universität innsbruck

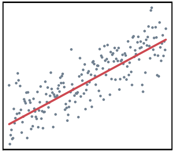
Distributional Regression Forests for Probabilistic Modeling and Forecasting

Lisa Schlosser, Torsten Hothorn, Heidi Seibold, Achim Zeileis

https://eeecon.uibk.ac.at/~zeileis/

Motivation

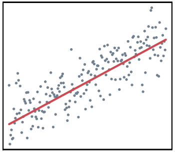
Motivation



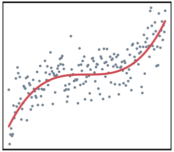
LM, GLM

`lm`
`glm`

Motivation



→



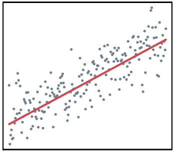
LM, GLM

GAM

`lm`
`glm`

`mgcv`
`VGAM`
...

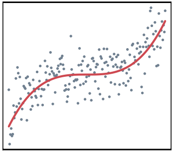
Motivation



LM, GLM

`lm`
`glm`

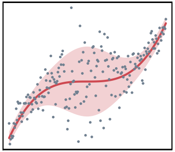
→



GAM

`mgcv`
`VGAM`
`...`

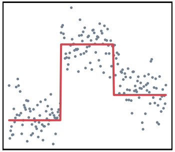
→



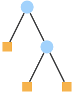
GAMLSS

`gamlss`
`mgcv`
`VGAM`
`gamboostLSS`
`...`

Motivation

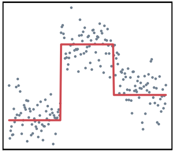


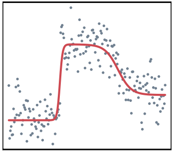
Regression Tree



`rpart`
`party(kit)`

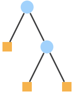
Motivation

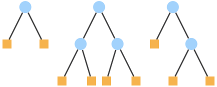




Regression Tree

Random Forest

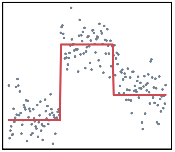
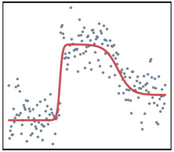
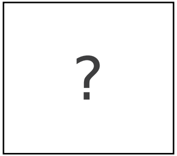




`rpart`
`party(kit)`

`randomForest`
`ranger`
`party(kit)`
...

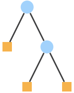
Motivation

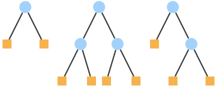
 →  → 

Regression Tree

Random Forest

Distributional trees and forests





`rpart`
`party(kit)`

`randomForest`
`ranger`
`party(kit)`
...

`disttree`
based on
`partykit`

Goals

Distributional:

- Specify the complete probability distribution (including location, scale, and shape).

Tree:

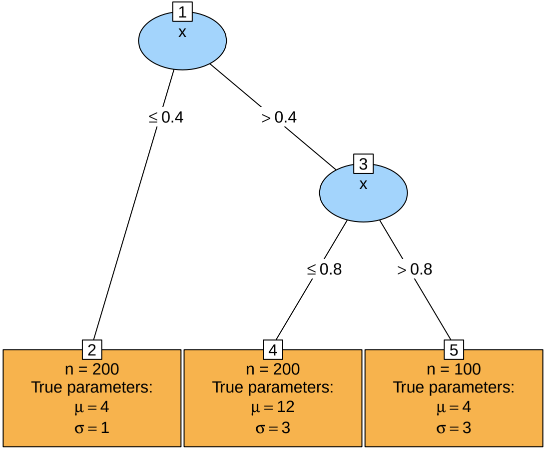
- Automatic detection of steps and abrupt changes.
- Capture non-linear and non-additive effects and interactions.

Forest:

- Smoother effects.
- Stabilization and regularization of the model.

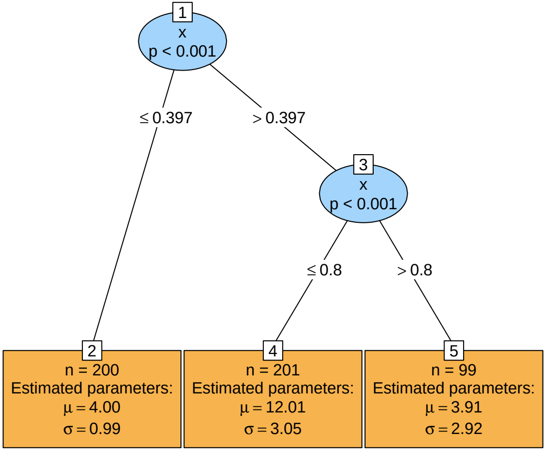
Distributional trees

DGP: $Y \mid X = x \sim \mathcal{N}(\mu(x), \sigma^2(x))$



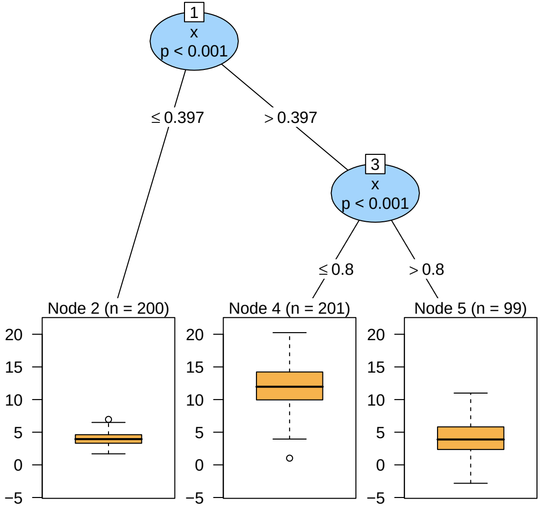
Distributional trees

Model: `disttree(y ~ x)`



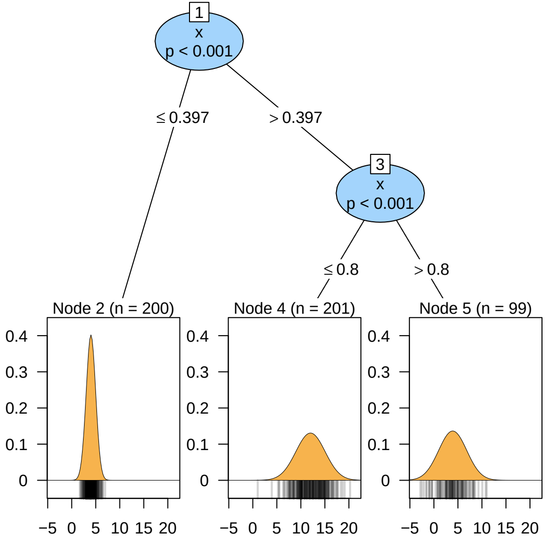
Distributional trees

Model: `disttree(y ~ x)`

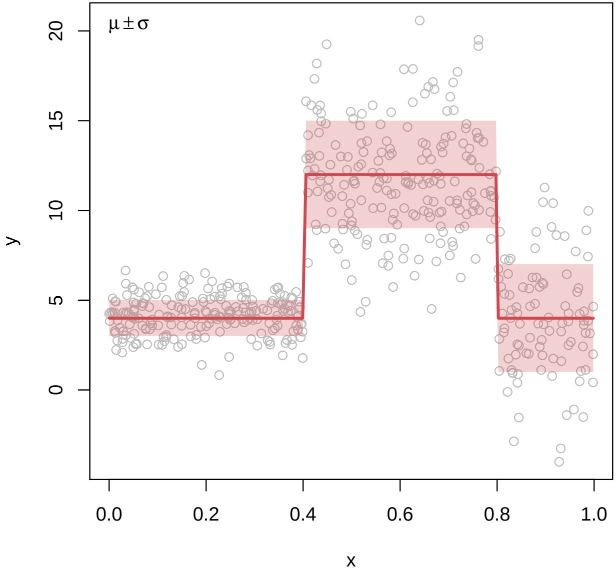


Distributional trees

Model: `disttree(y ~ x)`



Distributional trees



Estimation: Global likelihood

- Specify a parametric distribution family $F(\cdot;\theta)$ with parameter vector $\theta \in \Theta$ capturing location, scale, shape.
- Cumulative distribution function and log-likelihood:

$$\begin{aligned} F(y;\theta) &= \mathbb{P}_\theta(Y \leq y) \\ \ell(\theta;y) &= \log(f(y;\theta)) \end{aligned}$$

- Estimate $\hat{\theta}$ via maximum likelihood based on a learning sample $y_1, \dots, y_n$:

$$\hat{\theta} = \max_{\theta\in\Theta} \sum_{i=1}^{n} \ell(\theta; y_i)$$

Estimation: Adaptive local likelihood

Idea: Covariates captured through adaptive weights.

$$\hat{\theta}(\mathbf{x}) = \max_{\theta \in \Theta} \sum_{i=1}^{n} w_i(\mathbf{x}) \cdot \ell(\theta; y_i).$$

Question: How to choose weighting function $w_i(\mathbf{x})$?

Possible answers: Based on learning sample $y_1, \dots, y_n$ and (possibly new) observation $\mathbf{x}$.

- *Tree:* $w_i(\mathbf{x}) \in \{0, 1\}$ indicates whether $\mathbf{x}$ and y_i are classified into the same subgroup.
- *Forest:* $w_i(\mathbf{x}) \in [0, 1]$ averages the weights for $\mathbf{x}$ and y_i across trees.

Estimation: Distributional trees and forests

Tree:

1. Estimate $\hat{\theta}$ via maximum likelihood (without covariates).
2. Test for associations or instabilities of the scores $\frac{\partial \ell}{\partial \theta}(\hat{\theta}; y_i)$ and each partitioning variable x_i.
3. Split the sample along the partitioning variable with the strongest association or instability. Choose breakpoint with highest improvement in log-likelihood.
4. Repeat steps 1–3 recursively until some stopping criterion is met, yielding B subgroups $\mathcal{B}_b$ with $b = 1, \dots, B$.

Forest: Ensemble of T trees.

- Bootstrap or subsamples.
- Random input variable sampling.

Estimation: Adaptive local likelihood

Estimator:

$$\hat{\theta}(\mathbf{x}) = \max_{\theta \in \Theta} \sum_{i=1}^{n} w_i(\mathbf{x}) \cdot \ell(\theta; y_i)$$

Weights:

$$w_i^{\text{global}}(\mathbf{x}) = 1$$

$$w_i^{\text{tree}}(\mathbf{x}) = \sum_{b=1}^{B} I((\mathbf{x}_i \in \mathcal{B}_b) \wedge (\mathbf{x} \in \mathcal{B}_b))$$

$$w_i^{\text{forest}}(\mathbf{x}) = \frac{1}{T} \sum_{t=1}^{T} \sum_{b=1}^{B^t} \frac{I((\mathbf{x}_i \in \mathcal{B}_b^t) \wedge (\mathbf{x} \in \mathcal{B}_b^t))}{|\mathcal{B}_b^t|}$$

Model specification

Covariates: Automatically through adaptive forest weights.

Response: Distributional specification needed.

- Continuous responses: Gaussian, . . .
- Limited responses: Censored Gaussian, . . .
- Survival times: Exponential, Weibull, . . .
- Count: Poisson, negative binomial, . . .
- Circular: Von Mises, wrapped distributions, . . .

Guidance: Literature, theory, experience, . . .

Model specification

Illustration: Hourly mean wind direction at Innsbruck Airport (at wind speeds higher than $1\,ms^{-1}$).

Distribution: Von Mises with location μ and concentration κ.

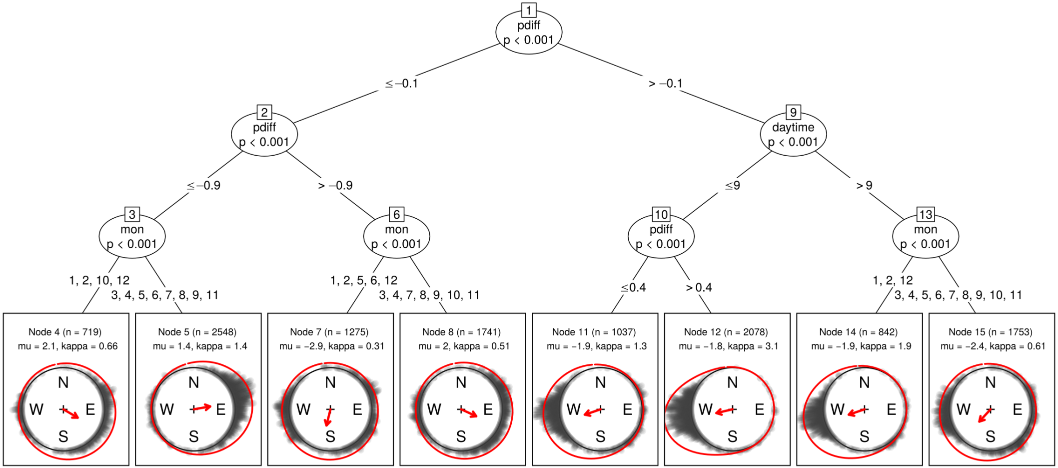
$$f(y;\mu,\kappa) = \frac{1}{2\pi I_0(\kappa)}\, e^{\kappa \cos(y-\mu)},$$

where $I_0(\kappa)$ is the modified Bessel function of the first kind and order 0.

Regressors: Pressure difference between Innsbruck and Kufstein (pdiff, numeric), months (mon, factor), hour of the day (daytime, numeric).

Model: Tree up to depth 3, forest is work in progress.

Model specification



Transformation models

Alternative: When no obvious classic distribution assumption is available.

Advantages:

- Does not require specification of distribution family.
- More flexible framework.

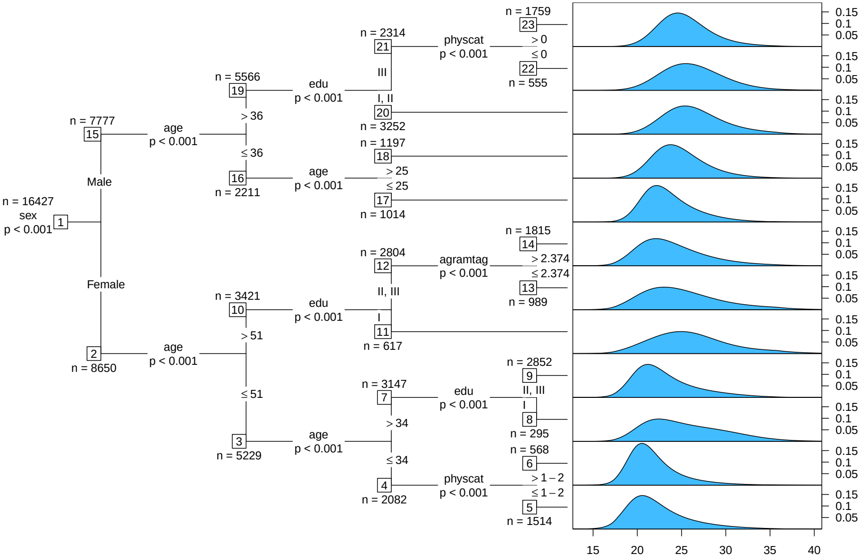
Distribution function:

$$F(y;\theta) = \Phi(\mathbf{a}_{Bs,d}(y)^\top \theta)$$

- $\mathbf{a}_{Bs,d}(y)^\top \theta$ is a smooth, monotone Bernstein polynomial of degree d.
- $d = 1$ corresponds to $\mathcal{N}(\mu, \sigma^2)$.
- $d = 5$ is surprisingly flexible.

Example: Body Mass Index explained by lifestyle factors (Switzerland).

Transformation models



Software

Package: *disttree* available on R-Forge at

`https://R-Forge.R-project.org/projects/partykit/`

Main functions:

`distfit`	Distributional fit (ML, `gamlss.family`/custom `list`). No covariates.
`disttree`	Distributional tree (`ctree`/`mob` + `distfit`). Covariates as partitioning variables.
`distforest`	Distributional forest (`disttree` ensemble). Covariates as partitioning variables.

References

Schlosser L, Hothorn T, Stauffer R, Zeileis A (2018). "Distributional Regression Forests for Probabilistic Precipitation Forecasting in Complex Terrain." *arXiv 1804.02921*, arXiv.org E-Print Archive. `http://arxiv.org/abs/1804.02921`

Hothorn T, Zeileis A (2017). "Transformation Forests." *arXiv 1701.02110*, arXiv.org E-Print Archive. `http://arxiv.org/abs/1701.02110`

Hothorn T, Zeileis A (2015). "partykit: A Modular Toolkit for Recursive Partytioning in R." *Journal of Machine Learning Research*, **16**, 3905–3909. `http://www.jmlr.org/papers/v16/hothorn15a.html`

Hothorn T, Hornik K, Zeileis A (2006). "Unbiased Recursive Partitioning: A Conditional Inference Framework." *Journal of Computational and Graphical Statistics*, **15**(3), 651–674. `doi:10.1198/106186006X133933`

Zeileis A, Hothorn T, Hornik K (2008). "Model-Based Recursive Partitioning." *Journal of Computational and Graphical Statistics*, **17**(2), 492–514. `doi:10.1198/106186008X319331`

Stasinopoulos DM, Rigby RA (2007). "Generalized Additive Models for Location Scale and Shape (GAMLSS) in R." *Journal of Statistical Software*, **23**(7), 1–46. `doi:10.18637/jss.v023.i07`