Generative Forests

Richard Nock

Mathieu Guillame-Bert

Google Research

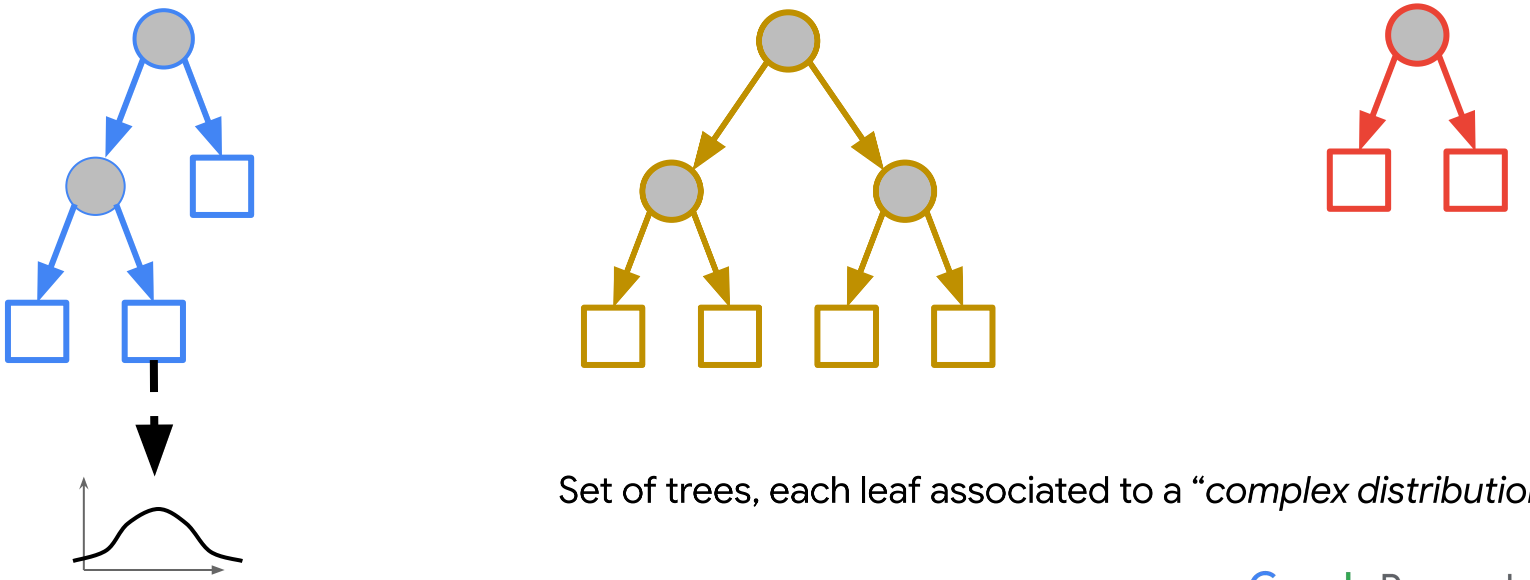
Summary

- We introduce new generative models for tabular data, **Generative Forests**:
  - Natively model any kind of tabular data, fast generation
  - Also enable efficient missing data imputation *and* density estimation
- **Training**:
  - Efficient, *boosting compliant*
  - Reduction trick from binary supervised decision trees (top-down) induction
  - Natively processes data with missing values
- **Implementation**: standard top-down DT induction routines (many packages)
- **Compute**: cheap *on purpose*

Google Research

GFs vs tree-based generators: **key difference**

- An Adversarial Random Forest (ARF, Watson et al., AISTATS'23)



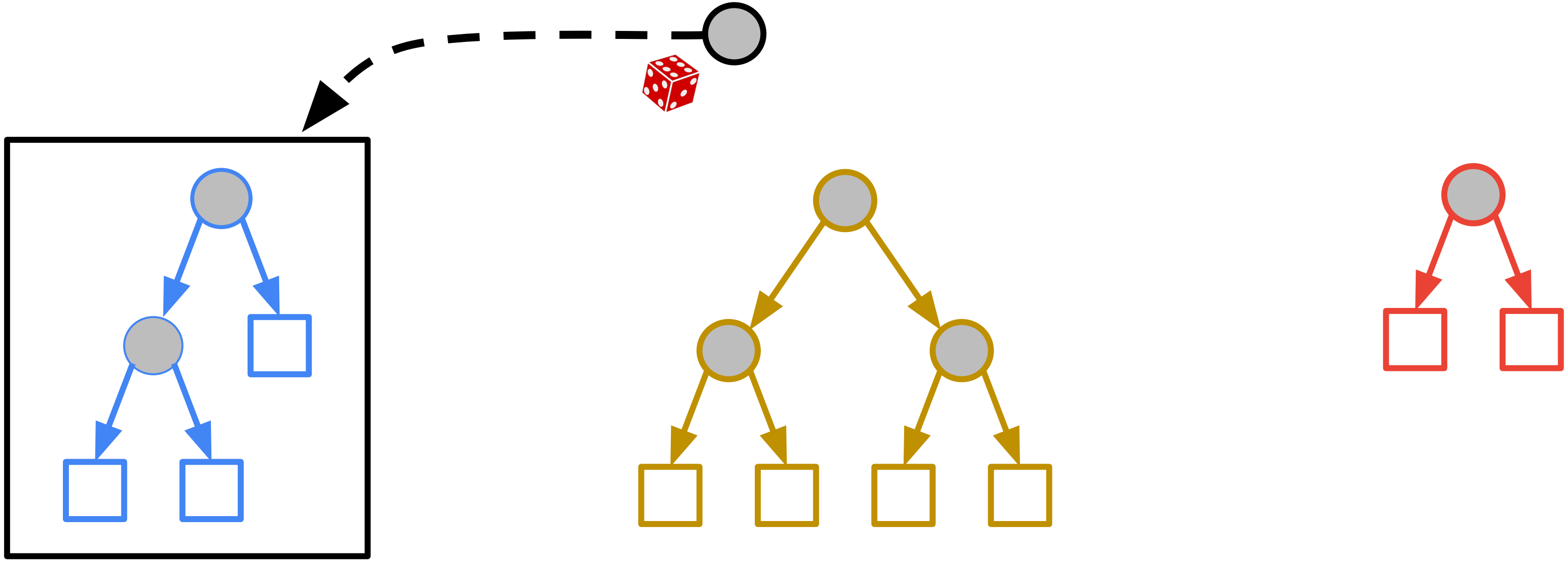
Set of trees, each leaf associated to a "*complex distribution*"

Google Research

Nock & Guillame-Bert, "Generative Forests", NeurIPS'24

ARF – generation

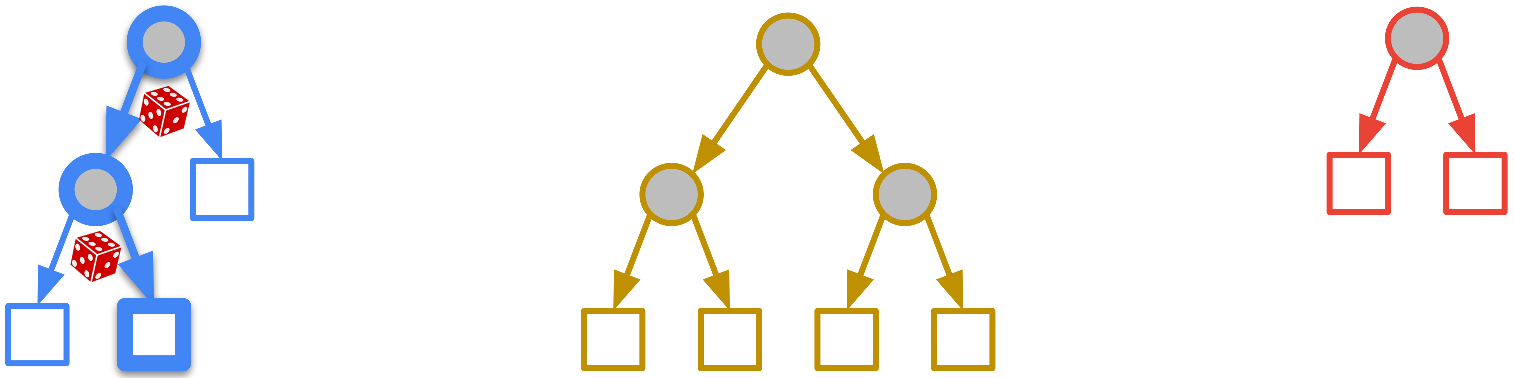
ARF – generation, **Step 1: pick a tree**



Google Research

Nock & Guillame-Bert, "Generative Forests", NeurIPS'24

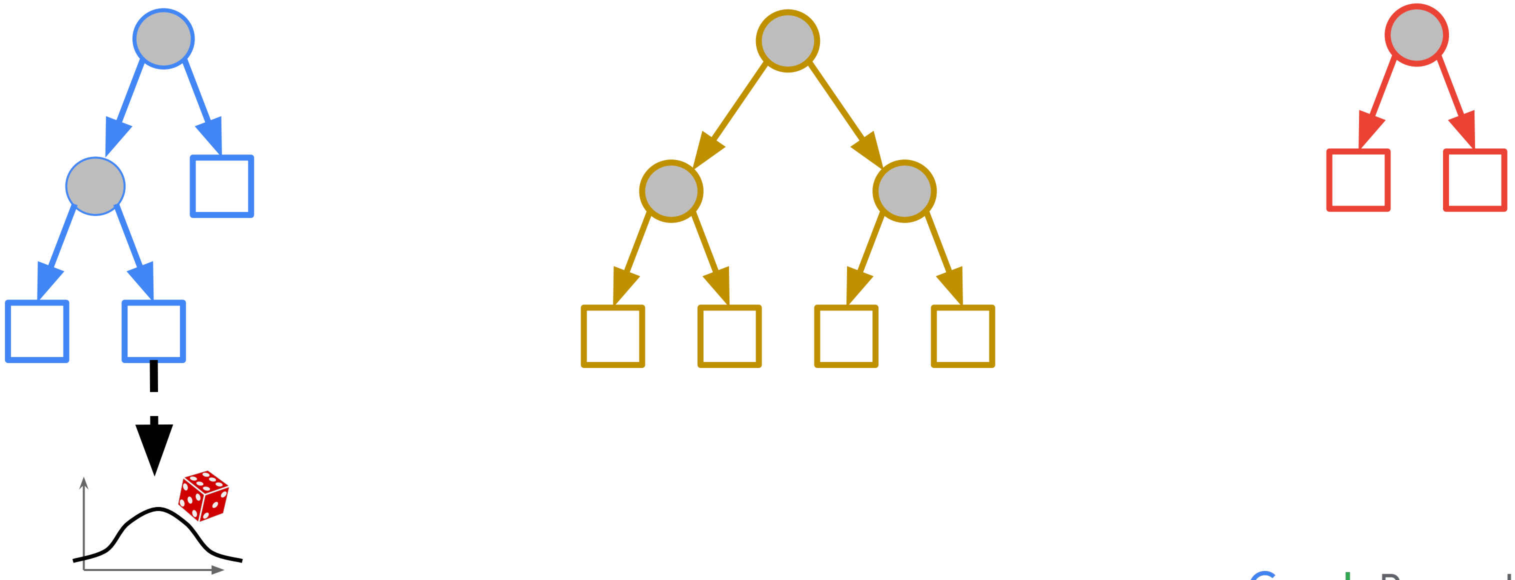
ARF – generation, Step 2: stochastic activation of arcs



Google Research

Nock & Guillame-Bert, “Generative Forests”, NeurIPS’24

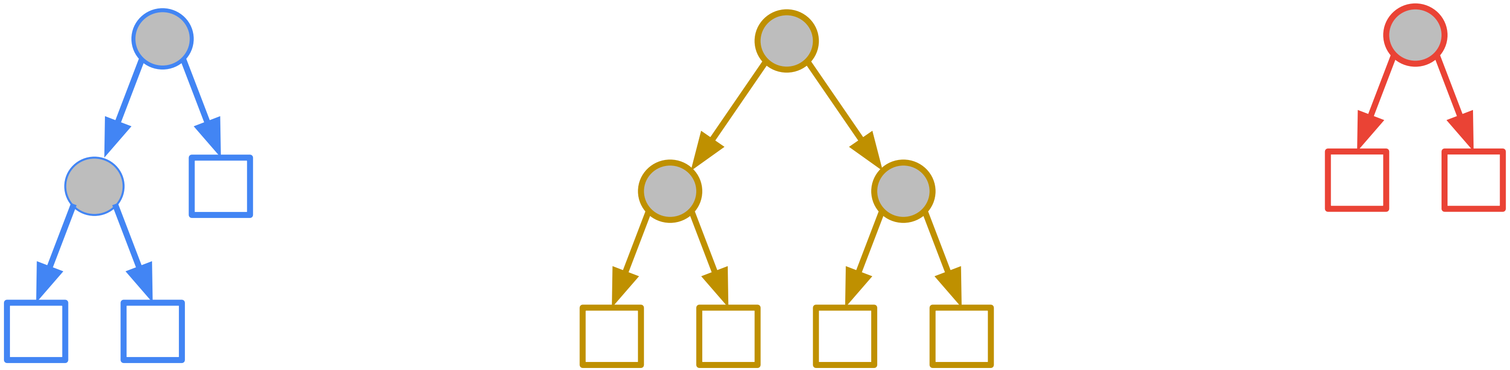
ARF – generation, Step 3: sample at the chosen leaf



Google Research

Nock & Guillame-Bert, "Generative Forests", NeurIPS'24

ARF : good models have big trees



Because of Step 1 which picks 1 tree (and then samples from it), each tree needs to be a good model *separately* (⇒ "big" trees)

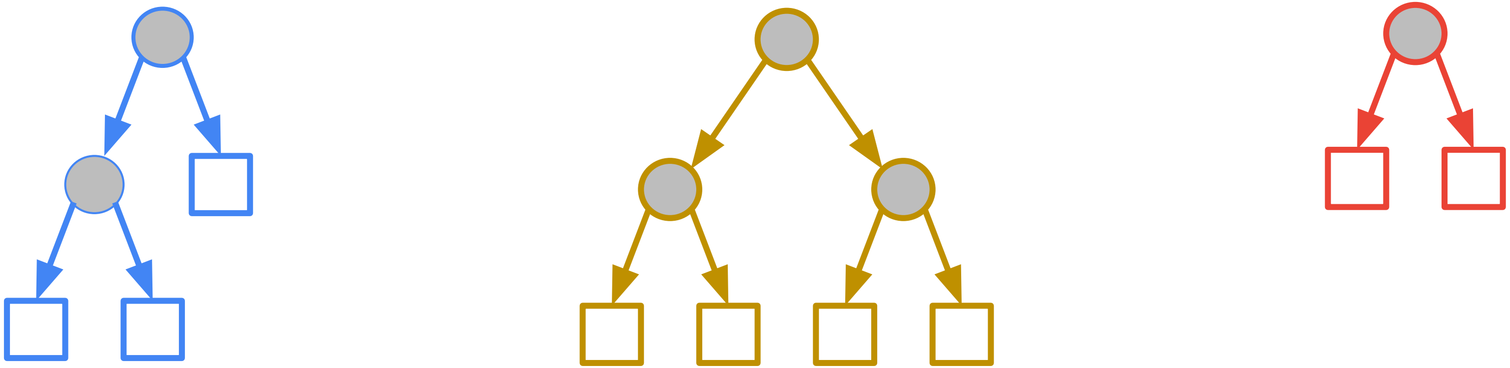
Google Research

Nock & Guillame-Bert, "Generative Forests", NeurIPS'24

Generative Forests

Google Research

Generative Forest – model



+ empirical measure

Set of trees

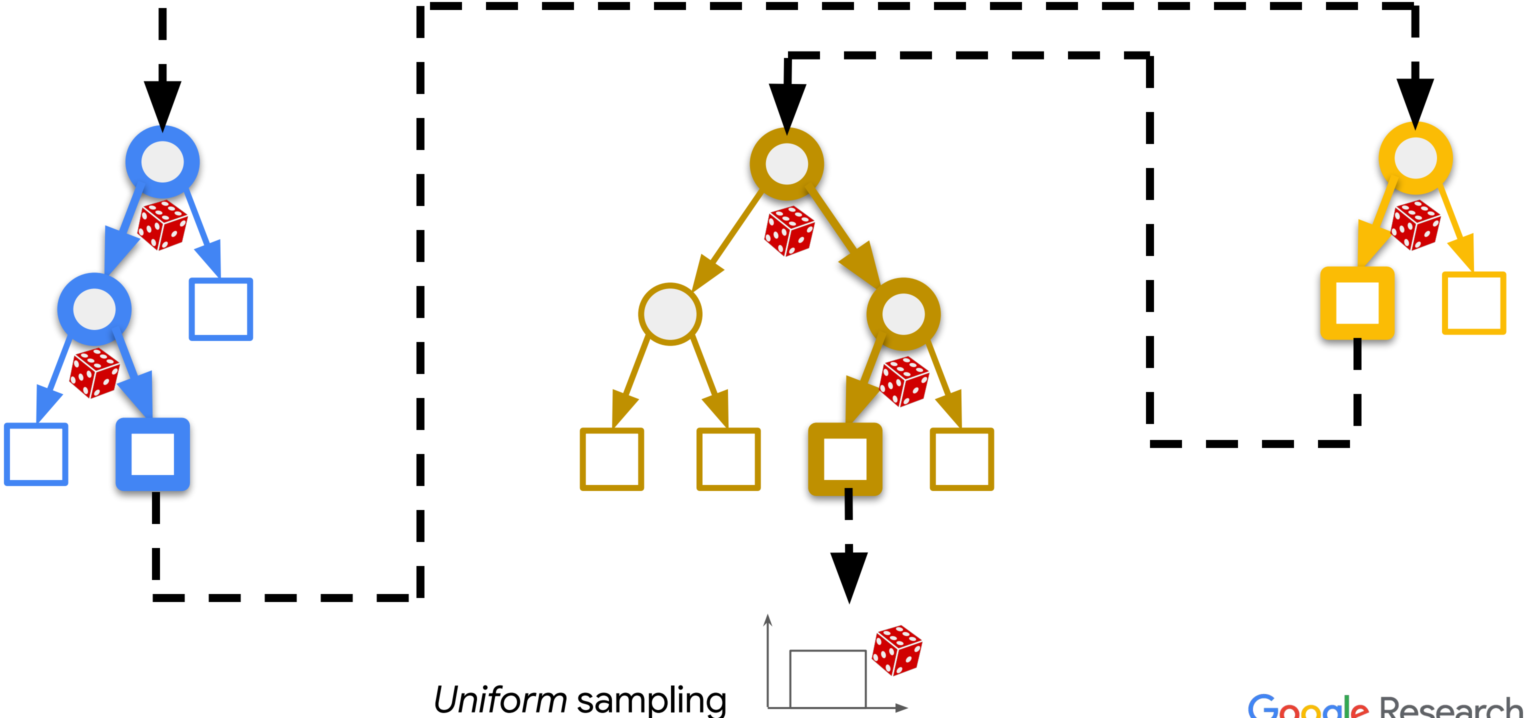
Google Research

Nock & Guillame-Bert, "Generative Forests", NeurIPS'24

Generative Forest – generation

Google Research

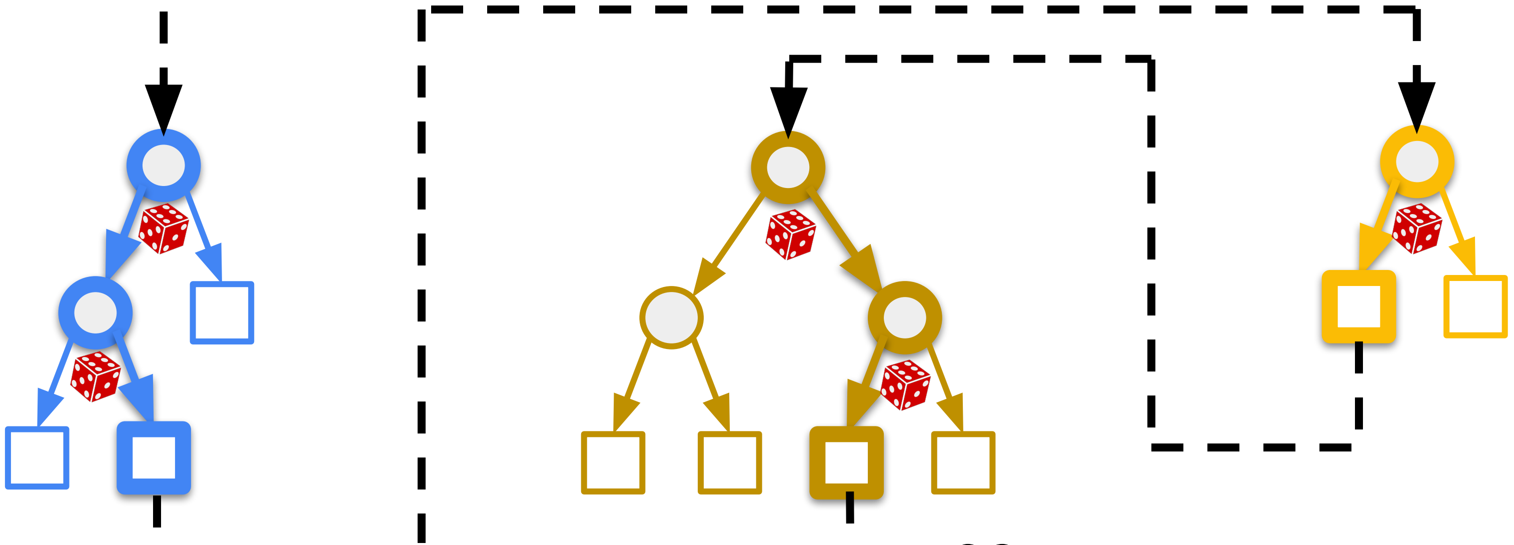
Generative Forest – generation uses **all** trees and R



Google Research

Nock & Guillame-Bert, "Generative Forests", NeurIPS'24

Generative Forest – generation uses **all** trees and R



Using all trees gives a model that #bins space $\propto$ product of trees' number of leaves, so *small* models can be *very* accurate

Google Research

Nock & Guillame-Bert, "Generative Forests", NeurIPS'24

Training: boosting using decision tree (DT) induction !

- Optimize a density ratio loss to fit B to A using a Bregman divergence

$$\mathbb{D}_\ell\left(\mathrm{A}, \mathrm{B}\right) \doteq \pi \cdot \mathbb{E}_{\mathrm{U}}\left[D_\varphi\left(\frac{\mathrm{dA}}{\mathrm{dU}} \middle\| \frac{\mathrm{dB}}{\mathrm{dU}}\right)\right]$$

π = user-fixed prior, U = uniform distribution
see paper for generator φ

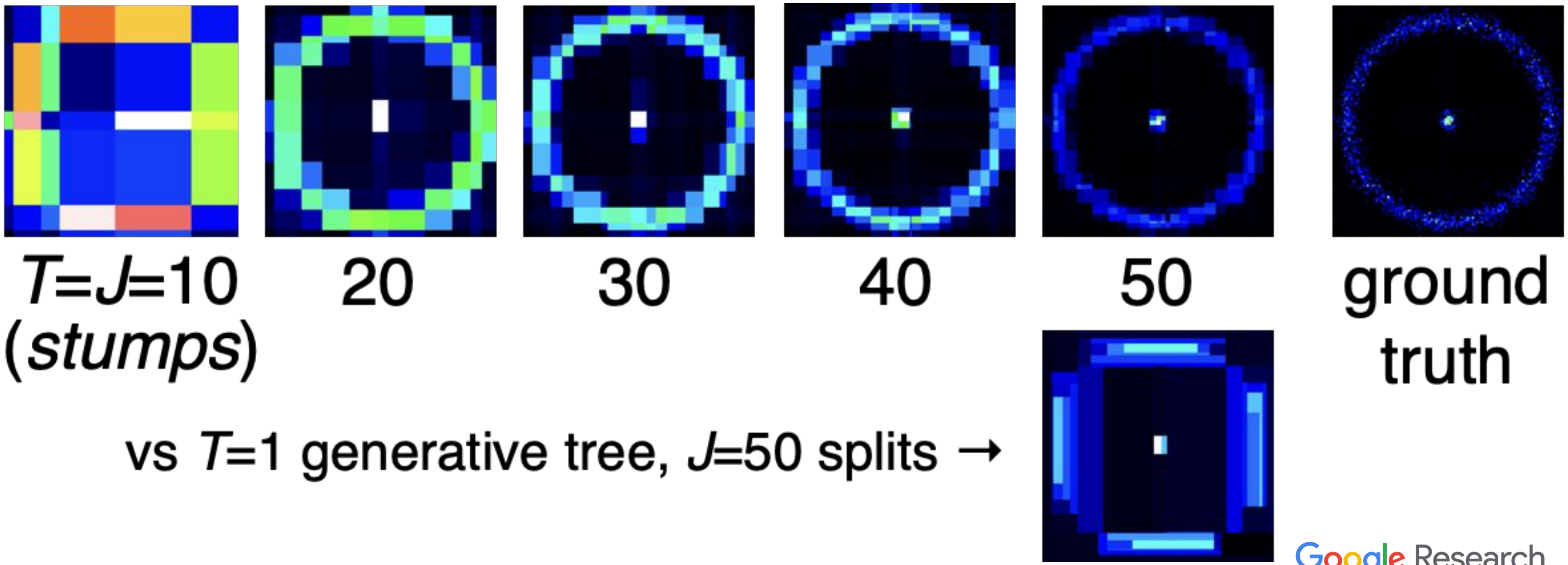
- Learn a GF G to fit empirical R
- Trick: "recycle" 2-class DT induction to distinguish positive = R vs negative = U and with prior π (=P[Y=1]) – same training at the core as e.g. CART, C4.5, etc. !
- Rate: using a weak learning assumption, get at iteration *J* with *T*<*J* trees GF G.,

$$\mathbb{D}_\ell\left(\mathrm{R}, \mathrm{G}_J\right) \leqslant \mathbb{D}_\ell\left(\mathrm{R}, \mathrm{G}_0\right) - \frac{\kappa\gamma^2\kappa^2}{8} \cdot T \log\left(1 + \frac{J}{T}\right)$$

ch

Experiment 1: “power” of GFs vs single trees

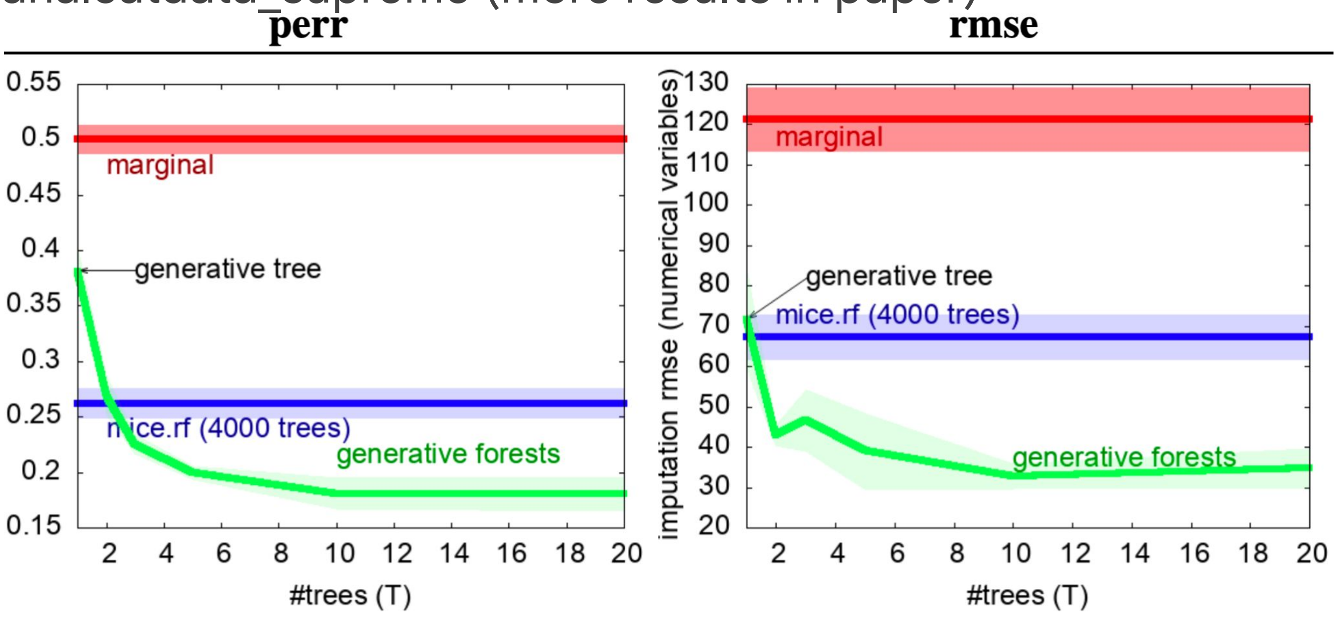
- Small GFs with just stumps can approximate non-boxy / complex distributions



Nock & Guillame-Bert, “Generative Forests”, NeurIPS’24

Experiment 2: missing data imputation

- Comparison vs MICE with random forests (4 000 total #trees !) on UCI analcatdata_supreme (more results in paper)



perr : metric for categorical variables
RMSE : metric for numerical variables

Google Research

Nock & Guillame-Bert, "Generative Forests", NeurIPS'24

Main experiment: quality of generated data (summary)

- Contenders of different types: CT-GAN, Vine copulas AE, Forest Flow (FF), ARFs
- Four metrics: optimal transport ↓, coverage ↑, density ↑ and F1 measure ↓
- Our models' size:
  - "**Medium**": T = 500 trees, total #splits J = 2 000 (average #splits/tree = 4)
  - "**Small**": T = 200 trees, total #splits = 500
- Summary for **Medium**: substantially better than NN based methods (CT-GAN, VCAE) & ARF on all metrics; better than FF. For **Small**: same picture vs NNs, still better than ARF on 3 metrics, on par with FF except density
- Compute / complexity: FF & ARF model sizes huge compared to ours; NNs required "big" desktop (our models = low-end laptop)

Google Research

Thank You

Richard Nock Google Research
Mathieu Guillame-Bert Google