Random Planted Forest

Munir Eberhardt Hiabu

Why apply an algorithm?

- Why apply an algorithm?
  - Prediction
    - We want to predict the response of new observations
  - Inference
    - We want to understand the data generating mechanism

See also Breiman (2001) and Efron (2020) for related discussions about the different culture in statistics and machine learning.

Interpretability

- Interpretability of an algorithm is clearly important if inference is the goal.
- But it may also be important in sole prediction tasks
  - decisions shall be explained to stakeholders
  - risk/robustness of prediction shall be assessed.

Accuracy vs Interpretability?

Breiman (2001):

> Occam's Razor, long admired, is usually interpreted to mean that simpler is better. Unfortunately, in prediction, accuracy and simplicity(interpretability) are in conflict.

> […] models that best emulate nature in terms of predictive accuracy are also the most complex and inscrutable.

In the following I want to challenge this point of view.

A functional decomposition

Assume a data set with d features . Also assume that we can describe the regression function m by a (q-th) order $(q << p)$ functional decomposition:

$$m(x) = m_0 + \sum_{k=1}^{d} m_k(x_k) + \sum_{k_1<k_2} m_{k_1 k_2}(x_{k_1}, x_{k_2}) + \cdots + \sum_{k_1<\cdots<k_q} m_{k_1,\ldots,k_q}(x_{k_1}, \ldots, x_{k_q}).$$

- Optimal rates of convergence under the assumption that m has two continuous partial derivatives:

Model general	$d = 6$	**Comparable sample sizes for $d = 6$**
Full model	$O_p(n^{-2/(d+4)}) = O_p(n^{-1/5})$	1 000 000
Interaction (q)	$O_p(n^{-2/(q+4)})$	1 000 -- 1 000 000
Interaction (q=2)	$O_p(n^{-1/3})$	4 000
Additive (q=1)	$O_p(n^{-2/5})$	1 000

Random Forest in additive models

Random Forests do not perform well in additive models.

Example

$$Y_i = m(X_i) + \varepsilon_i,$$

with X_i uniform in $[0, 1]^d$ and

$$m(x) = \sum_{k=1}^{d} 1\{x_k \leq 0.5\}.$$

- Need $O(2^d)$ splits/final leaves to approximate m well.
- Have only $O_P(n/2^d)$ data points in each leave which may be very small for large dimension d.

Random Planted Forest: A Directly Interpretable Tree Ensemble

Random Planted Forest: A Directly Interpretable Tree Ensemble

This is joint work with

Joseph Meyer
Heidelberg University

Enno Mammen
Heidelberg University

Paper is available on https://arxiv.org/abs/2208.06151

Random Planted Forest: The algorithm

Random Planted Forest The algorithm

- Start with a fixed structure and become more complex in a data driven way along a pre-specified path

$$m(x) = m_0 + \sum_{k=1}^{d} m_k(x_k) + \sum_{k<l} m_{kl}(x_k, x_l) + \sum_{j<k<l} m_{jkl}(x_j, x_k, x_l) + \cdots.$$

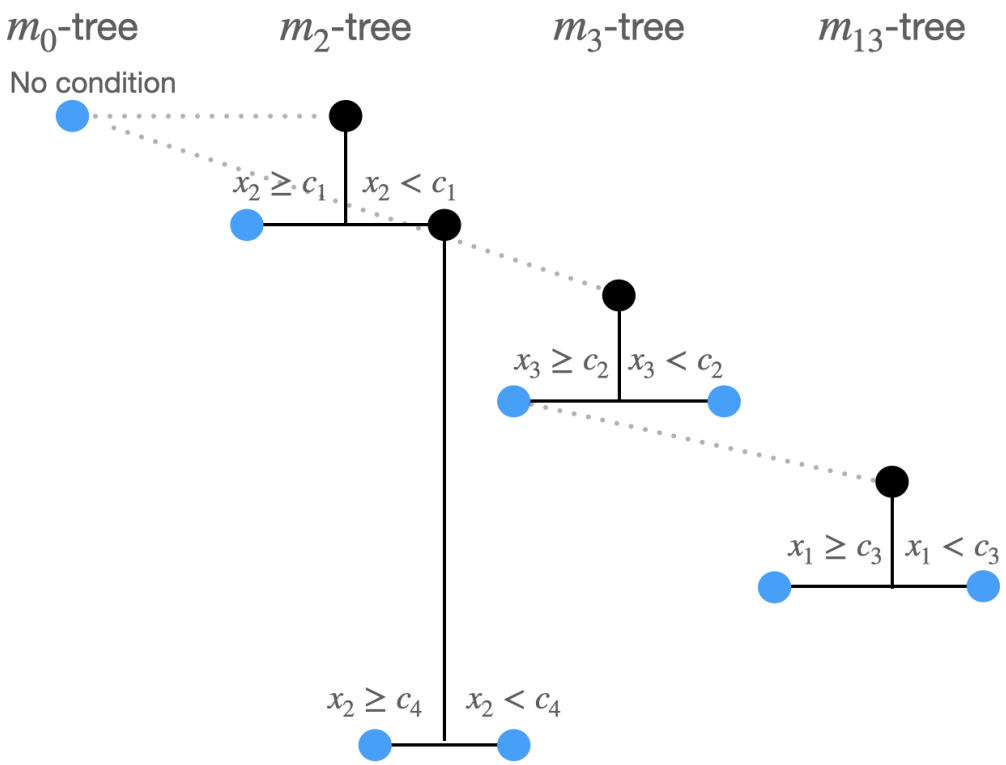
- We aim to estimate the components $m_k, m_{kl}, \ldots$ simultaneously
- This is different to Lou et al. (2013) and related literature that first specify the components and then estimate them iteratively.

How:

- Replace tree in classical random forest by a family of trees that grow simultaneously.
- Each tree corresponds to a component in the functional decomposition.
  - first order approximation $\longrightarrow$ additive model.
  - second order approximation $\longrightarrow$ fitting main term and pairwise interactions.
  - order of approximation = number of covariates $\Rightarrow$ no restriction on functional form

Random Planted Forest The algorithm

Setting: dim=3, t_{try}=0.6, $split_{try}$=5, n_{splits}=4



Split 1 | Split 2 | Split 3 | Split 4

- 3 possible combinations:
  - $m_0 \rightarrow m_1 : x_1$
  - $m_0 \rightarrow m_2 : x_2$
  - $m_0 \rightarrow m_3 : x_3$
- $t_{try} : 0.6 \times 3 = 1.8 \rightarrow$ 2 viable combinations randomly picked, say:
  - $m_0 \rightarrow m_1 : x_1$,
  - $m_0 \rightarrow m_2 : x_2$
- $split_{try}$:For each viable split option we consider 5 randomly picked split points $\rightarrow 2 \times 5 = 10$ split options.
- Compare the 10 split options: $\sum_i (\widehat{m}(X_i) - Y_i)^2$
- $(m_2 tree \rightarrow x_2, c_1)$ produces minimal least squares loss.

Random Planted Forest in additive models

Random Planted Forests does perform well in additive models.

Example

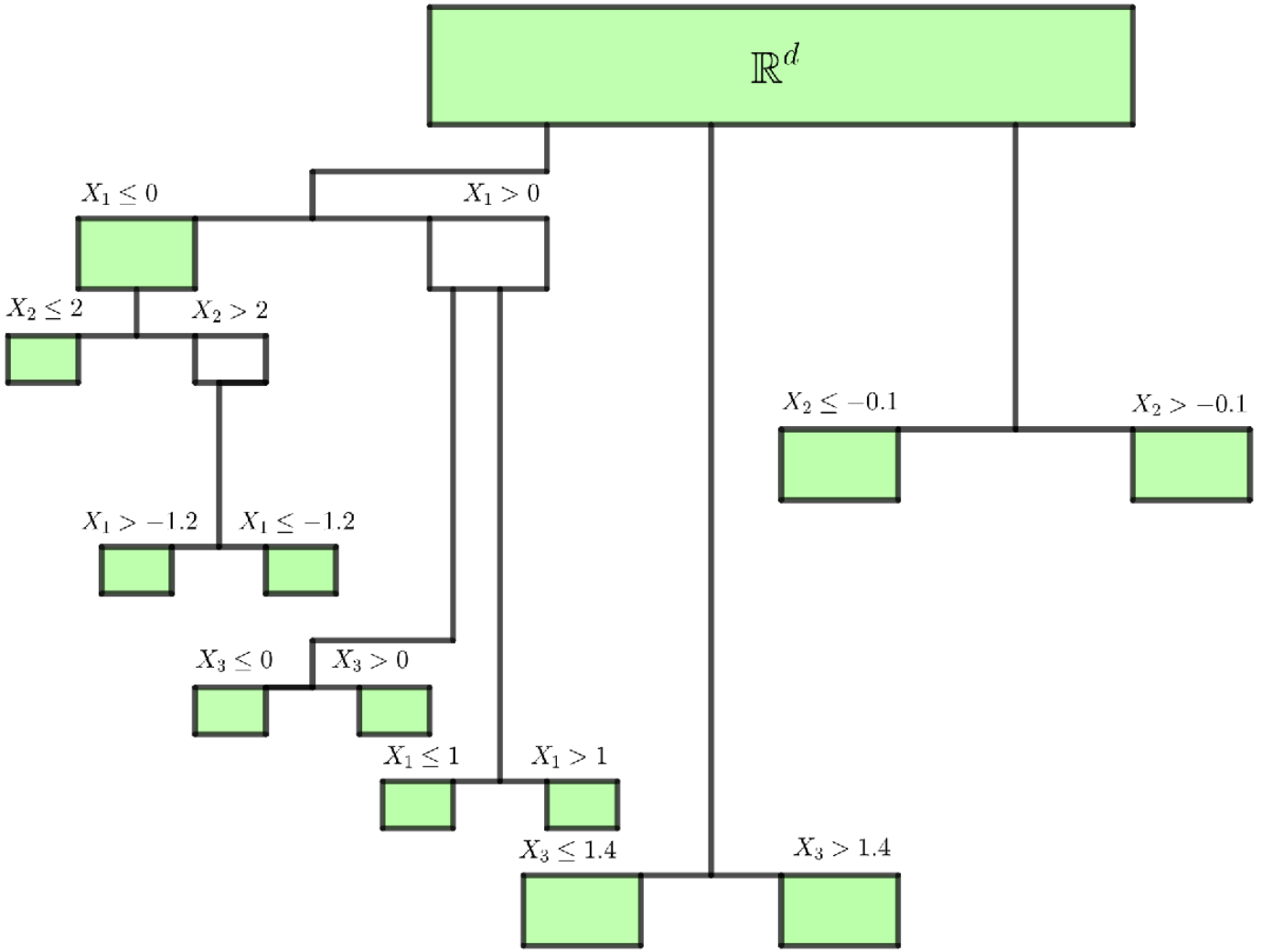
$$Y_i = m(X_i) + \varepsilon_i,$$

with X_i uniform in $[0, 1]^d$ and

$$m(x) = \sum_{k=1}^{d} 1\{x_k \leq 0.5\}.$$

- Need $O(d)$ instead of $O(2^d)$ splits/final leaves to approximate m well.
- Have $O_P(n/d)$ data points in each leave which may be large compared to $O_P(n/2^d)$ for large dimension d.

Random Planted Forest The algorithm (alternative representation)



Random Planted Forest: Simulation results

Random Planted Forest Accuracy: Simulation

- We will evaluate how Random Planted forest deals with combinations of interactions, variable selection/sparsity, non-linearity/jumps.. Simulations are run with optimal parameters

Regression setting (In all simulations):

$$Y_i = m(x_i) + \varepsilon_i, \quad i = 1, \ldots, 500, x_i \in \mathbb{R}^{\dim}$$

- $\dim = 30$
- $Corr(X_j, X_k) \approx 0.3,\ j \neq k$
- $\varepsilon_i \sim N(0,1), iid$

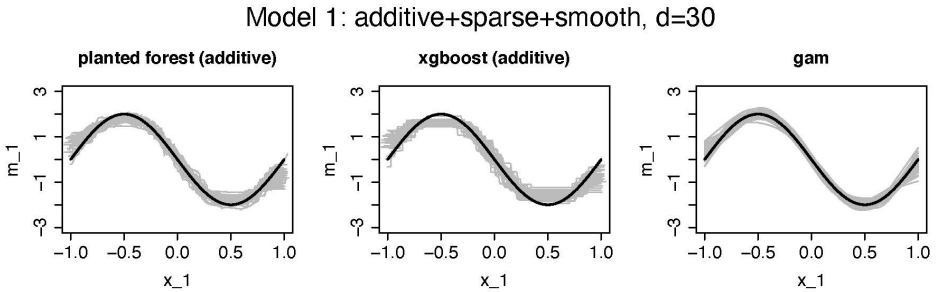
Comparison between

- Random Planted Forest
- XGboost – a gradient boosting variant
- gam via mgcv package (= GLM with smoothing splines)

First setting | Second setting | Third setting:

Additive, Sparse (2/30 features), Non-linear (sin-curve)

- $m(x) = m_1(x_1) + m_2(x_2) + \cdots + m_{30}(x_{30})$
- $m_1(x_1) = -2\sin(\pi x_1),\ m_2(x_2) = 2\sin(\pi x_2)$
- $m_3(x_3) = m_4(x_4) = \cdots = m_{30}(x_{30}) = 0$



- True function: black solid line. Grey lines: 40 Monte Carlo simulations
- xgboost (additive=depth=1), planted forest (additive= max interaction=1)

Random Planted Forest Simulation results

sparse+smooth+additive. Table shows Mean Squared Error (Standard deviation)

Method	Assumption	dim=4	dim=10	dim=30
xgboost	additive	0.119 (0.021)	0.142 (0.021)	0.176 (0.027)
xgboost	-	0.141 (0.024)	0.166 (0.028)	0.193 (0.033)
xgboost-CV	-	0.139 (0.028)	0.152 (0.029)	0.194 (0.035)
rpf	additive	0.087 (0.018)	0.086 (0.017)	0.097 (0.019)
rpf	interaction(2)	0.107 (0.015)	0.121 (0.025)	0.142 (0.026)
rpf	-	0.112 (0.017)	0.134 (0.026)	0.162 (0.028)
rpf-CV	-	0.103 (0.02)	0.102 (0.035)	0.105 (0.022)
rf	-	0.209 (0.021)	0.252 (0.027)	0.3 (0.029)
sbf	additive+smooth	0.071 (0.026)	0.134 (0.013)	0.388 (0.073)
gam	additive+smooth	0.033 (0.012)	0.035 (0.013)	0.058 (0.021)
BART	-	0.085 (0.019)	0.076 (0.017)	0.091 (0.023)
BART-CV	-	0.09 (0.019)	0.081 (0.014)	0.09 (0.02)
MARS	smooth	0.054 (0.014)	0.061 (0.025)	0.076 (0.031)

Random Planted Forest Simulation results

sparse+jump+additive. Table shows Mean Squared Error (Standard deviation)

Method	Assumption	dim=4	dim=10	dim=30
xgboost	additive	0.19 (0.029)	0.282 (0.044)	0.401 (0.045)
xgboost	-	0.198 (0.031)	0.265 (0.053)	0.286 (0.034)
xgboost-CV	-	0.209 (0.028)	0.281 (0.052)	0.313 (0.058)
rpf	additive	0.159 (0.033)	0.198 (0.075)	0.179 (0.041)
rpf	interaction(2)	0.185 (0.028)	0.24 (0.066)	0.259 (0.043)
rpf	-	0.192 (0.026)	0.251 (0.065)	0.282 (0.043)
rpf-CV	-	0.169 (0.033)	0.207 (0.072)	0.183 (0.042)
rf	-	0.274 (0.035)	0.322 (0.05)	0.375 (0.037)
sbf	additive+smooth	0.342 (0.049)	0.603 (0.053)	1.112 (0.138)
gam	additive+smooth	0.41 (0.047)	0.406 (0.027)	0.431 (0.06)
BART	-	0.177 (0.047)	0.162 (0.038)	0.157 (0.034)
BART-CV	-	0.179 (0.051)	0.163 (0.041)	0.159 (0.036)
MARS	smooth	0.751 (0.136)	0.74 (0.104)	0.687 (0.123)

Random Planted Forest Simulation results

sparse+smooth+hierarchical interaction. Table shows Mean Squared Error (Standard deviation)

Method	Assumption	dim=4	dim=10	dim=30
xgboost	-	0.374 (0.035)	0.481 (0.064)	0.557 (0.089)
xgboost-CV	-	0.393 (0.051)	0.499 (0.058)	0.563 (0.089)
rpf	interaction(2)	0.248 (0.038)	0.327 (0.045)	0.408 (0.07)
rpf	-	0.263 (0.034)	0.357 (0.044)	0.452 (0.076)
rpf-CV	-	0.277 (0.039)	0.366 (0.051)	0.463 (0.083)
rf	-	0.432 (0.039)	0.575 (0.061)	0.671 (0.08)
BART	-	0.214 (0.03)	0.223 (0.04)	0.252 (0.037)
BART-CV	-	0.242 (0.043)	0.276 (0.053)	0.315 (0.047)
MARS	smooth	0.355 (0.089)	0.282 (0.038)	0.414 (0.126)
1-NN	no noise	2.068 (0.156)	5.988 (0.624)	11.059 (0.676)
average	no covariates	8.366 (0.43)	8.086 (0.246)	8.207 (0.496)

Random Planted Forest Simulation results

sparse+smooth+pure interaction. Table shows Mean Squared Error (Standard deviation)

Method	Assumption	dim=4	dim=10	dim=30
xgboost	-	0.417 (0.082)	0.797 (0.16)	1.381 (0.234)
xgboost-CV	-	0.443 (0.078)	0.872 (0.136)	1.497 (0.326)
rpf	interaction(2)	0.416 (0.082)	1.289 (0.224)	1.822 (0.208)
rpf	-	0.219 (0.035)	0.556 (0.143)	1.186 (0.236)
rpf-CV	-	0.233 (0.033)	0.603 (0.163)	1.313 (0.253)
rf	-	0.304 (0.047)	0.744 (0.305)	1.295 (0.317)
BART	-	0.168 (0.022)	0.172 (0.032)	0.202 (0.021)
BART-CV	-	0.192 (0.03)	0.199 (0.039)	0.223 (0.025)
MARS	smooth	0.245 (0.088)	0.831 (0.728)	0.429 (0.403)
1-NN	no noise	1.323 (0.117)	2.642 (0.317)	4.173 (0.413)
average	no covariates	2.187 (0.125)	2.226 (0.174)	2.177 (0.146)

Random Planted Forest: Theoretical Results

Random Planted Forest Theoretical Results

Assumptions (for stylized Random Planted Forest)

- The components $m_1, \ldots, m_{k_1,\ldots,k_q}$ are twice continuously differentiable functions.
- Conditionally on X_i and the iterative construction of the leaves, the error variables ε_i have mean zero, variance bounded by a constant, and the products $\varepsilon_i \varepsilon_j$ are mean zero for $i \neq j$. This excludes splitting choices based on the CART criterion.
- Conditionally on X_i the iterative construction of the leaves in the different trees are i.i.d.
- For simplicity we also assume that the planted trees are based on the original data and not on bootstrap samples.

Random Planted Forest Theoretical Results

Challenges in analyzing Random Planted Forest (compared to Random Forest)

In Random Forest

- the tree estimator only depends on the leaves but not on other structural elements of the tree, and in particular not on the way how the tree has grown.
- tree estimates are given by leaf averages

In our setting both facts do not hold. This creates the main difficulty in our theory.

Random Planted Forest Theoretical Results

Main Results

- Up to logarithmic factors, the forest estimator achieves optimal rates for $q \leq 2$, i.e. same rates as for estimating a nonparametric regression function with one- or two-dimensional argument, if the number `nsplits` of splits is chosen optimally $(O(n^{q/5}))$.
- The forest estimator achieves faster rates than the tree family estimator.

Random Planted Forest Idea of proof for $q = 1$: tree family

After s splits we have

$$\hat{m}_{k_s}^{(s),l}(x_{k_s}) = \hat{\bar{m}}_{k_s}^{(s),l}(x_{k_s}) - \sum_{k' \neq k_s} \int \frac{\hat{p}_{k_s,k'}^{(s),l}(x_{k_s}, x_{k'})}{\hat{p}_{k_s}^{(s),l}(x_{k_s})} \hat{m}_{k'}^{(s-1),l}(x_{k'}) \mathrm{d}x_{k'}$$

After the last iterations $s = S$ one gets that

$$\hat{m}_k^l(x_k) \approx \hat{\bar{m}}_k^{(S),l}(x_k) - \sum_{k' \neq k} \int \frac{p_{k,k'}(x_k, x_{k'})}{p_k(x_k)} \hat{m}_{k'}^l(x_{k'}) \mathrm{d}x_{k'}$$

for $1 \leq k \leq d$. Subtracting

$$m_k(x_k) = \bar{m}_k(x_k) - \sum_{k' \neq k} \int \frac{p_{k,k'}(x_k, x_{k'})}{p_k(x_k)} m_{k'}(x_{k'}) \mathrm{d}x_{k'}.$$

gives an integral equation for $(\hat{m}_k^l - m_k)_{k=1}^d$

Random Planted Forest Idea of proof for $q = 1$: tree family

$$\hat{m}_k^l(x_k) - m_k(x_k) \approx \hat{\bar{m}}_k^{(S),l}(x_k) - \bar{m}_k(x_k) - \sum_{k' \neq k} \int \frac{p_{k,k'}(x_k, x_{k'})}{p_k(x_k)} \big(\hat{m}_{k'}^l(x_{k'}) - m_{k'}(x_{k'})\big) \mathrm{d}x_{k'}.$$

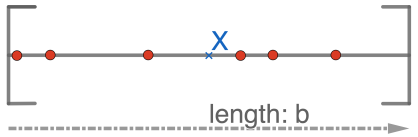
Under assumptions on the kernel $\frac{p_{k,k'}(x_k,x_{k'})}{p_k(x_k)}$ it holds:

The solution $\left(\hat{m}_k^l(x_k) - m_k(x_k)\right)_{k=1,\dots,d}$ is of the same order as the intercept $\left(\hat{\bar{m}}_k^{(S),l}(x_k) - \bar{m}_k(x_k)\right)_{k=1,\dots,d}$,

which as a piecewise constant estimator has optimal rate $n^{-1/3}$ if **`nsplits`** is chosen of order $n^{1/3}$.

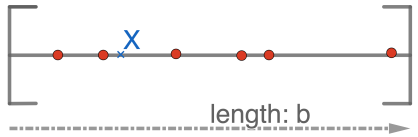
Random Planted Forest Idea of proof for $q = 1$: from tree family to forest

- Smoothing kernel methods put the same weight left and right to every point



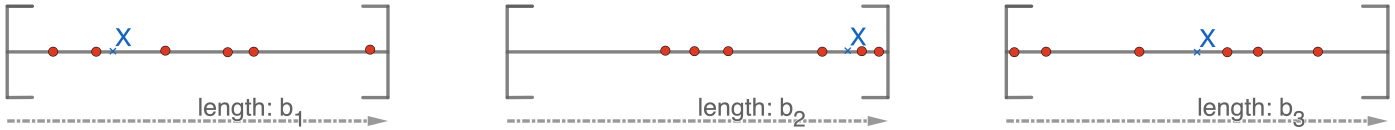
Bias = $O_p(b^2)$

- Histograms do not put the same weight left and right to every point



Bias= $O_p(b)$

- Random Planted Forest do via the $split_{try}$ parameter



Random Planted Forest
Bias= $O_p(b^2)$,
$b = L^{-1} \sum_{l=1}^{L} (b_l)$.

- $\longrightarrow$ For Random Planted Forest to achieve optimal convergence rate it is essential that the split points are picked from a random selection
- The intercept $(\hat{\bar{m}}_k^{(S)} - \bar{m}_k)_{k=1}^d$ has a faster rate of convergence than $(\hat{\bar{m}}_k^{(S),b} - \bar{m}_k)_{k=1}^d$: $n^{-2/5}$ (if `nsplits` is chosen of order $n^{1/5}$).
- This rates carries over to the solution $(\hat{m}_k - m_k)_{k=1}^d$ of the integral equation.

Random Planted Forest Idea of proof for $q = 1$

Comment on limitation of theory

- A central argument in the proof is the convergence of $\hat{p}_{k,k'}^{(s),b}(x_k, x_{k'})$ to $p_{k,k'}(x_k, x_{k'})$
- For $q \geq 3$ this convergence is not fast enough to carry over the above arguments of $q = 1$. In particular, one cannot achieve rates of convergence that are the same as for the estimation of q-dimensional functions. For $q \geq 4$ one would not even get consistency of $\hat{p}_{k,k'}^{(s),b}(x_k, x_{k'})$.

Random Planted Forest: Identifying the components

Random Planted Forest Identifying the components

Definition [Marginal identification]

Consider a regression function $m : \mathbb{R}^d \to \mathbb{R}$ with functional decomposition

$$m(x) = \sum_{S \subseteq \{1,\dots,d\}} m_S$$

If for every $S \subseteq \{1, \dots, d\}$,

$$\sum_{T : T \cap S \neq \emptyset} \int m_T(x_T) p_S(x_S) \mathrm{d}x_S = 0,$$

we say that the functional decomposition satifies the marginal identification.

Random Planted Forest Identifying the components

Theorem [Marginal identification $\leftrightarrow$ Interventional SHAP]

If $\hat{m}_n = \sum_{S \subseteq \{1,\dots,d\}} \hat{m}_S$ satisfies the marginal identification, then

- Interventional SHAP values are weighted averages of the components
  - interaction component is equally split between involved features:

$$\phi_k(x) = \hat{m}_k(x_k) + \frac{1}{2}\sum_j \hat{m}_{kj}(x_{kj}) + \cdots + \frac{1}{d}\hat{m}_{1,\dots,d}(x_{1,\dots,d}).$$

Corollary [Marginal identification $\leftrightarrow$ PDP]

If $\hat{m}_n = \sum_S \hat{m}_S$ satisfies the marginal identification, then the partial dependence plot of feature k is

$$\xi_k(x_k) = \int \hat{m}_n(x) p_{X_{-k}}(x_{-k}) \mathrm{d}x_{-k} = \hat{m}_0 + \hat{m}_k(x_k).$$

- Under certain causal assumptions, the component $\hat{m}_k$ can be interpreted as average natural direct effect of feature k on the output $\hat{m}_n(x)$.

Random Planted Forest: An ongoing project

Random Planted Forest An ongoing project

Joseph Meyer
Heidelberg University

Enno Mammen
Heidelberg University

Jinyang Liu
University of Copenhagen

Lukas Burk
The Leibniz Institute for Prevention Research and Epidemiology - BIP

Marvin Wright
The Leibniz Institute for Prevention Research and Epidemiology - BIP

Random Planted Forest An ongoing project

randomPlantedForest

`randomPlantedForest` implements “Random Planted Forest”, a directly interpretable tree ensemble (arxiv).

Installation

You can install the development version of `randomPlantedForest` from GitHub with

```
# install.packages("remotes")
remotes::install_github("PlantedML/randomPlantedForest")
```

or from r-universe with

```
install.packages("randomPlantedForest", repos = "https://plantedml.r-universe.dev")
```

Random Planted Forest for insurance claim prediction

Random Planted Forest for insurance claim prediction

```r
library(CASdatasets)
data(freMPL1)
data(freMPL2)
data(freMPL3)
data(freMPL4)

freMPL3 <- subset( freMPL3 , select = -DeducType )
freMPL4 <- subset( freMPL4 , select = -DeducType )
freMPL <- rbind(freMPL1,freMPL2,freMPL3,freMPL4)
mydata <- freMPL[,-c(3,4,19)]

head(mydata)
```

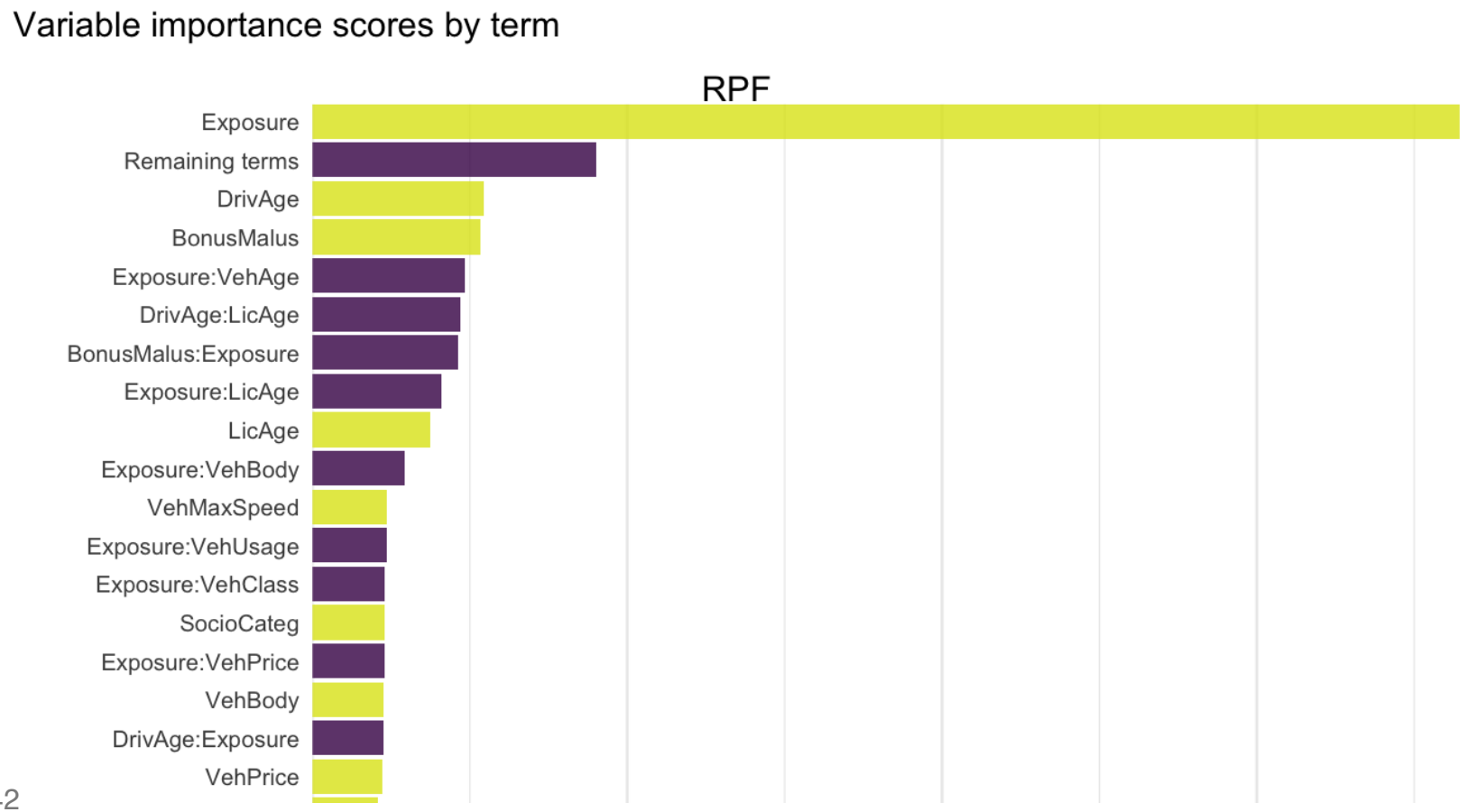
```
  Exposure LicAge VehAge Gender MariStat SocioCateg              VehUsage
1    0.583    366      2 Female    Other       CSP1           Professional
2    0.200    187      0   Male    Alone      CSP55 Private+trip to office
3    0.083    169      1 Female    Other       CSP1           Professional
4    0.375    170      1 Female    Other       CSP1           Professional
5    0.500    224      3   Male    Other      CSP47           Professional
6    0.499    230      3   Male    Other      CSP47           Professional
  DrivAge HasKmLimit BonusMalus        VehBody VehPrice
1      55          0         72          sedan        D
2      34          0         80       microvan        K
3      33          0         63 other microvan        L
4      34          0         63 other microvan        L
5      53          0         72            van        L
6      53          0         68            van        L
                    VehEngine VehEnergy  VehMaxSpeed VehClass RiskVar
1                   injection   regular 160-170 km/h        B      15
2 direct injection overpowered    diesel 170-180 km/h       M1      20
3 direct injection overpowered    diesel 170-180 km/h       M1      17
4 direct injection overpowered    diesel 170-180 km/h       M1      17
5 direct injection overpowered    diesel 140-150 km/h        0      19
6 direct injection overpowered    diesel 140-150 km/h        0      19
         Garage ClaimInd
1          None        0
2          None        0
3          None        0
```

Random Planted Forest for insurance claim prediction

```r
library(randomPlantedForest)

rpf_fit   = rpf(ClaimInd~.,
               data=train,
               ntrees = 50,
                       max_interaction=2,
                       splits =300,
                       split_try = 2,
                       t_try=0.1,
                       nthreads=10,
                       epsilon=0.001,
                       delta=0.001,
                       loss="exponential",
                       predict_type = "prob")

glex_rpf <- glex(rpf_fit, test)
```

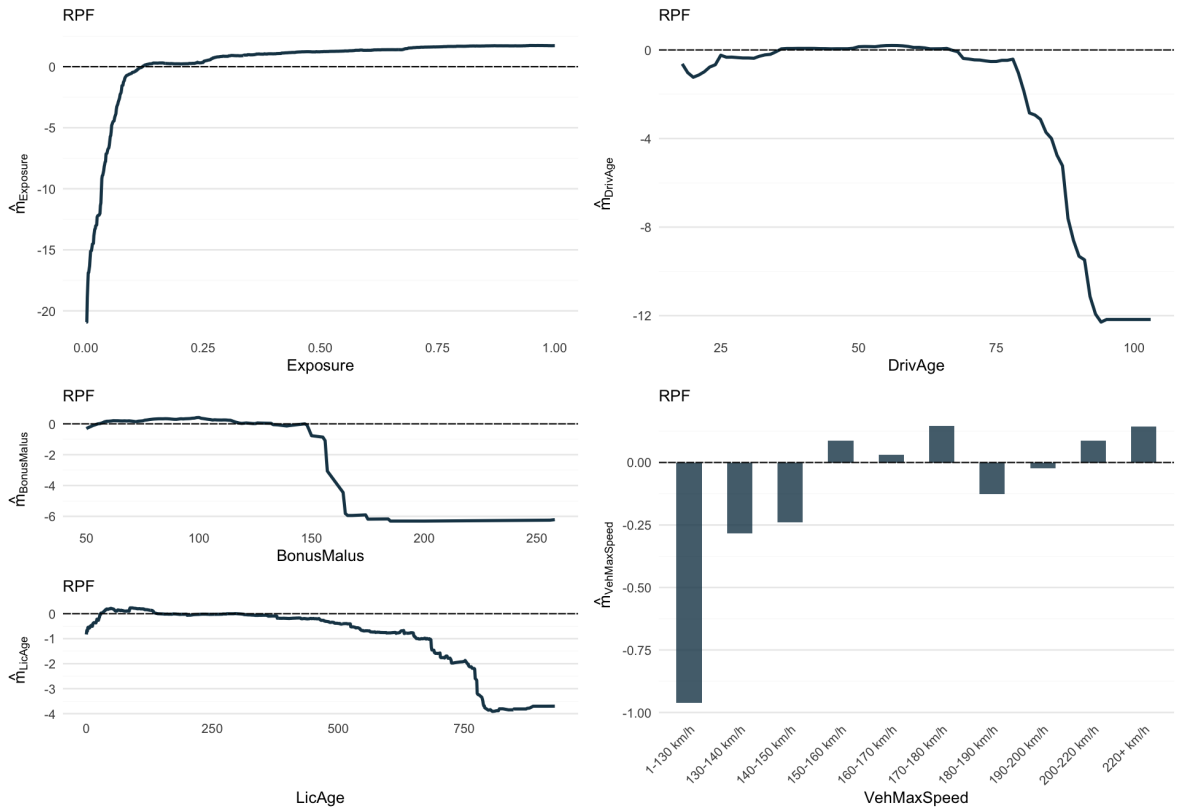
Random Planted Forest for insurance claim prediction

```r
vi_rpf <- glex_vi(glex_rpf)

p_vi <- autoplot(vi_rpf, threshold = .03) +
  labs(title = NULL, tag = "RPF")

p_vi + plot_annotation(title = "Variable importance scores by term") &
  theme(plot.tag.position = "top")
```



Random Planted Forest for insurance claim prediction

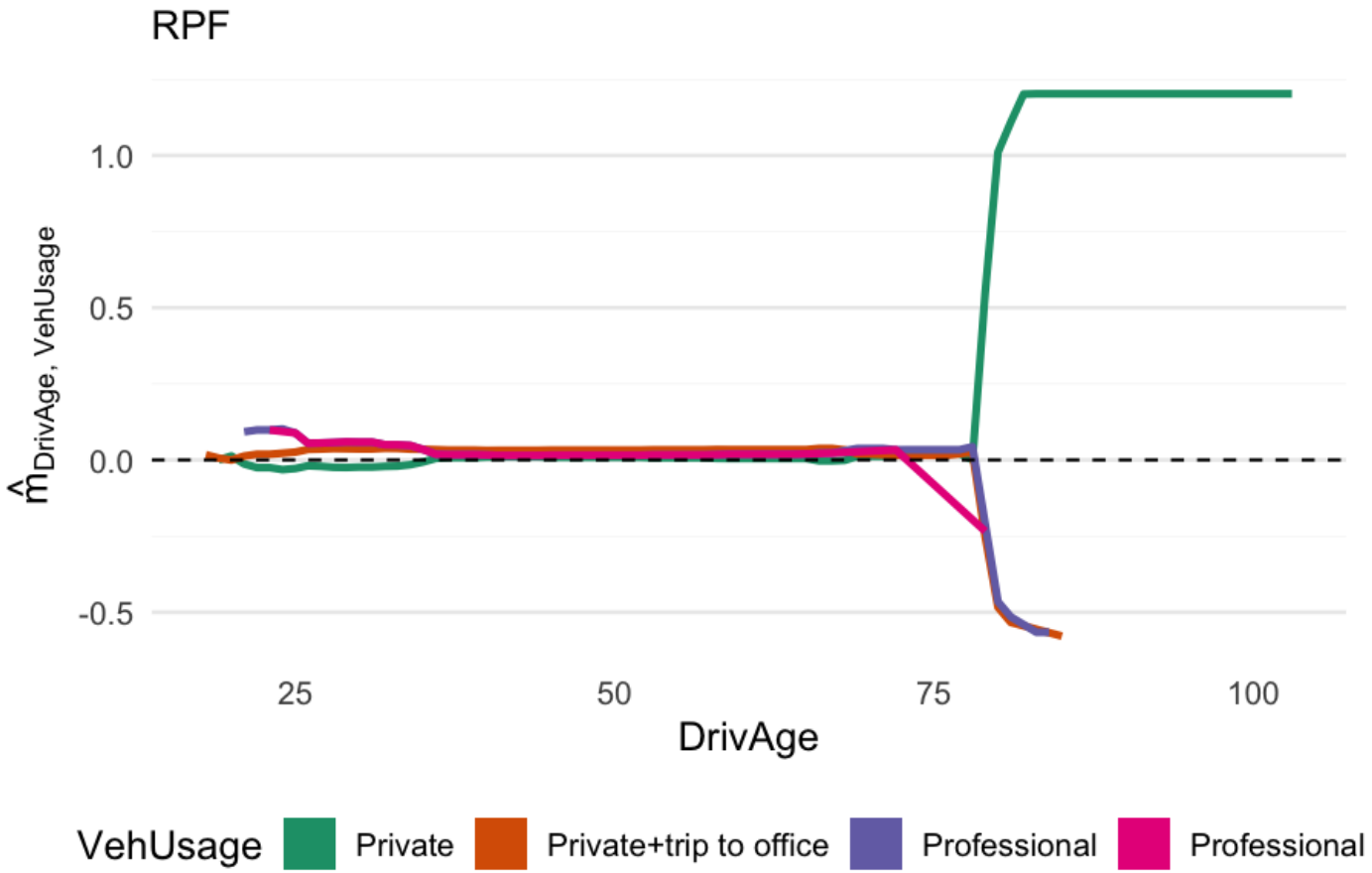
```
p1 <- autoplot(glex_rpf, "Exposure") + labs(subtitle = "RPF")
p2 <- autoplot(glex_rpf, "DrivAge") + labs(subtitle = "RPF")
p3 <- autoplot(glex_rpf, "BonusMalus") + labs(subtitle = "RPF")
p4 <- autoplot(glex_rpf, "LicAge") + labs(subtitle = "RPF")
p5 <- autoplot(glex_rpf, "VehMaxSpeed", las = 2) + labs(subtitle = "RPF")

p11 + p2 + p3 / p4 + p5 + theme(axis.text.x = element_text(angle = 45)) +  plot_layout(heights = c(.5, .5))
```



Random Planted Forest for insurance claim prediction

- Two-way interaction: `DrivAge`, `VehUsage`

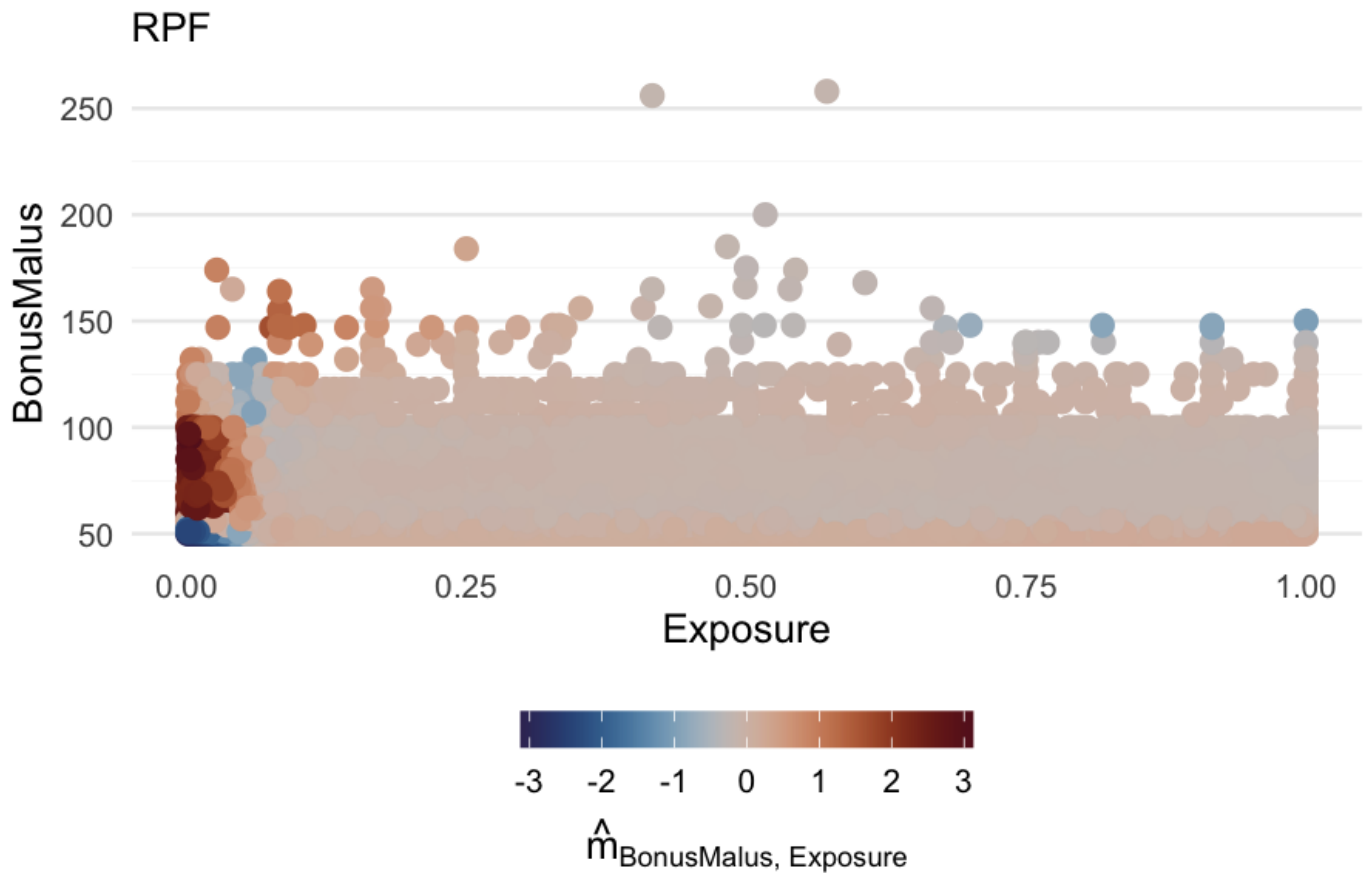
```
1 autoplot(glex_rpf, c("DrivAge","VehUsage")) + labs(subtitle = "RPF") +
2   theme(plot.tag.position = "top")
```



Random Planted Forest for insurance claim prediction

- Two-way interaction: `Exposure`, `Bonus Malus`

```
1 autoplot(glex_rpf, c("DrivAge","VehUsage")) + labs(subtitle = "RPF") +
2   theme(plot.tag.position = "top")
```



Thank You!

References

Breiman, Leo. 2001. “Statistical Modeling: The Two Cultures (with Comments and a Rejoinder by the Author).” *Statistical Science* 16 (3): 199–231.

Efron, Bradley. 2020. “Prediction, Estimation, and Attribution.” *International Statistical Review* 88: S28–59.

Lou, Yin, Rich Caruana, Johannes Gehrke, and Giles Hooker. 2013. “Accurate Intelligible Models with Pairwise Interactions.” In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 623–31.