Supplementary Materials: Composite Quantile Regression BART

Supplementary Materials

Proofs of Theoretical Results

Proof of Theorem 1: Posterior Consistency

Proof. We establish posterior consistency for the CQR-BART model under the stated regularity conditions. The proof proceeds in three main steps following the general framework for Bayesian nonparametric consistency (**?**).

Step 1: Prior Support. We first show that the BART prior places positive mass on Kullback-Leibler neighborhoods of the true quantile functions $f_{0,k}$. For any $\epsilon > 0$ and each quantile level τ_k, we need to show:

$$\Pi\left(\mathrm{KL}(p_{f_{0,k}}, p_{f_k}) < \epsilon\right) > 0 \tag{1}$$

where $\mathrm{KL}(p_{f_{0,k}}, p_{f_k})$ is the Kullback-Leibler divergence between the true and proposed models.

From the Lipschitz continuity condition (Condition 1) and the compact support condition (Condition 2), the true quantile functions $f_{0,k}$ can be approximated by piecewise constant functions on sufficiently fine partitions. The BART prior with $\alpha \in (0, 1)$ and $\beta > 0$ (Condition 3) allows trees of sufficient depth to create these partitions, following **?**. The growth condition on m (Condition 4) ensures that the sum of trees can approximate the true function with arbitrarily small error.

The asymmetric Laplace working likelihood satisfies the identifiability condition (Condition 5) through the pinball loss structure, ensuring that different quantile functions yield different distributions.

Step 2: Existence of Tests. We construct exponentially consistent tests for testing $H_0 : f_k = f_{0,k}$ against $H_1 : \|f_k - f_{0,k}\|_{L_2(P)} > \epsilon$. For the composite quantile regression setting, we adapt the testing arguments of **?** for Bayesian quantile regression. The check function ρ_τ induces a metric equivalent to the L_1 norm, and we can construct tests based on the empirical process:

$$\mathbb{G}_n(f_k) = \frac{1}{\sqrt{n}} \sum_{i=1}^{n} \left[\rho_{\tau_k}(Y_i - f_k(\mathbf{X}_i)) - \rho_{\tau_k}(Y_i - f_{0,k}(\mathbf{X}_i))\right] \tag{2}$$

The subgradient of the check function is bounded, enabling the application of maximal inequalities and the construction of exponentially powerful tests.

Step 3: Application of General Theory. Combining Steps 1 and 2 with Theorem 2.1 of **?**, we obtain the posterior consistency result:

$$\lim_{n\to\infty} \Pi\left(\|f_k - f_{0,k}\|_{L_2(P)} > \epsilon \mid \mathcal{D}_n\right) = 0 \quad \text{almost surely } P^\infty \tag{3}$$

for each quantile level τ_k, $k = 1, \ldots, K$.

The multi-quantile structure requires careful handling due to the shared data across quantile levels. However, the conditional independence in the composite likelihood and the factorization of the prior across quantile levels allow us to establish consistency for each quantile function separately. The boundedness of the check function's influence function ensures that the estimation at one quantile level does not unduly affect others. □

Proof of Theorem 2: Bounded Influence Function

Proof. We derive the influence function for the CQR-BART estimator and establish its boundedness. Let $\hat{\theta}_n$ be the posterior mean estimator based on n i.i.d. observations from distribution P. For a contamination point $(Y_c, \mathbf{X}_c)$ and contamination proportion ϵ, define the contaminated distribution:

$$P_\epsilon = (1-\epsilon)P + \epsilon\delta_{(Y_c,\mathbf{X}_c)} \tag{4}$$

where $\delta_{(Y_c,\mathbf{X}_c)}$ is the point mass at $(Y_c, \mathbf{X}_c)$. The influence function is defined as:

$$\Psi(Y_c, \mathbf{X}_c; \theta) = \lim_{\epsilon\to 0} \frac{\hat{\theta}_\epsilon - \hat{\theta}}{\epsilon} \tag{5}$$

where $\hat{\theta}_\epsilon$ is the estimator under P_ϵ.

For the CQR-BART estimator, the functional form of the influence function can be derived from the estimating equations of the posterior mode. The composite quantile regression estimating equations are:

$$\sum_{k=1}^{K}\sum_{i=1}^{n} \psi_{\tau_k}(Y_i - f_k(\mathbf{X}_i))\nabla f_k(\mathbf{X}_i) = 0 \tag{6}$$

where $\psi_\tau(u) = \tau - \mathbb{I}(u < 0)$ is the subgradient of the check function.

Under contamination, the estimating equations become:

$$(1-\epsilon)\sum_{k=1}^{K}\sum_{i=1}^{n} \psi_{\tau_k}(Y_i - f_k(\mathbf{X}_i))\nabla f_k(\mathbf{X}_i) + \epsilon\sum_{k=1}^{K} \psi_{\tau_k}(Y_c - f_k(\mathbf{X}_c))\nabla f_k(\mathbf{X}_c) = 0 \tag{7}$$

Differentiating with respect to ϵ and evaluating at $\epsilon = 0$ yields:

$$\Psi(Y_c, \mathbf{X}_c; \theta) = -H(\theta)^{-1}\sum_{k=1}^{K} \psi_{\tau_k}(Y_c - f_k(\mathbf{X}_c))\nabla f_k(\mathbf{X}_c) \tag{8}$$

where $H(\theta)$ is the Hessian matrix of the composite objective function.

The boundedness follows from three key observations:

1. **Bounded Check Function Influence**: The subgradient $\psi_\tau(u)$ is bounded by $\max(\tau, 1-\tau) \leq 1$ for all $u \in \mathbb{R}$.

2. **BART Prior Regularization**: The tree prior $\pi(\mathcal{T}_{kj}, \mathcal{M}_{kj})$ restricts the complexity of f_k and bounds the gradients ∇f_k. Specifically: - The leaf parameter prior $N(0, \sigma_\mu^2/m)$ prevents arbitrarily large parameter values - The tree structure prior with $\alpha \in (0,1)$ and $\beta > 0$ limits tree depth and complexity - The sum-of-trees representation with m fixed ensures bounded variation

3. **Positive Definite Hessian**: Under the identifiability conditions, $H(\theta)$ is positive definite with eigenvalues bounded away from zero, ensuring that $H(\theta)^{-1}$ is bounded.

Combining these observations, there exists a constant $M < \infty$ such that:

$$|\Psi(Y_c, \mathbf{X}_c; \theta)| \leq M \quad \text{for all } Y_c \in \mathbb{R}, \mathbf{X}_c \in \mathcal{X} \tag{9}$$

where M depends on the prior parameters and quantile levels but not on the contamination magnitude $|Y_c|$.

This establishes the robustness of CQR-BART to outliers and heavy-tailed contamination. □

Proof of Theorem 3: Computational Complexity

Proof. We analyze the computational complexity of the CQR-BART Gibbs sampling algorithm.

Time Complexity: The per-iteration cost decomposes as follows:

1. **Tree Updates**: For each of K quantile levels and m trees, updating a single tree requires: - Computing partial residuals: $\mathcal{O}(n)$ - Tree modification proposals (GROW/PRUNE/CHANGE/SWAP): $\mathcal{O}(n \log n)$ due to binary tree operations - Leaf parameter updates: $\mathcal{O}(L)$ where L is the number of leaves, typically $\mathcal{O}(\log n)$ Total per tree: $\mathcal{O}(n \log n)$

2. **Latent Variable Updates**: For each of n observations and K quantile levels, sampling ω_{ik} from the inverse Gaussian distribution: $\mathcal{O}(1)$ per update.

3. **Scale Parameter Updates**: For each of K quantile levels, computing the sufficient statistics: $\mathcal{O}(n)$

The total per-iteration complexity is:

$$T(n, p, K, m) = \mathcal{O}(Kmn \log n) + \mathcal{O}(nK) + \mathcal{O}(nK) = \mathcal{O}(Kmn \log n) \tag{10}$$

The $\mathcal{O}(p)$ factor for covariate dimension is absorbed in the tree operations, as splitting rules are evaluated for each available covariate.

Space Complexity: The memory requirements include: - Tree structures: $\mathcal{O}(KmL)$ where $L = \mathcal{O}(\log n)$ is the average number of leaves - Latent variables: $\mathcal{O}(nK)$ - Sufficient statistics: $\mathcal{O}(p)$ per tree for split statistics Total: $\mathcal{O}(nKmp)$

Mixing Time: The geometric ergodicity follows from: - The Gaussian proposals in Metropolis-Hastings steps for tree modifications - The conjugacy in

parameter updates (leaf parameters, scale parameters) - The data augmentation ensuring full conditional distributions are standard families

Following **?**, the spectral gap is bounded below by a constant depending on the prior hyperparameters, yielding the stated mixing time. □

Additional Simulation Results

Comprehensive Performance Metrics

Table 1: Integrated Mean Squared Error (IMSE) for conditional mean estimation across simulation scenarios. Standard errors in parentheses.

Method	Scenario 1	Scenario 2	Scenario 3	Scenario 4
CQR-BART (proposed)	0.124 (0.008)	**0.187** (0.012)	**0.231** (0.015)	**0.356** (0.022)
Single-quantile BART	0.138 (0.009)	0.219 (0.014)	0.268 (0.017)	0.421 (0.026)
Standard BART	**0.115** (0.007)	0.245 (0.016)	0.412 (0.026)	0.398 (0.025)
Frequentist CQR	0.312 (0.020)	0.335 (0.022)	0.391 (0.025)	1.045 (0.065)
Bayesian CQR Linear	0.289 (0.018)	0.324 (0.021)	0.378 (0.024)	0.892 (0.056)

Table 2: Continuous Ranked Probability Score (CRPS) for distribution estimation. Lower values indicate better performance. Standard errors in parentheses.

Method	Scenario 1	Scenario 2	Scenario 3	Scenario 4
CQR-BART (proposed)	**0.285** (0.015)	**0.342** (0.018)	**0.398** (0.021)	**0.523** (0.028)
Single-quantile BART	0.312 (0.016)	0.387 (0.020)	0.445 (0.023)	0.598 (0.031)
Standard BART	0.301 (0.016)	0.421 (0.022)	0.512 (0.027)	0.634 (0.033)
Frequentist CQR	0.412 (0.022)	0.456 (0.024)	0.523 (0.028)	1.124 (0.059)
Bayesian CQR Linear	0.398 (0.021)	0.432 (0.023)	0.501 (0.026)	0.987 (0.052)

Table 3: Average width of 95% predictive intervals. Standard errors in parentheses.

Method	Scenario 1	Scenario 2	Scenario 3	Scenario 4
CQR-BART (proposed)	3.89 (0.21)	4.23 (0.23)	5.12 (0.28)	4.87 (0.26)
Single-quantile BART	3.76 (0.20)	4.05 (0.22)	4.89 (0.26)	4.65 (0.25)
Standard BART	3.45 (0.18)	3.92 (0.21)	4.23 (0.23)	4.12 (0.22)
Frequentist CQR	4.23 (0.23)	4.56 (0.25)	5.45 (0.30)	5.78 (0.31)
Bayesian CQR Linear	4.12 (0.22)	4.41 (0.24)	5.23 (0.28)	5.45 (0.29)

Table 4: Performance under varying contamination proportions in Scenario 3. QSL values reported with standard errors.

Method	5% Contamination	10% Contamination	15% Contamination	20% C
CQR-BART (proposed)	0.587 (0.031)	0.612 (0.032)	0.645 (0.034)	0.
Single-quantile BART	0.645 (0.034)	0.698 (0.037)	0.756 (0.040)	0.
Standard BART	0.712 (0.038)	0.825 (0.043)	0.945 (0.050)	1.
Frequentist CQR	0.698 (0.037)	0.745 (0.039)	0.801 (0.042)	0.
Bayesian CQR Linear	0.678 (0.036)	0.731 (0.038)	0.789 (0.041)	0.

Robustness Analysis

MCMC Diagnostics and Convergence Analysis

We conducted comprehensive convergence diagnostics for all MCMC runs across simulation scenarios and real data applications.

Gelman-Rubin Diagnostics

For each key parameter class, we computed $\hat{R}$ statistics across multiple chains:

- Scale parameters: $\hat{R}(\sigma_k) = 1.02$ for $k = 1, \ldots, K$
- Leaf parameters: $\hat{R}(\mu_{kjl}) < 1.05$ for all trees and leaves
- Tree structure indicators: $\hat{R} < 1.10$ for splitting rules

All $\hat{R}$ statistics are below the recommended threshold of 1.1, indicating convergence.

Effective Sample Sizes

We monitored effective sample sizes (ESS) for key parameters:

- Scale parameters: $\text{ESS}(\sigma_k) > 1000$ for all quantile levels
- Leaf parameters: $\text{ESS}(\mu_{kjl}) > 500$ for terminal nodes
- Function evaluations: $\text{ESS}(f_k(\mathbf{X}_i)) > 800$ for test points

These ESS values indicate sufficient independent samples for reliable inference.

Trace Plots and Autocorrelation

Visual inspection of trace plots showed good mixing with no apparent trends or stuck chains. Autocorrelation functions decay rapidly, typically becoming negligible after 20-50 lags, indicating efficient exploration of the posterior distribution.

Predictive Checks

Posterior predictive checks showed good calibration, with observed data distributions well within the posterior predictive distributions. Quantile-quantile plots indicated appropriate coverage across the response distribution.

Software Implementation Details

The `cqrbart` package provides a comprehensive implementation of the proposed methodology with the following features:

Core Functionality

The main model fitting function:

```
cqrbart(y, x, tau = seq(0.1, 0.9, by = 0.1),
        n.trees = 50, n.iter = 10000, n.burn = 2000,
        n.chains = 1, n.threads = 1,
        alpha = 0.95, beta = 2.0,
        prior = list(nu = 3, lambda = NULL),
        keepevery = 1, verbose = TRUE)
```

Key Arguments

- `y`: Response vector of length n
- `x`: Covariate matrix of dimension $n \times p$
- `tau`: Vector of quantile levels (default: 0.1, 0.2, ..., 0.9)
- `n.trees`: Number of trees per quantile level (default: 50)
- `n.iter`: Total MCMC iterations (default: 10000)
- `n.burn`: Burn-in iterations (default: 2000)
- `n.chains`: Number of parallel chains (default: 1)
- `n.threads`: Number of threads for parallel computation across quantiles
- `alpha, beta`: Tree prior parameters
- `prior`: List containing inverse Gamma prior parameters for scale parameters

S3 Methods

The package provides standard S3 methods for analysis and visualization:

```
# Model fitting
fit <- cqrbart(y, x, tau = c(0.1, 0.5, 0.9))

# Prediction on new data
pred <- predict(fit, newx = x_test)

# Summary statistics
summary(fit)

# Visualization
plot(fit)  # Trace plots, variable importance, quantile processes

# Diagnostic plots
diagnostics(fit)  # Convergence diagnostics, residual analysis

# Variable importance
vi <- varimp(fit)
plot(vi)

# Conditional density estimation
cde <- conditional_density(fit, newx = x_test)
plot(cde)
```

Computational Optimizations

The implementation includes several optimizations for efficiency:

- **C++ Backend**: Core computational routines implemented in C++ via Rcpp
- **Parallelization**: Embarrassingly parallel computation across quantile levels
- **Memory Management**: Sparse representation of tree structures and incremental updates
- **Efficient Sampling**: Vectorized operations for latent variable and parameter updates
- **Convergence Monitoring**: Automated diagnostics with early stopping options

Reproducibility Features

- **Random Seed Control**: Deterministic reproducibility with explicit seed setting
- **Version Control**: Git repository with complete development history
- **Unit Testing**: Comprehensive test suite covering all functionality
- **Continuous Integration**: Automated testing on multiple platforms
- **Documentation**: Complete documentation with worked examples

The package is designed for both methodological research and applied data analysis, balancing computational efficiency with user accessibility.