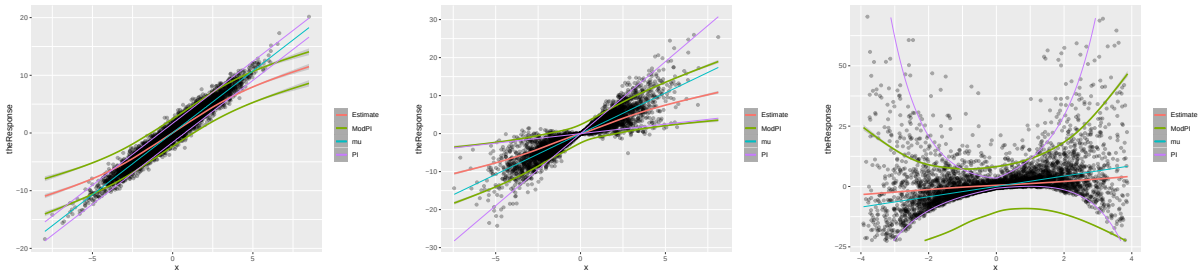

Figure 1: PI for additive regression data, left: homoskedastic noise, middle: heteroskedastic noise, right: asymmetric noise, plotted 95% of the data.

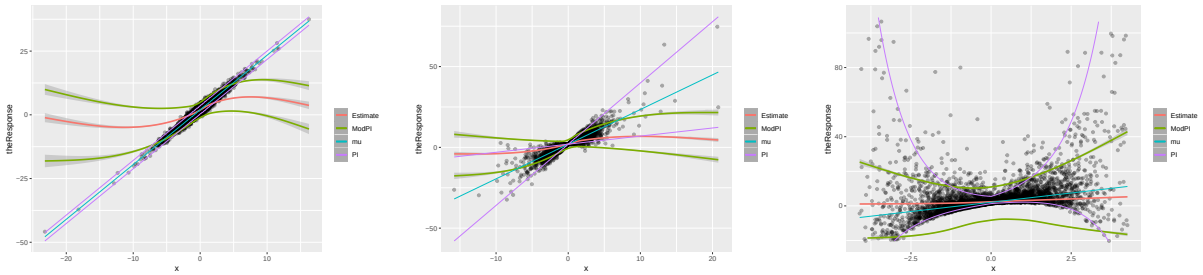

Figure 2: PI for polynomial regression data, left: homoskedastic noise, middle: heteroskedastic noise, right: asymmetric noise, plotted 95% of the data.

the R Package `mcgv` [Wood, 2019]. These are a shrinkage version of a cubic spline basis defined by a modest sized set of knots spread evenly through the covariate values and are penalized by the conventional intergrated square second derivative cubic spline penalty [Wood, 2017]. In this subchapter, we use $n = 5000$ for both the training and test data sets, respectively. Using 20 runs with different training data and a single test set we can observe that the prediction intervals deliver average coverage. Throughout all simulations, we find that the pi-QF may not cover the full complexity of the data as we cannot find the correct signal in the tails (e.g. Figure 2). This result is not too surprising since quantile/random forests are known having difficulties delivering stable estimations in the tails of the data [Hothorn and Zeileis, 2017]. However, in Section 5.3 we find that pi-QF has strong improvements in stability of the prediction intervals compared to a quantile forest.

Additive Regression Data. In Figure 1 the results for homoskedastic, heteroskedastic and asymmetric noise are shown. Despite the slight problems in pulling up the noise in the tails we see that for the homoskedastic case the average prediction interval is quite narrow and also resembles the real band for most of the data. The same holds for the heteroscedastic case. In the case of asymmetric noise, the tails are not covered and in the center the upper bound seems reasonable whereas the lower bound underestimates. The interesting question is if the competitors can outperform or not (see Section 5.3).

Polynomial Regression Data. The results of the data can be found in Figure 2. It is noticeable that ip-RF does not recognize any signal at the tails of the homoscedastic data. Otherwise, the signal is covered satisfactorily in this case. For the heteroscedastic noise the algorithm already recognizes much more at the edge, whereby again the full complexity of the data is probably not detected completely. The change in variance in the middle of the plot, however, is sufficiently well estimated. In the case of asymmetric data, strong parallels can be drawn to the previous data model. Also here the major proportion of the set seems well covered, but the tails of the data are almost not recognized.

Nonlinear Function After Selection. Figure 3 indicates that the pi-QF performs on the hardest problem reasonably well, as we think due to sparsity and non-linearity, in all of the three settings. The main structure except for the very tails is mapped and for all three plots we see the predictive intervals to be reasonalby near the true bands. What is more, the estimate (more precisely the estimated median of

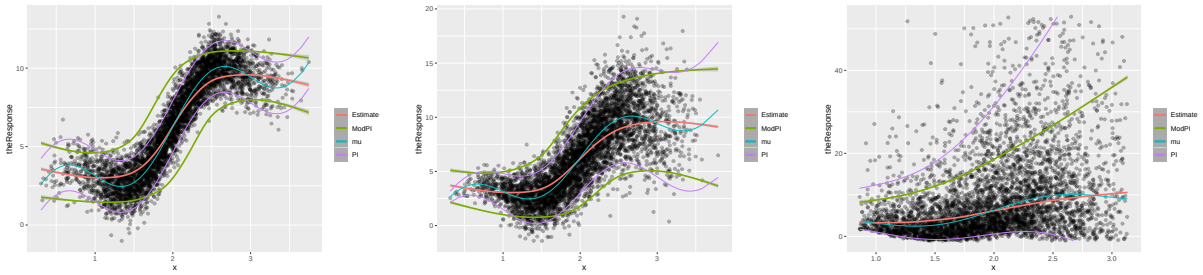

Figure 3: PI for nonlinear function after selection data, left: homoskedastic noise, middle: heteroskedastic noise, right: asymmetric noise, plotted 95% of the data.

	Coverage		Width		MSE	
Algorithm	Mean	SD	Mean	SD	Mean	SD
pi-QF	0.907	0.019	6.736	0.529	15394	2375
pi-sQF	0.897	0.022	6.441	0.555	14146	2249
confRF	0.891	0.019	5.846	0.390	10614	1337
QRF	0.981	0.008	9.610	0.130	28277	681
true	0.902	0.010	3.290	0.000		

Table 1: Results for additive regression data with homoskedastic noise, $n = 1000$, $p = 10$, 20 repetitions each.

pi-QF) follows the true mean very narrowly. This behavior is what we somehow expected for a medium sized data set with $n = 5000$ for both, training and test size.

5.3 Discussion

We use the following settings to compare our proposed algorithm pi-QF to:

1. A conformal quantile forest using a single stretch parameter (s-QF), see Remark 3.1.
2. Conformal random forest (confRF) using Algorithm 1 and set $\mathcal{A}$ to be a random forest.
3. Quantile regression forests estimating the 0.1 and 0.9 quantiles (QRF).

The comparisons can be found in figures 4 to 12 and tables 1 to 9, divided according to data generating process and noise term. Each graphic consists of three plots, one of which stands for a comparison with a competitor. As default setting we use $p = 10$, $n = 1000$ for tables and graphics. Furthermore, for the tables we have the results from 20 learning sets evaluated on one test set whereas for the plots we evaluate one training set on one test set. In case of asymmetric noise we additionally performed a large-scale simulation with $n = 20000$ for each data generating process.

	Coverage		Width		MSE	
Algorithm	Mean	SD	Mean	SD	Mean	SD
pi-QF	0.906	0.018	8.242	0.772	19218	2927
pi-sQF	0.903	0.022	7.997	0.733	17984	2086
confRF	0.897	0.022	7.879	0.731	18520	2204
QRF	0.966	0.008	10.798	0.225	28803	1019
true	0.896	0.010	5.229	0.100		

Table 2: Results for additive regression data with heteroskedastic noise, $n = 1000$, $p = 10$, 20 repetitions each.

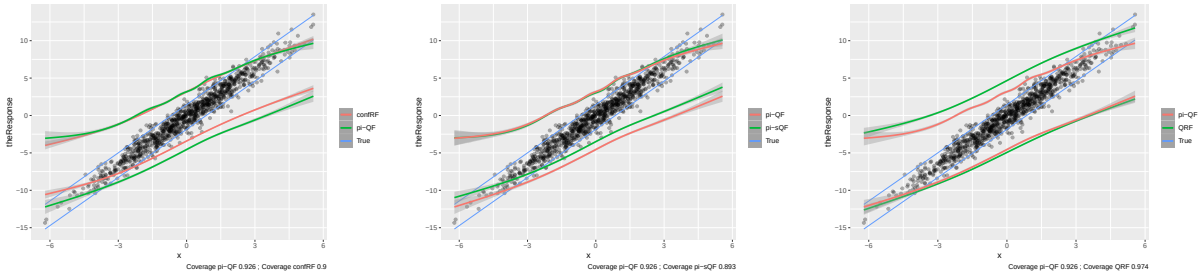

Figure 4: PI for additive regression data with homoskedastic noise.

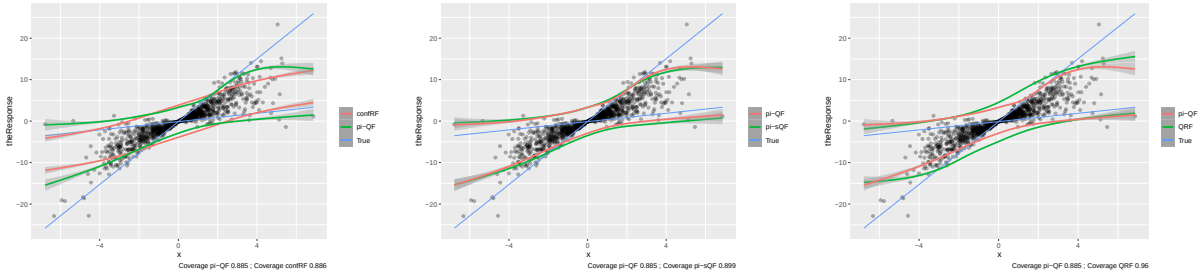

Figure 5: PI for additive regression data with heteroskedastic noise.

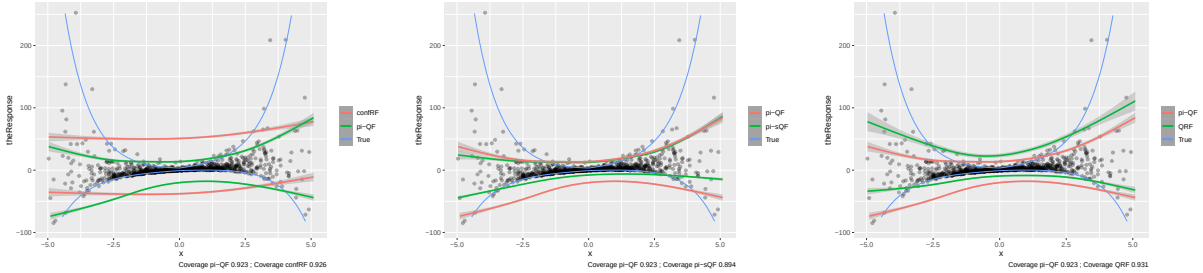

Figure 6: PI for additive regression data with asymmetric noise.

		Coverage		Width		MSE	
n	Algorithm	Mean	SD	Mean	SD	Mean	SD
1000	pi-QF	0.885	0.021	45.990	9.705	5.806e+07	6.496e+07
	pi-sQF	0.898	0.016	42.840	6.737	5.443e+07	6.416e+07
	confRF	0.900	0.022	72.150	21.470	5.737e+07	6.698e+07
	QRF	0.918	0.012	69.370	16.690	1.227e+08	2.060e+08
	True	0.899	0.009	59.880	9.512		
20000	pi-QF	0.872	0.005	35.37	1.748	9.765e+08	5.975e+08
	pi-sQF	0.9	0.006	36.03	2.185	9.315e+08	5.621e+08
	confRF	0.900	0.006	57.520	2.987	1.059e+09	6.024e+08
	QRF	0.953	0.002	66.140	2.784	9.961e+08	5.103e+08
	True	0.9	0.002	59.690	2.084		

Table 3: Results for additive regression data with asymmetric noise, $p = 10$, 20 repetitions each.

	Coverage		Width		MSE	
Algorithm	Mean	SD	Mean	SD	Mean	SD
pi-QF	0.906	0.029	8.873	1.067	51917	9188
pi-sQF	0.905	0.026	8.789	0.989	51089	10489
confRF	0.907	0.021	9.101	0.982	44262	6727
QRF	0.960	0.012	10.783	0.286	65500	4755
True	0.901	0.010	3.290	0.000		

Table 4: Results for polynomial regression data with homoskedastic noise, $n = 1000$, $p = 10$, 20 repetitions each.

Additive Regression Data. The results for the fist case, additive regression data with homoskedastic errors, are shown in figure 4 and table 1. There you can see that the results are almost the same for most of the competitors. It seems that the QRF strongly overpredicts the intervals in both coverage and width. Apart from these differences differences, there are not many advantages or disadvantages, so that in this case all other prediction intervals are of the same quality. It is noticeable that the width of the intervals on average deviates greatly from the true values. This can be explained by the fact that the signal at in the tails is not recognized very well, but in the rest you can hardly see a difference between Predicted Intervals and True intervals on the plots.

For heteroskedastic noise (figure 5 and table 2) we cannot find any significant differences between pi-QF and pi-sQF. Which means that in this case it does not matter if we use one or two stretching parameters in the conformal step of the proposed algorithm. Compared to confRF we see that the former can allocate more noise in the tails, but from the information in the table there is no further difference. As far as the QRF is concerned, for the first time we see that the predicted intervals are unstable, meaning that they are significantly overestimated and the compactness of the data in the range around $(0, 0)$ is hardly recognized. In contrast to the data with homoscedastic signal, the predicted intervals of all algorithms except QRF are much closer to the true intervals.

In case of asymmetric noise (figure 6), table 3 presents results for two simulations. In the upper part a run with $n = 1000$ and in the lower part a run with $n = 20000$. We find that all predicted intervals have sufficient coverage, however the QRF overestimates the bands again. The values for width and MSE in table 3 also show that the predictions from QRF and confRF are highly unstable for $n = 1000$ and seem to have stabilized for $n = 20000$, but only the confRF bands have become significantly tighter. Furthermore, the predictions for pi-QF and pi-sQF are very similar. They are on average much narrower than the true intervals, which is because they cannot grow exponentially in the asymptotic range, as the error terms do. For the mode of the distribution we find both algorithms to deliver accurate values.

Polnomial Regression Data. For polynomial regression data with homoskedastic errors we have similar results to the case for additive regression data: All algorithms except QRF show average coverage

	Coverage		Width		MSE	
Algorithm	Mean	SD	Mean	SD	Mean	SD
pi-QF	0.902	0.025	9.500	1.088	53536	8204
pi-sQF	0.895	0.023	8.984	1.003	50797	8837
confRF	0.908	0.017	10.447	0.966	56214	6225
QRF	0.947	0.010	11.302	0.444	64120	4801
True	0.902	0.010	4.598	0.117		

Table 5: Results for polynomial regression data with heteroskedastic noise, $n = 1000$, $p = 10$, 20 repetitions each.

		Coverage		Width		MSE	
n	Algorithm	Mean	SD	Mean	SD	Mean	SD
1000	pi-QF	0.888	0.031	345.3	576.2	3.428e+18	1.368e+19
	pi-sQF	0.901	0.023	646.6	1486	3.428e+18	1.368e+19
	confRF	0.904	0.024	3555	8236	3.522e+18	1.367e+19
	QRF	0.912	0.012	1.341e+06	4.243e+06	1.165e+19	2.801e+19
	True	0.897	0.009	7.86e+05	2.128e+06		
20000	pi-QF	0.873	0.006	1.413e+11	4.434e+11	1.671e+42	7.399e+42
	pi-sQF	0.9	0.005	4.269e+07	1.888e+08	1.671e+42	7.399e+42
	confRF	0.899	0.005	1092	715.5	1.671e+42	7.399e+42
	QRF	0.938	0.003	2.502e+11	6.785e+11	1.671e+42	7.399e+42
	True	0.9	0.002	1.888e+16	7.696e+16		

Table 6: Results for polynomial regression data with asymmetric noise, $p = 10$, 20 repetitions each.

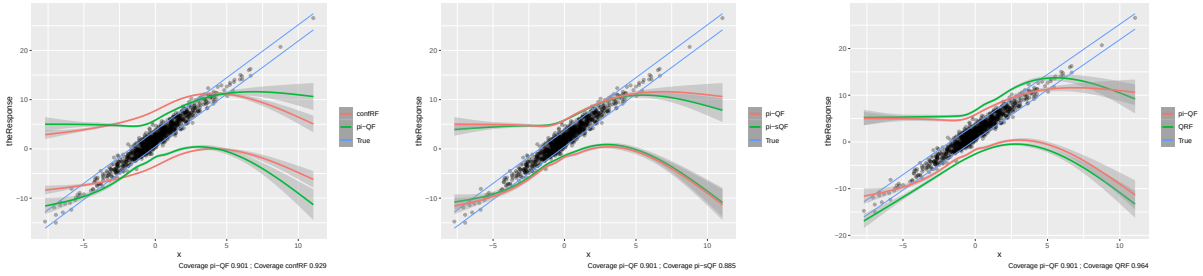

Figure 7: PI for polynomial regression data with homoskedastic noise.

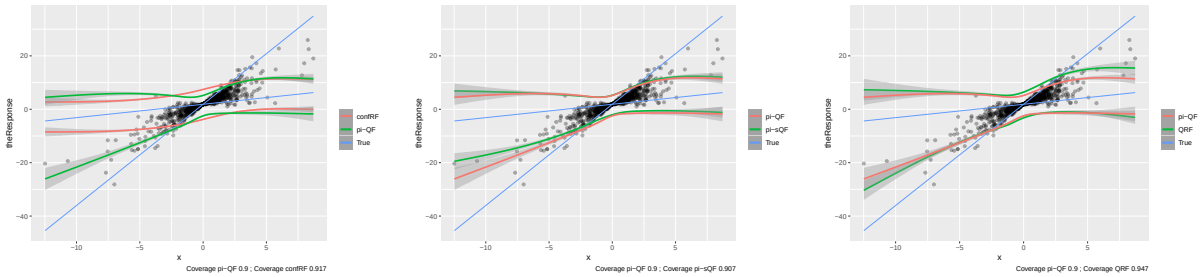

Figure 8: PI for polynomial regression data with heteroskedastic noise.

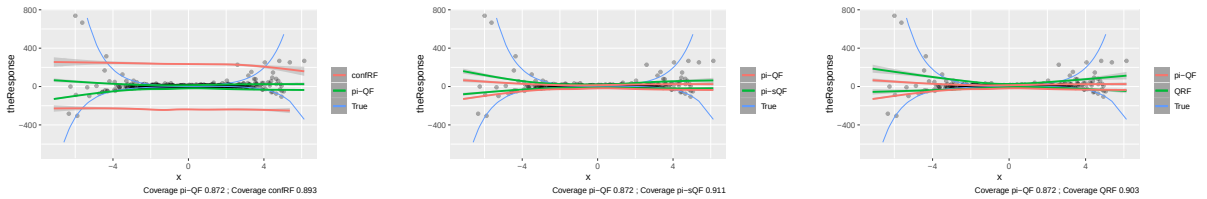

Figure 9: PI for polynomial regression data with asymmetric noise.

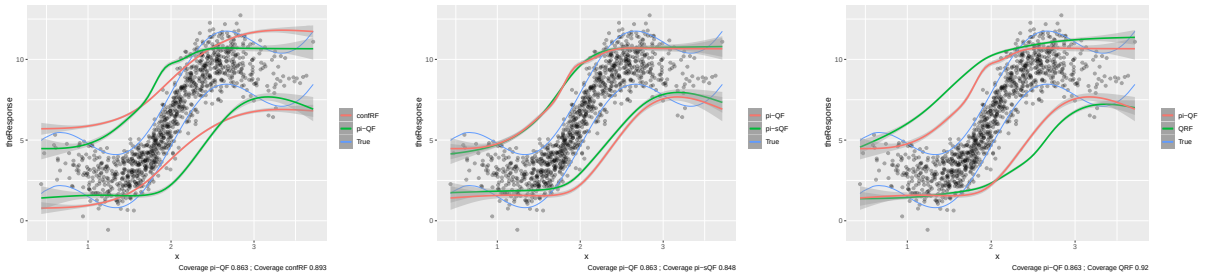

Figure 10: PI for nonlinear function after selection data with homoskedastic noise.

of level α. The QRF strongly overestimates the bands, which is also reflected in the fact that they are the furthest apart. However, the prediction intervals seem to be quite stable here, because the corresponding SD is smaller than the others at level 2/3. From table 4, there is hardly any difference between pi-QF, pi-sQF and confRF. You can see, however, that the latter shows the best values for the MSE, so we can assume that the random forest's average predictions outperform those of the QRFs. With figure 7 we can see that almost all algorithms can estimate prediction intervals near the mode of the distribution quite well, but hardly recognize any signal at the edges. This can be seen on the one hand by the high variances of the bands at the edges, and on the other hand by the fact that the true intervals run diagonally through the plot whereas the predictions are diametrically opposed to the true values in the tails.

In the case of heteroscedastic errors (table 4), we see that, perhaps due to symmetry, the pi-sQF provides the best results. With regard to coverage, the same results as before can be found, with pi-sQF having the lowest value of .895 and confRF having the highest value of .908 of the three models. However, we consider this all close enough to level α, in complete contrast to the QRF, which is at .947 and also shows the longest intervals. Furthermore, it's easy to see in figure 5 that all models have problems to recognize the signal or to display the full complexity of the data. Furthermore, similar to the homoscedastic case, the confRF does not map the vanishing variance of the data close to the mode of distribution, but this is due to the definition of the algorithm. In the plot pi-QF and pi-sQF are nearly identical, only the QRF has the wide prediction intervals in the range $x, y > 0$.

The last case now shows very interesting results in table 6 and figure 9. If you first look at the results for the case $n = 1000$, we can already identify acceptably stable results with regard to coverage. Mean squared errors are incredibly high in all cases, but on the same level as the corresponding SD. For the values in width we find that the pi-QF predict the narrowest intervals, i.e. the asymmetry, best, whereas the pi-sQF predict almost twice as wide bands. Additionally, the SD is larger by a factor 3. In contrast, ConfRF doesn't seem to work very well (the intervals are a factor 10 wider than those of pi-QF) and the QRF shows extremely wide intervals. The asymmetries in the edges therefore seem to produce extremely wide bands that are not tightened near the mode of the empirical distribution. This is attenuated by estimating the IQR with a subsequent conformal step. If we now discuss the lower part of table 6, it becomes clear that a higher number of data ($n = 20000$), which is used both for training and test data, leads to completely opposite results. It seems that the distribution will be significantly more difficult to capture by a given model, and, except for the confRF, all algorithms need very long intervals to achieve average coverage. Hence, one can assume that much more information lies in the asymmetric tails and therefore rises to the widths really strongly. Only the confRF now shows very stable and narrow prediction bands. Based on theory, we did not expect these results and they could be explained by the fact that an estimate of the QRF for IQR in this case also quickly becomes unstable.

	Coverage		Width		MSE	
Algorithm	Mean	SD	Mean	SD	Mean	SD
pi-QF	0.906	0.029	8.873	1.067	51917	9188
pi-sQF	0.905	0.026	8.789	0.989	51089	10489
confRF	0.907	0.021	9.101	0.982	44262	6727
QRF	0.960	0.012	10.783	0.286	65500	4755
True	0.901	0.010	3.290	0.000		

Table 7: Results for nonlinear function after selection data with homoskedastic noise, $n = 1000$, $p = 10$, 20 repetitions each.

	Coverage		Width		MSE	
Algorithm	Mean	SD	Mean	SD	Mean	SD
pi-QF	0.902	0.025	9.500	1.088	53536	8204
pi-sQF	0.895	0.023	8.984	1.003	50797	8837
confRF	0.908	0.017	10.447	0.966	56214	6225
QRF	0.947	0.010	11.302	0.444	64120	4801
True	0.902	0.010	4.598	0.117		

Table 8: Results for nonlinear function after selection data with heteroskedastic noise, $n = 1000$, $p = 10$, 20 repetitions each.

		Coverage		Width		MSE	
n	Algorithm	Mean	SD	Mean	SD	Mean	SD
1000	pi-QF	0.855	0.021	31.203	3.129	593242	111837
	pi-sQF	0.905	0.025	33.331	5.361	352997	97677
	confRF	0.904	0.022	40.676	5.887	490378	66920
	QRF	0.899	0.014	39.574	2.226	401061	360728
	True	0.902	0.012	36.112	0.639		
20000	pi-QF	0.848	0.004	29.930	0.650	10772171	514074
	pi-sQF	0.899	0.005	29.767	1.082	4721560	360720
	confRF	0.901	0.006	38.803	1.327	8552330	131202
	QRF	0.906	0.003	38.483	0.477	3223824	422639
	True	0.900	0.003	36.307	0.143		

Table 9: Results for nonlinear function after selection data with asymmetric noise, $p = 10$, 20 repetitions each.

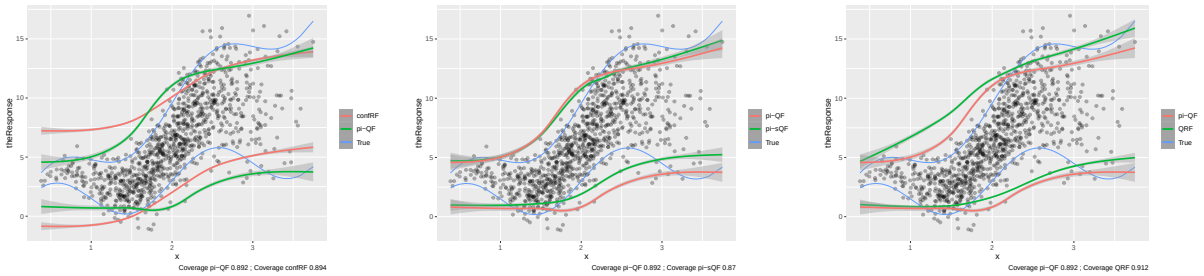

Figure 11: PI for nonlinear function after selection data with heteroskedastic noise.

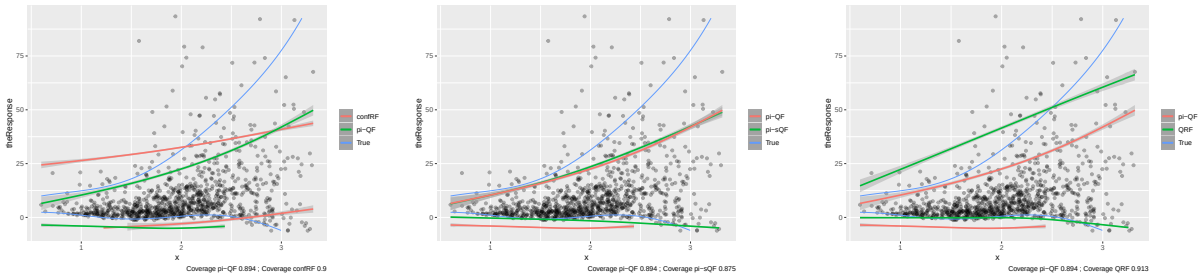

Figure 12: PI for nonlinear function after selection data with asymmetric noise.

Nonlinear Function after Selection. Previously in section 5.2 we found that for $n = 5000$ the pi-QF shows good results in each setting. By looking at figures 10 to 12 we see that $n = 1000$ is a much more difficult case for the goal of modeling the noise correctly. The sparsity of the data definitively becomes an issue with so few observations, even in the case of 10 predictios where only 4 are non-zero. Figure 10 and table 7 show the results for the homoskedastic case. We can see that, compared to the left plot in figure 3, the pi-QF delivers slightly more jagged prediction intervals that tend to be wider and map the details of the data less accurately. Analyzing the competitors we find that the QRF stronly overestimates the coverage and interval width, whereas the other three algorithms provide similarly good results and the prediction intervals from pi-QF and pi-sQF are nearly identical.

For the heteroskedastic case we see alomst the same results in table 8 compared to table 7, but in figure 11 the results are not so clear. The pi-QF manages to find most of the singal in the data, but may not cover the full complexity. When looking at the other models, piQF shows the most accurate results. It is noticable that pi-sQF has troubles in the lower bound and the two remaining models are not pulling up the information thoroughly. We make this assumption because QRF strongly overpredicts and confRF predicts diagonal lines through the data.

Adding asymmetric noise to the data makes estimating the models a greater challenge. The results from section 5.2 with $n = 5000$ can also be found in this context. We can see in figure 12 that the reduction of observations to n=1000 increases the sparsity of the data. Table 9 shows some astonishing findings. First, for the upper part the table, it seems pi-QF has problems converging against α and remains with $\alpha = .855$ slightly below for the average of 20 predictions, whereas all competitors reach the alpha level. For the estimated width of the intervals, we find the pi-QF bands as the narrowest and even smaller than the true bands on average, because in the asymptotic range they do not rise exponentially and as do the true intervals. The width for pi-sQF predictions is quite similar to the just discussed and still smaller than the true values, in contrast to this the bands of QRF and confRF exceed the width of the true bands. Graphic 12 explains this behavior, because confRF and QRF predict almost linear interval limits that overestimate on one side and underestimate on the other side of the data. In contrast, pi-QF and pi-sQF follow the exponentially increasing variance of the data from left to right. Since they are very similar, it's no wonder that they have quite similar statistics in table 9 show. But in this table we also see that remarkably the MSE are significantly different, whereas due to the definition of the models, a certain proximity between pi-QF and pi-sQF could have been expected for the prediction of the medians. This is absolutely not the what happened, also for the QRF. Thus, the latter shows an abstruse high SD for the MAD statistics. These results also hold for the case with $n = 20000$ in the lower part of table 9. Again confRF has the smallest deviation in mean prediction and the other values are three to four times the SD. However, the confRF also has a very high value for the mean of the MSE. This is even surpassed by the median predictions of the pi-QF. Strangely enough, the predictions of QRF and pi-SQF are much lower. This leads us to assume that there are extreme inconsistencies within a quantile regression forest where the splits are set. This has not yet become very important, but it becomes clear for these data. Last but not least, when looking at the coverage ratios, it is noticeable that pi-QF remains below the alpha level again, although all other models reach it. This time also the interval width cannot be identified as a cause, because the pi-sQF at alpha level has similar intervals on average. one can therefore only assume that the pi-QF is too restrictive for outliers and that the pi-sQF eliminates this behaviour in this case. Another possibility, however, lies in the inconsistencies of the qrf estimation. It is quite possible that we could not compensate these effects with only 20 repetitions.

Model	Coverage	Width	MAD
pi-QF	0.898	173	28.2
pi-sQF	0.910	186	27.4
confRF	0.910	213	38.1
QRF	0.970	258	26.7
BART	0.897	107	22.6

Table 10: Bike sharing data results, one repetition.

	Coverage		Width	
Algorithm	Mean	SD	Mean	SD
pi-QF	0.911	0.007	181	4.920
pi-sQF	0.910	0.007	182	5.811
confRF	0.898	0.006	204	6.200
QRF	0.968	0.001	253	0.892

Table 11: Bike sharing data results, 20 repetitions.

6 Real data applications

Additionally to synthetic data, we evaluate the algorithm on three real world data sets. With BART, Bayesian Additive Regression Trees, (see Section 7.4) we add another competitor to the list. In real data applications, we cannot trust our oracle anymore and therefore want to utilize another promising model-based approach yielding predictive intervals.

Bike Sharing Data. The first data contains the hourly and daily count of rental bikes between years 2011 and 2012 in Capital bikeshare system (Washington DC, USA) with the corresponding weather and seasonal information[1]. We have 12 covariates to predict standardized count of total rental bikes including both casual and registered. 6 of the features are factor variables and the remaining 6 are numeric. The data contains 17379 observations and we split them s.t. $|D_1| = 11500$, $|D_2| = 2300$ and $|D_3| = 3579$. The results are shown in tables 10 and 11. On this data we see that using one run all predicted intervals are nearly of the same coverage, only the QRF strongly overestimates the intervals which can be seen in both coverage and width. In terms of width and MAD, BART delivers the best results, i.e. the tightest intervals. This seems to be a scenario where BART work outstandingly well. The pi-QF produces the smallest intervals in comparison to s-QF and confRF. Summing up, we see that the proposed method works better than most competitors but cannot outperform BART.

House Prices. This data contains sale prices for homes in King County, Washington, between May 2014 and May 2015, including $21,613$ observations which we split s.t. $|D_1| = 15,000$, $|D_2| = 3,000$ and $|D_3| = 3,613$. There are 19 covariates used to predict log sale prices[2] and all features are standardized to have mean 0 and variance 1. The results are shown in tables 12 and 13 where we can see that the BART delivers the tightest intervals but shows the weakest coverage ratio of nealy 0.7. All other models deliver predictive intervals of the same quality, except the QRF that overestimates the intervals as in the bike sharing data set. pi-QF delivers the most stable results with respect to mean and sd. pi-sQF is nealy as good as the former, whereas confRF shows first signs of instable prediction intervals.

Traffic Data. The third data set contains hourly Minneapolis-St Paul, MN traffic volume for west-bound I-94. Includes weather and holiday features from 2012-2018[3]. There are 6 features containing 2 factor variables and 4 numeric ones to predict the normalized traffic volume. The set has 48204 observations which we split s.t. $|D_1| = 27500$, $|D_2| = 5500$ and $|D_3| = 15204$. In tables 14 and 15 we

[1] Source: https://archive.ics.uci.edu/ml/datasets/bike+sharing+dataset
[2] Source: https://www.kaggle.com/gabriellima/house-sales-in-king-county-usa/data
[3] Source: https://archive.ics.uci.edu/ml/datasets/Metro+Interstate+Traffic+Volume

Model	Coverage	Width	MAD
pi-QF	0.890	0.589	0.128
s-QF	0.887	0.576	0.129
confRF	0.880	0.538	0.134
QRF	0.941	0.688	0.126
BART	0.689	0.304	0.129

Table 12: Housing data results, one repetition.

	Coverage		Width	
Algorithm	Mean	SD	Mean	SD
pi-QF	0.905	0.005	0.593	0.014
pi-sQF	0.904	0.005	0.586	0.013
confRF	0.899	0.006	0.565	0.014
QRF	0.955	0.002	0.688	0.001

Table 13: Housing data results, 20 repetitions.

see the prediction results. The first major take-away is the bad performance of BART. It seems that the algorithm has severe problems in picking up the information from categorical variables with many categories. We tried different ways of defining these but all attempts ended up with a coverage near the reported result (which is the best attempt).

7 Further Literature

7.1 Conformalized Quantile Regression

Romano et al. [2019] discuss a split conformal procedure for quantile regression using an additive parameter yielding predictive intervals of level α. The procedure can be seen as a correction parameter because the authors propose to estimate the α-level intervals using quantile regression and then add a constant that is retrieved from a split-conformal procedure to provide coverage. In addition, the conformalized quantile regression is adapted towards quantile regression forests and neurnal networks. In our view, this method uses the same split-conformal intervals as Lei et al. [2017] and the same adjustment trying to account for heteroskedasticity. The underlying model for the QRF is knwon to be unstable and by adding a conformal constant, the authors do not fix any of the instabilities. What is more, there is no proper discussion on how the extention towards a quantile regression forest or a neural network provides coverage.

7.2 Asymptotic Normality of Random Forests

Literature has recently shown that under strict assumptions the asymptotic normality of random forests applies and that confidence intervals can be calculated. All methods are based on a rather strict partial scanning procedure, which may not be applicable in practice because the complexity of the divisions becomes too high. Mentch and Hooker [2014a] are able to show that predicting by subsampling the training data set and averaging over the trees built on these subsets the resulting estimator takes the form of a U-statistic. Therefore, the predictions for individual feature vectors are asymptotically normal so that confidence intervals can follow the predictions. Extending these findings, Mentch and Hooker [2014b] find a way to define a grid structure such that hypothesis test for both variable importance and underlying additive model structure can be performed. Another approach on asymptotic normality of random forests is the work of Wager et al. [2013], Efron [2014]. The authors found variance estimates based on the jackknife and the infinitesimal jackknife, where the latter requires less bootstrap estimates than the former to achieve a given accuracy. The asymptotic variance can consistently be estimated using an infinitesimal jackknife for bagged ensembles. Furthermore, Wager [2014] analyze a random

Model	Coverage	Width	MAD
pi-QF	0.880	2.92	1.09
s-QF	0.894	2.90	1.11
confRF	0.901	2.94	1.19
QRF	0.903	2.88	1.12
BART	0.091	0.31	1.21

Table 14: Traffic data results, one repetition.

	Coverage		Width	
Algorithm	Mean	SD	Mean	SD
pi-QF	0.883	0.012	3.00	0.044
pi-sQF	0.899	0.004	2.98	0.047
confRF	0.902	0.003	2.95	0.019
QRF	0.906	0.004	2.88	0.014

Table 15: Traffic data results, 20 repetitions.

forest model based on subsampling. For a scaled subsampling size the predictions are asymptotically normal and one can characterize and estimate the error distribution of random forest predictions. This paper proves a weaker condition for the subsampling size than the results shown in Mentch and Hooker [2014a,b].

7.3 Transformation Forests

Transformation Forests(TF) as introduced by Hothorn and Zeileis [2017] are built based on a parametric family of distributions that are characterized by their transformation function. These transformation functions are obtained in a transformation tree that is able to find distributional changes in the data. The resulting forests are shown to be adaptive local likelihood estimators of conditional distribution functions, are fully parametric and hence allow for inference procedures. The basis of the algorithm are conditional inference trees and model-based recursive partitioning keeping the unbiased variable selection property from the ancestors. The idea here is to find a improvement to QRT. Generally speaking, the authors propose a model for the estimation and prediction of conditional distributions for Y given predictor variables x in situations where moments beyond the mean are important.

7.4 Bayesian Additive Regression Trees

Introduced by Chipman et al. [2008] as a sum-of-trees model where each tree is constrained by a regularization prior to be a weak learner, and fitting and inference are accomplished via an iterative Bayesian backfitting MCMC algorithm that generates samples from a posterior. Hence, this BART is a nonparametric Bayesian regression approach and enables full posterior inference including point and interval estimates of the unknown regression function as well as the marginal effects of potential predictors. The BART algorithm was modified to faster posterior estimation by He et al. [2018] in their work 'XBART: Accelerated Bayesian Additive Regression Trees'.

8 Conclusion

Throughout all sections we found evidence that the proposed method is vaild. What is more, our approach allows us to define two methods. We can either use single or two different stretching parameters for the IQR in the conformal step. Both approaches are valid in the theoretical setting and showed empirical proof for coverage of level α, and hence the method controls the miscoverage ratio in finite samples. We are able to proof the result and see that the algorithm is capable of adapting the interval lengths to heteroskedasticity and asymmetry in the data. Furthermore, pi-QF and also pi-sQF outperform a

conformalized random forest as well as quantile regression forests estimating α-level predictive intervals directly.

For analyzing the real data results we add the BART to the competitors-list and identify its tendency of estimating too narrow bands, meaning that in two of the three application BART failes to reach α-coverage but delivers very tight prediction intervals. On two out of three datasets the QRF overestimates prediction intervals widely indicating that there are issues regarding the stability of these bands. Hence, we see that some of the critique to that method is reasonable and using a conformal procedure to overcome these problems is necessary. Comparing the results from pi-QF to BART, QRF or the true bands from artificial data, pi-QF provides reasonably tight prediciton intervals and average coverage at level α.

In addition to the findings of this thesis, some questions are still open in future research projects. First one can examine whether there is an ideal procedure to determine ρ and if so, how this ρ is determined. Another question is that of stretching parameters; we have seen some settings where using two stretching parameters is clearly more advantageous than using just one. In general, we are able stabilize the noisy predictions from a quantile regression forest using a split-conformal procedure.

Bibliography

Yali Amit and Donald Geman. Shape quantization and recognition with randomized trees. *Neural computation*, 9(7):1545–1588, 1997.

Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

Hugh A. Chipman, Edward I. George, and Robert E. McCulloch. Bart: Bayesian additive regression trees. 2008. doi: 10.1214/09-AOAS285.

Bradley Efron. Estimation and accuracy after model selection. *Journal of the American Statistical Association*, 109(507):991–1007, 2014.

Tibshirani Hastie, R Tibshirani, and J Friedman. The elements of statistical learning new york. *NY: Springer*, 2009.

Jingyu He, Saar Yalov, and P. Richard Hahn. Xbart: Accelerated bayesian additive regression trees. 2018.

Torsten Hothorn and Achim Zeileis. Transformation forests. 2017.

Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. *An Introduction To Statistical Learning*, volume 112. Springer, 2013.

Danijel Kivaranovic, Kory D. Johnson, and Hannes Leeb. Adaptive, distribution-free prediction intervals for deep neural networks. 2019.

Roger Koenker and Kevin F Hallock. Quantile regression. *Journal of economic perspectives*, 15(4): 143–156, 2001.

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

Jing Lei, James Robins, and Larry Wasserman. Distribution-free prediction sets. *Journal of the American Statistical Association*, 108(501):278–287, 2013.

Jing Lei, Alessandro Rinaldo, and Larry Wasserman. A conformal prediction approach to explore functional data. *Annals of Mathematics and Artificial Intelligence*, 74(1-2):29–43, 2015.

Jing Lei, Max G'Sell, Alessandro Rinaldo, Ryan J. Tibshirani, and Larry Wasserman. Distribution-free predictive inference for regression. 2017.

Yi Lin and Yongho Jeon. Random forests and adaptive nearest neighbors. *Journal of the American Statistical Association*, 101(474):578–590, 2006.

Nicolai Meinshausen. Quantile regression forests. *Journal of Machine Learning Research*, 7(Jun):983–999, 2006.

Lucas Mentch and Giles Hooker. Quantifying uncertainty in random forests via confidence intervals and hypothesis tests. 2014a.

Lucas Mentch and Giles Hooker. Formal hypothesis tests for additive structure in random forests. 2014b.

Harris Papadopoulos, Kostas Proedrou, Volodya Vovk, and Alex Gammerman. Inductive confidence machines for regression. In *European Conference on Machine Learning*, pages 345–356. Springer, 2002.

Yaniv Romano, Evan Patterson, and Emmanuel J. Candès. Conformalized quantile regression. 2019.

Glenn Shafer and Vladimir Vovk. A tutorial on conformal prediction. 2007.

Vladimir Vovk, Alex Gammerman, and Glenn Shafer. *Algorithmic learning in a random world*. Springer Science & Business Media, 2005.

Vladimir Vovk, Ilia Nouretdinov, Alex Gammerman, et al. On-line predictive linear regression. *The Annals of Statistics*, 37(3):1566–1590, 2009.

Stefan Wager. Asymptotic theory for random forests. 2014.

Stefan Wager, Trevor Hastie, and Bradley Efron. Confidence intervals for random forests: The jackknife and the infinitesimal jackknife. 2013.

Hadley Wickham. *Package 'tidyverse'*, 2017. URL https://CRAN.R-project.org/package=tidyverse. R package version 1.2.1.

Simon Wood. *Mixed GAM Computation Vehicle with Automatic Smoothness Estimation'*, 2019. URL https://CRAN.R-project.org/package=mgcv. R package version 1.8-28.

S.N Wood. *Generalized Additive Models: An Introduction with R*. Chapman and Hall/CRC, 2 edition, 2017.

A Summary

The thesis proposes a method that yields distribution-free prediction intervals for Quantile Regression Forests (pi-QF) that overcomes the deficiencies of these forests. To do so, we apply a split-conformal procedure to calibrate QRF estimation of the Inter Quartile Range (IQR). The proposed method stretches the upper and lower bounds separately. We prove the result and apply the algorithm on different artificial and real data sets. We thus provide empirical proof that our method improves the results from a standard QRF in terms of width, coverage and stability of the predictive intervals. What is more, the approach allows us to define two methods. We can either use single or two different stretching parameters for the IQR in the conformal step. Both approaches are valid in the theoretical setting and showed empirical proof for coverage of level α, and hence the method controls the miscoverage ratio in finite samples. We are show that the algorithm is capable of adapting the interval lengths to heteroskedasticity and asymmetry in the data. Furthermore, pi-QF and also pi-sQF outperform a conformalized random forest as well as quantile regression forests estimating α-level predictive intervals directly.

B Zusammenfassung der Arbeit

In dieser Arbeit wird eine Mehtode vorgeschlagen, die verteilungsfreie Vorhersageintervalle für Quantile Regression Forests (pi-QF) liefert und die Defizite der Quantile Regression Forests überwindet. Dazu wenden wir ein split-conformal Verfahren an, um die QRF-Schätzung des Inter-Quartils-Bereichs (IQR) zu kalibrieren. Die vorgeschlagene Methode streckt die Ober- und Untergrenze getrennt voneinander. Wir beweisen das Ergebnis und wenden den Algorithmus auf verschiedene künstliche und reale Datensätze an. Damit liefern wir den empirischen Beweis, dass unsere Methode die Ergebnisse eines Standard-QRFs in Bezug auf Breite, Abdeckung und Stabilität der Vorhersageintervalle verbessert. Darüber hinaus erlaubt uns der Ansatz, zwei Methoden zu definieren. Wir können entweder einen einzelnen oder zwei verschiedene Dehnungsparameter für die IQR im conformal step verwenden. Beide Ansätze sind im theoretischen Rahmen gültig und haben empirische Beweise für die Abdeckung des Niveaus α gezeigt, und daher kontrolliert die Methode die Falschüberdeckungswahrscheinlichkeit in endlichen Stichproben. Des weiteren können wir zeigen, dass der Algorithmus in der Lage ist, die Intervalllängen an Heteroskedastizität und Asymmetrie der Daten anzupassen. Darüber hinaus übertreffen sowohl pi-QF als auch pi-sQF einen conformalized random forest sowie QRFs, die direkt Vorhersageintervalle auf α-Niveau schätzen.