Regularization in Random Forests

Bruna Wundervald [1*], Andrew Parnell [2], Katarina Domijan [3]

[1,2,3] Hamilton Institute, Maynooth University, Maynooth, Ireland

*Corresponding author, Email: brunadaviesw@gmail.com

1 Introduction

- Predictors can frequently be hard or economically expensive to obtain
- For tree-based methods, there is not yet a well established regularization procedure in the literature

2 Random Forests

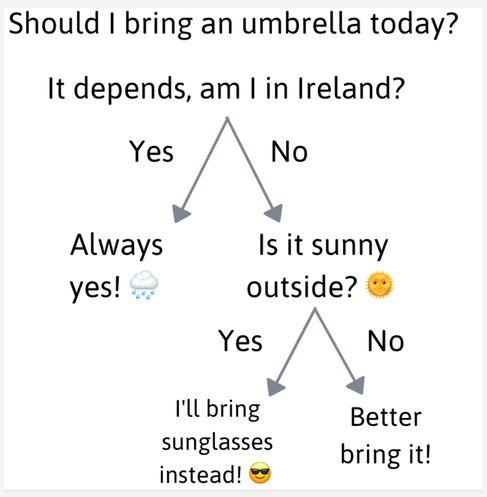

Figure 1: **Example of a decision tree.**

Considering a continuous variable of interest $Y_i \in \mathbb{R}$ and $\mathbf{x} = (x_{i1}, \ldots, x_{ip})'$ the set of predictor features, $i = 1 \ldots n$:

- It is an average of B trees grown in Bootstrap samples,
- Simple way to reduce variance in tree models:
  - build a separate prediction for each dataset
  - average their final predictions, resulting in

$$\hat{f}_{avg}(\mathbf{x}) = \frac{1}{B}\sum_{b=1}^{B}\hat{f}^{(b)}(\mathbf{x}), \qquad (1)$$

[1].

Variable importance: improvement in the splitting criteria (RSS) for each variable

- facilitates feature selection,
- struggles to separate features in the presence of highly correlated variables

3 Regularization in Random Forests

One option is presented in [2]

- The authors **penalise the gain (RSS reduction)** of each variable,
- The main idea is to weigh down the gains, with

$$Gain_R(X_i, \nu) = \begin{cases} \lambda_i Gain(X_i, \nu), i \notin F \text{ and} \\ Gain(X_i, \nu), i \in F, \end{cases}$$

where F represents the set of indices used in the previous nodes and $\lambda_i \in (0, 1]$ is the penalization applied to the splitting.

The variables will only get picked if their gain is very high.

3.1 Optimal values for λ

- Key-point for the method to work well
- Can depend on characteristics of each variables – e.g. marginal correlation to the response

Our extension: make it depend on the depth ν of the tree, with

$$\lambda_i = \lambda(\nu).$$

4 Implementation

The current implementation is being done as an extension of the `ranger` [3] package for `R`. The code is available at `https://github.com/brunaw/ranger`.

5 Results for simulated data

We simulated 10 datasets with the following relationship between response and predictors:

$$y_i = 0.1 sin(\pi x_{i1} x_{i2}) + 2(x_{i3} - 0.5)^2$$
$$+ 1x_{i4} + 0.5x_{i5} + \sum_{j=6}^{205} 0.9^{(j-5)/3} x_{ij} + \epsilon_i,$$
$$\epsilon_i \overset{iid}{\sim} N(0, \sigma^2),\ X_p \sim U[0, 1], p = 1, \ldots, 205.$$

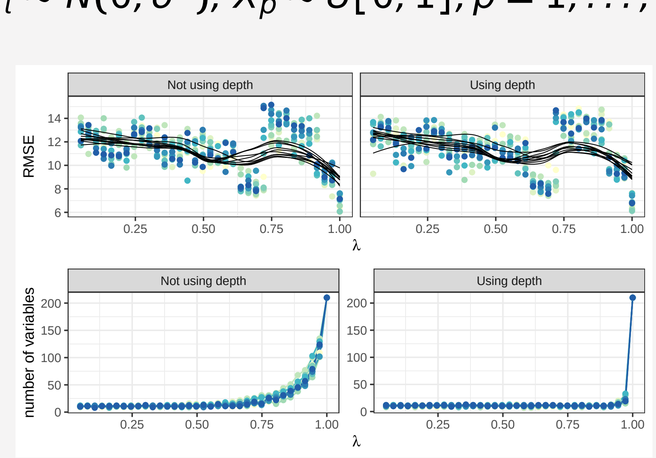

Figure 2: **Number of selected variables and RMSE for each dataset, varying λ, considering and not considering the depth of the trees.**

The number of selected variables depends on the number of tested variables (`mtry` at each iteration).

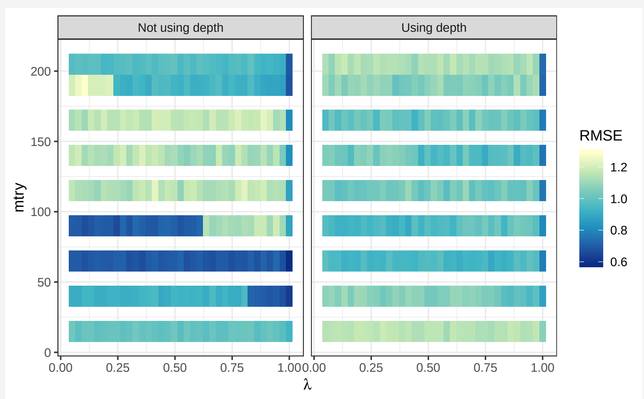

Figure 3: **Tile plot varying the number of variables tested at each iteration and λ, showing the mean RMSE for the 10 datasets, considering and not considering the depth of the trees, using 50 trees.**

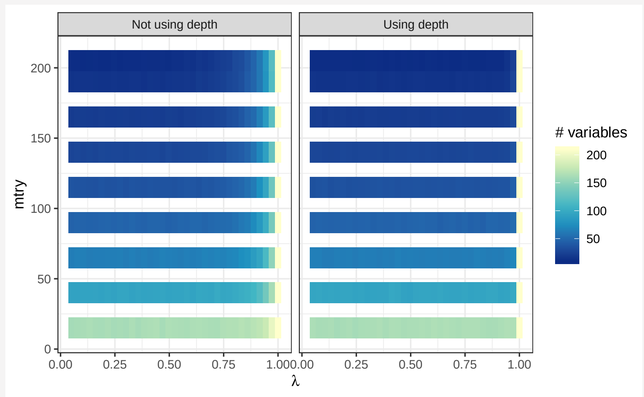

Figure 4: **Tile plot varying the number of variables tested at each iteration and λ, showing the mean number of seleceted variables for the 10 datasets, considering and not considering the depth of the trees, using 50 trees.**

6 Discussion

- Regularized random forests can achieve better or equal prediction power than the full model, but using **far fewer** variables
- The optimal model does not depend only on the regularization parameter

References

[1] Leo Breiman. "Random Forests". In: *Machine Learning* (2001). ISSN: 1098-6596. DOI: 10.1017/CBO9781107415324.004.

[2] Houtao Deng and George C. Runger. "Gene selection with guided regularized random forest". In: *CoRR* abs/1209.6425 (2012). eprint: 1209.6425.

[3] Marvin N. Wright and Andreas Ziegler. "ranger: A Fast Implementation of Random Forests for High Dimensional Data in C++ and R". In: *Journal of Statistical Software*

77.1 (2017), pp. 1–17. DOI: 10.18637/
jss.v077.i01.