**Introduction**

In the field of machine learning, the relationship between accuracy and interpretability is often seen as a trade-off. An ensemble of trees, like for example, a random forest, often provides highly accurate results but the final predictive model is too complex for human understanding (James et al., 2013). A high-accuracy, low-interpretability model like this will from now on be referred to as a *black box* model.

As the usage of black box models has become more commonplace it is increasingly important to understand how these models make their predictions (Bodó, & Janssen, 2021; Varshney & Alemzadeh, 2017). As pointed out by Rudin (2019): “the lack of transparency and accountability of predictive models can have (and has already had) severe consequences”. In the USA, for example, people have been incorrectly denied parole, or incorrectly released on bail (Wexler, 2017). Recently, the Dutch government came under scrutiny in an event called “Dutch childcare benefits scandal” (Dutch: “Toeslagenaffaire”), where, between 2013 and 2019, an estimated 26,000 parents of Bulgarian descent were wrongly accused of making fraudulent benefit claims, driving many families into severe (financial) hardships (van Bruxvoort & van Keulen, 2021; Huisman, 2020). The European Digital Rights (EDRi) organization, a network of more than 40 European civil organizations, advocate for more regulation and transparency on artificial intelligence decision-making systems in the European Union, especially in the area of law enforcement and criminal justice.

While highly accurate black box models are very useful, the call for models with higher interpretability is valid and cannot be ignored. In practice, post-hoc explanation techniques that attempt to interpret black box models are often used. Local Interpretable Model-agnostic Explanations (LIME; Ribeiro, et al., 2016) and Shapley additive explanations (SHAP; Lundberg & Lee, 2017) are popular examples. Kaur et al. (2020) however, point out that while these tools are useful in uncovering issues with datasets or models, they are often over-trust and misused for interpretation. Rudin (2019) argues that instead of trying to explain black models, the way forward is to create *glass box* models that are not only accurate, but also inherently interpretable. She disregards the trade-off between accuracy and interpretability as a myth. She shows that in numerous domains, highly interpretable glass box models exist that have accuracy close to, equal, or higher than black box models. Furthermore, she notes that:

Generally, in the practice of data science, the small difference in performance between machine learning algorithms can be overwhelmed by the ability to interpret results and process the data better at the next iteration. In those cases, the accuracy/interpretability tradeoff is reversed – more interpretability leads to better overall accuracy, not worse.

That said, high accuracy is of course still needed. Perfectly interpreting a model that is not accurate would not be a fruitful enterprise. In the current thesis, I will attempt to improve the accuracy of the Generalized Linear Mixed-Model (GLMM) tree model. As explained below, this glass box model is easily interpretable, as it results in a single decision tree. It can also produce more accurate results as compared to decision trees created using the CART mechanism (Fokkema et al., 2018; Hajjem et al., 2017; Sela & Simonoff, 2012). GLMM trees however, fall short on the accuracy of black box models. Using the *born-again tree* algorithm, combined with a Bayesian tree ensemble method, I will attempt to improve the accuracy of the GLMM tree to be comparable to that of a black box model, while maintaining the high interpretability.

**Born-Again Trees**

In 1996, Breiman and Shang introduced the Born-Again (BA) tree algorithm. The BA approach allows the user to create a single tree model, which has an accuracy that is close to the black box model, but is much easier to interpret and can be used by humans to make predictions using only an image of the tree and the predictor variables occurring in the tree.

A BA algorithm follows these four steps:

1. A black box model is fitted on the original predictor variables (**X**).
2. Based on **X**,a number of observations are artificially generated (**X­gen**).
3. The black box model is used on **X­gen** to create predictions for the generated data (**Ygen**).
4. A single tree is fitted on **Ygen** to create a BA tree.

Researchers studying BA trees can thus focus on three aspects:

* The black box model (steps 1 and 3)
* The data generation method (step 2)
* The single BA tree (step 4)

Below, I will describe how these aspects are studied in the current thesis.

***The Black box model: BART***

Previous studies focusing on black box models in the context of BA trees, found accurate results with boosted tree ensembles (Breiman & Shang, 1996; …), support vector machines (BRON BRON), and neural networks (Craven & Shavlik, 1995; …). In the current study, I will focus only on tree ensembles. In previous research, Classification And Regression Trees based ensembles (CART) were used. In the current study however, I will use Bayesian Additive Regression Trees based ensembles (BART).

BART models are created by first creating K trees (usually 200) with a single root node. The mean of this node is sampled from a prior, which is a normal distribution where 95% of the means lie between y­min and ymax (when using shrinkage parameter *k* = 2). The model then calculates the residuals and goes into the second iteration, where the trees (number of splits, or means) are randomly permuted based on the priors. A tree can grow extra nodes of prune off nodes based on a beta distribution prior. Usually, = 0.95 and = 2 is set so that most trees end up having size 2 or 3 (55% and 28% respectively). The number of iterations is usually set to 1000. Every iteration, trees that improve the fit are favored. The end result is not a single per observation but a distribution of predicted values for every iteration. This distribution is called the Posterior Probability Distribution (PPD). The first 200 iterations are usually very inaccurate and are thus removed. These are called the burn-in samples. For a more detailed explanation of the BART model, read Chipman et al. (2010).

All parameters can be optimized by cross-validation, but this is computationally very demanding. As the “standard” parameters are usually effective, it is recommended to use them instead (Chipman et al., 2010; Sparapani et al., 2021). Using the standard parameters is not only computationally advantageous, but is a more data-centric approach that is less likely to lead to overfitting (Carnegie, 2020). BART based ensembles have been shown to outperform Boosting, Neural Networks, Random Forests, and a host of other machine learning algorithms in terms of predictive accuracy (Chipman et al., 2010; Dorie et al., 2019).

Though the BART model assumes independence among observations, it can be altered to allow for multilevel analysis (Sparapani et al., 2021; Tan, 2018; Wundervald et al., 2022). In the current thesis I will be utilizing this fact, as the BA tree will be a multilevel tree as well. The datasets that I use have a multilevel structure too. Most importantly for this thesis, BART models result in a PPD for every observation, which will be used in the data generation method.

***The Data Generation Method***

*Smearing*

In order for the BA tree to have a high accuracy, it needs a large sample size to build the tree (Breiman & Shang 1996). Breiman and Shang therefore suggest manufacturing data to increase the sample size. In their study, they apply *smearing* on the original set of predictor variables **X**, to create a new dataset **X­gen.** The smearing algorithm repeats the following steps:

1. A row *i* is randomly selected from **X**(*i, j*), with [*i* = 1, … , *N*].
2. A random number *r* is selected for every predictor *j*, with [*j* = 1, … , *P*].
3. If *r* > *palt*, **X­gen**(*i, j*) = **X**(*i, j*). Else, a random value of *j* in [*j* = 1, … , *P*] is selected for **Xgen**(*i, j*) = **X**(*i, jrandom*).
4. Using the black box model defined earlier, **Xgen**(*i, j*) is used to create **Ygen**(*i, j*)

Where *N* is the number of participants, *P* is the number of predictors, and *palt* is a threshold number that is set beforehand between 0 and 1. Note that for *palt* = 0, we are performing permutation and when *palt* = 1, we are sampling random participants as is. *palt* thus represents the degree of permutation. *palt* can be optimized for more accurate results. Breiman and Shang (1996) use *palt* = 0.25 and *palt* = 0.50 in their study. In the current thesis, I will compare the effectivity of *palt* = 0, *palt* = 0.25, and *palt* = 0.50, with the use of a new data generation method: the PPD method.

*PPD sampling*

There has not been significant improvement to smearing as the data generation method, but the use of permutation may lead to unrealistic data patterns and has been criticized (Hooker et al., 2021). In the current study, I will sample the **Y­gen** from the PPD acquired when using BART ensembles.Data generate in this manner should more closely resemble **X** and thus lead to a more accurate BA tree. As stated before, this is possible when using a BART ensemble as a black box.

*Size of* ***Ygen*** *(****­****n\_Ygen)*

Breiman and Shang (1996) state that a larger size of **Ygen** could lead to more accurate trees. In their own preliminary experiments, they generated data that was a factor 10 to 20 times the original training set (i.e., *n\_Ygen ­*= 10 to *n\_Ygen ­*= 20). They found trees that were more accurate, but so large that they were not interpretable. For their main experiment, they used *n\_Ygen ­*= 1. They further stated that their results were not sensitive to the value of *n\_Ygen.* In the current thesis,I will attempt to replicate these results by varying *n\_Ygen* between 1, 5, and 10, and studying the result on interpretability and accuracy. For computational purposes, I weighted **Ygen** when fitting the BA Trees equally with wi = 1 / *n\_Ygen*.

***The BA Tree***

Breiman and Shang (1996) used the CART algorithm to create the BA Tree. While a single CART is useful for decision making, as it functions as a flow-chart, the accuracy of the tree is usually low (Fernández-Delgado et al., 2014; Gacto et al., 2019; Zhang et al., 2017). When multiple trees are combined to form ensembles, as in the BART, Random-Forest, Bagging, or Boosting algorithms, the accuracy is greatly improved at the cost of dramatically decreasing the interpretability. There have however, been attempts to improve the single-tree algorithm without using the BA Tree approach (BRONNEN). Namely the recently developed Generalized Linear Mixed-Model (GLMM) Tree shows promise in providing accurate and interpretable results.

As stated earlier, the BART based ensembles allow for incorporation of mixed-effects structures. To take advantage of that, I will use a Generalized Linear Mixed-Model (GLMM) tree model as the single BA tree. GLMM trees account for correlated structures in decision-tree analyses, and have been shown to yield more accurate, as well as less complex trees than gained from the CART algorithm (e.g., Fokkema et al., 2018; Hajjem et al., 2017; Sela & Simonoff, 2012).

***Research Questions and Hypotheses***

The main goal of the current paper is to study how BART-based BA GLMM trees perform to “regular” GLMM trees. As stated before, I prioritize interpretability over accuracy when comparing these models, but accuracy should still be compared. The first two research questions are thus:

**RQ1:** Does a BART-based BA approach improve the interpretability of GLMM trees?

**RQ2:** Does a BART-based BA approach improve the predictive accuracy of GLMM trees?

Interpretability being measured in number of nodes and accuracy being measured in MSE (explained further below), we get the following working hypotheses:

**H01:** On average, BART-based BA GLMM trees have equal tree size as GLMM trees on the same datasets. (Tree SizeBAbart = Tree SizeBAGLMM).

**H02:** On average, BART-based BA GLMM trees have equal MSE as GLMM trees on the same datasets (MSEBAbart = MSEGLMM).

I will also compare BART-based BA GLMM trees to smearing-based BA GLMM trees, as proposed by Breiman and Shang (1996). This leads us to the third research question:

**RQ3:** In terms of interpretability and predictive accuracy, does a BART-based BA approach outperform a smearing-based approach?

As before, number of nodes and MSE are used to measure interpretability and accuracy respectively. This gives us the third and fourth hypotheses:

**H03:** On average, BART-based BA GLMM trees have equal tree size as smearing-based BA GLMM trees on the same datasets. (Tree SizeBAbart = Tree SizeBAsmear).

**H04:** On average, BART-based BA GLMM trees equal lower MSE as smearing-based BA GLMM trees on the same datasets (MSEBAbart = MSEBAsmear).

Finally, I will study two variables whose effect on interpretability and accuracy, although briefly discussed by Breiman and Shang (1996), is not yet clear. These variables are N­gen, and *palt*. Breiman and Shang (1996) mention N­gen should be large, but not how large or what effects its’ size has. The same can be said for *palt*. It is only mentioned that the authors used *palt* = 0.25 and *palt* = 0.50. The final research question is thus:

**RQ4** In terms of interpretability and predictive accuracy, what effect does varying Ngen and *palt* in BA trees have?

Ngen will be varied between [1, 5, 10] times the original dataset, and *palt* will be varied between [0.00, 0.25, 0.50]. This gives us three BART-based BA GLMM tree models: [**M**BAbart.1, **M**BAbart.2, **M**BAbart.3], and different smearing-based GLMM tree models: [**M**BAsmear.1, **M**BAsmear.2, …, **M**BAsmear.9]. As **RQ4** is exploratory in nature, our final hypotheses are:

**H05:** For every value of Ngen and *palt*: (NodesBAbart.1 = NodesBAbart.2 = NodesBAbart.3) & (NodesBAsmear.1, NodesBAsmear.2, …, NodesBAsmear.9).

**H06:** For every value of Ngen and *palt*: (MSEBAbart.1 = MSEBAbart.2 = MSEBAbart.3) & (MSEBAsmear.1, MSEBAsmear.2, …, MSEBAsmear.9).

**Methods**

***Procedures***

First, samples of N = 1000 (or 999, see *Datasets* section above) were taken from each dataset. The data was then randomly split in ten equal folds to use in 10-fold Cross Validation (10-CV). When a fold was assigned as test set, it was checked whether all factor levels in the test set were also present in the training set. If not, the data was moved to the training set. This decreases the randomness of the data, but was necessary as leaving the data resulted in errors.

Next, all analyses were run and the MSE and number of Nodes was saved. First, a GLMM tree was made, then a multilevel BART model was calculated and the PPD was saved. Using this PPD, three new training sets were created using PPD sampling described above. These were then used to calculate three BART-based GLMM tree models. Finally, nine training sets were created using the smearing algorithm described above which were used to calculate nine smearing-based GLMM tree models.

The running of the analyses was repeated for all 10 train/test splits. The 10-CV was repeated 10 times, resulting in 10 × 10 = 100 different values of MSE and number of Nodes for every model on every sample.

***Comparisons***

The models were compared on interpretability and accuracy. These are measured in Tree Size, and MSE respectively.

*Tree Size*

Tree models are often described as interpretable, yet inaccurate models. Overfitting however, can lead to very large, complex trees that are accurate, yet impossible to interpret fully for humans. The size of trees is thus a natural measure for interpretability defined as the number of Nodes divided by two minus one, which will be used to compare the interpretability between tree models. The interpretability of the BART model will not be compared, as being a black box model, it is too complex to be interpreted by a human. **(SOURCE)**

*MSE*

To compare the accuracy of the models, I will use the test MSE. To compare the MSE of different datasets to each other, I will calculate the relative MSE for every split. To compare MSE for different datasets, I will calculate the relative MSE, which is defined by the MSE divided by the minimum MSE for each test/train split (Chipman et. Al, 2010).

***Datasets***

The hypotheses were tested on four different benchmark datasets described below. The datasets all had a multilevel structure. Because of computational limitations, random samples of N = 1000 (when possible) were taken for analyses. For time-series data, random patients were sampled instead of random observations, so that every level had enough data.

*AIDS Clinical Trials Group Study 175 (ACT)*

This dataset was created by Hammer et al. (1996) measures human immunodeficiency virus type 1 (HIV-1) infected patients and measures the amount of CD4 T cells present in the blood as outcome measure. A number of drugs, demographics and confounding variables are present to predict CD4 count. Multiple measurements were taken over time within patients, creating a multilevel structure. There are 6417 observations from 2139 patients 24 variables. A sample of N = 1000 was taken, containing 377 patients.

*Safety*

This dataset made publicly available by Hox et al. (2017). A sample of 100 streets are selected, and on each street a random sample of 10 persons are asked how often they feel unsafe while walking that street. The question about feeling unsafe is asked using three answer categories: 1 = never, 2 = sometimes, 3 = often. Predictor variables are age and gender; street characteristics are an economic index (standardized Z-score) and a rating of the crowdedness of the street (7-point scale). The multilevel structure comes from the person that are nested in the streets. No sample was taken as the original data has N = 1000 already.

*Marriage*

This is a simplified version of a dataset created by Lax and Phillips (2009) where support for gay marriage in the USA is predicted based on age, education, gender, Christianity and political vote. 6525 Participants are nested within the 49 states they live in. A sample of N = 1000 containing all 49 states was taken.

*Early Childhood Longitudinal Study Kindergarten Class of 2010-11 (ECLS-K*)

Mulligan et al. (2016) collected data about kindergarteners over the course of five years. The data includes information on selected child and family characteristics, such as poverty status, parental education, family type, and primary home language. This is used to predict the children's knowledge and skills in math, reading and science, in separate datasets. As multiple measurements are taken within the children, we get a multilevel structure. For the math and reading datasets, samples of N = 1000 containing 200 children were taken. For the science dataset a sample of N = 999 containing 333 children were taken, as the science test was only measured at three ages.

We also transformed the dataset to include school wide statistics. We calculated the percentage of males, the mean Social Economic Score, the mean race, and mean certification per school. This new data is called ECLSK-School was used to predict the reading score of the individual children, nested in schools. From this dataset a sample of N = 999 containing 146 schools was taken.

***Software***

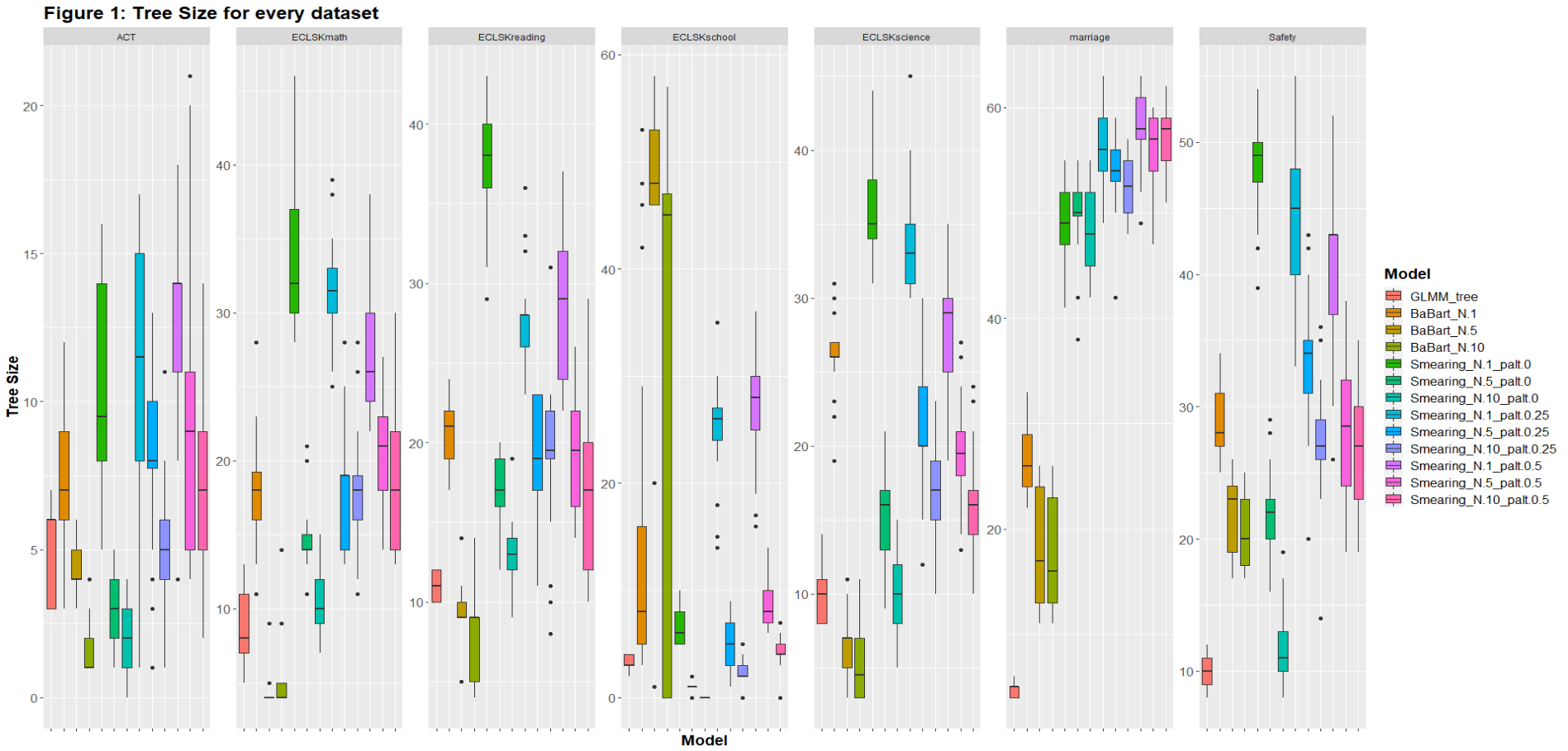
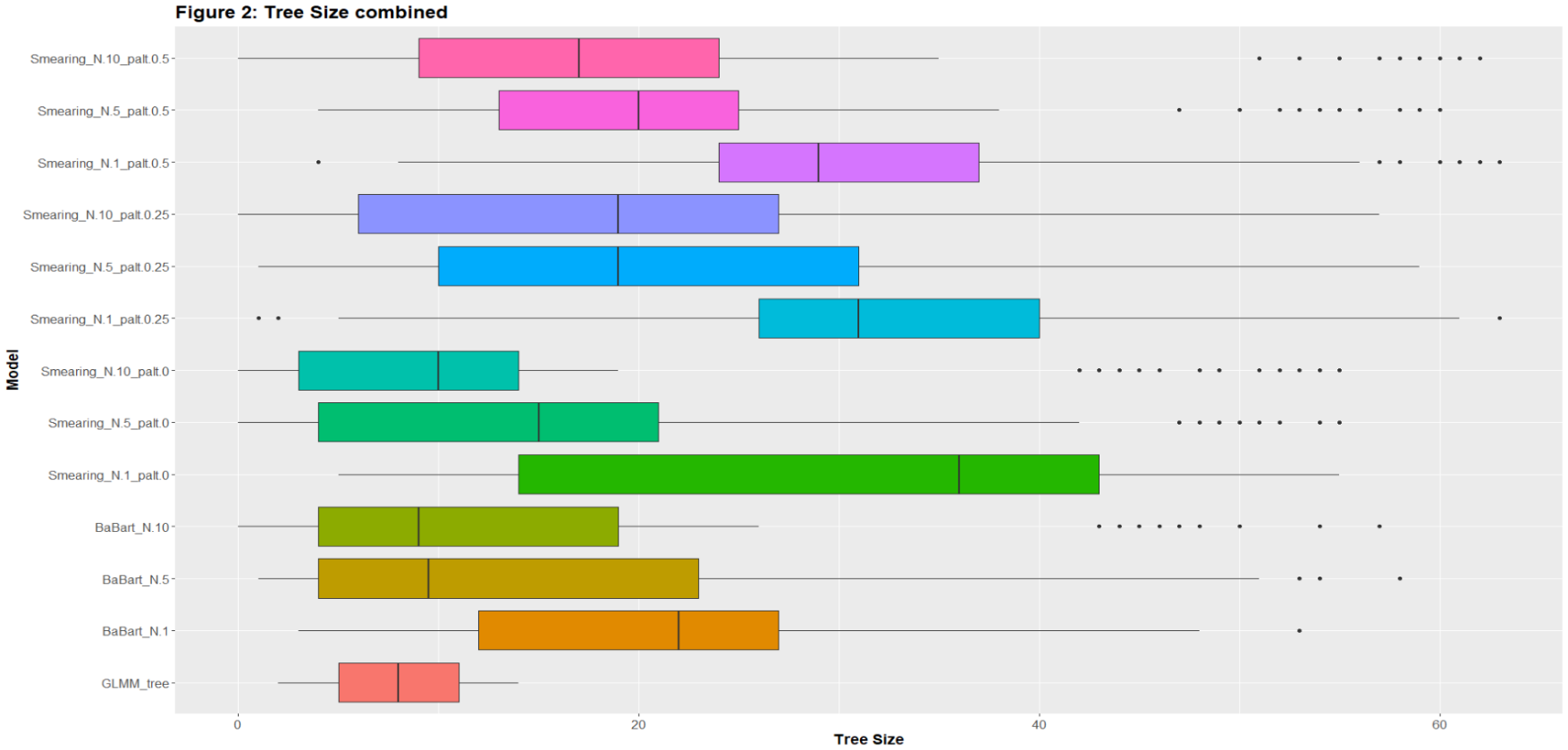
All analyses were performed using R Statistical Software (v4.1.1; R Core Team, 2021). The BART models were calculated using the rbart\_vi function in the dbarts package (v0.9-20; Dorie, 2021). The GLMM trees were made using the lmertree function in the glmertree package (v0.2-0; Fokkema et al., 2018).

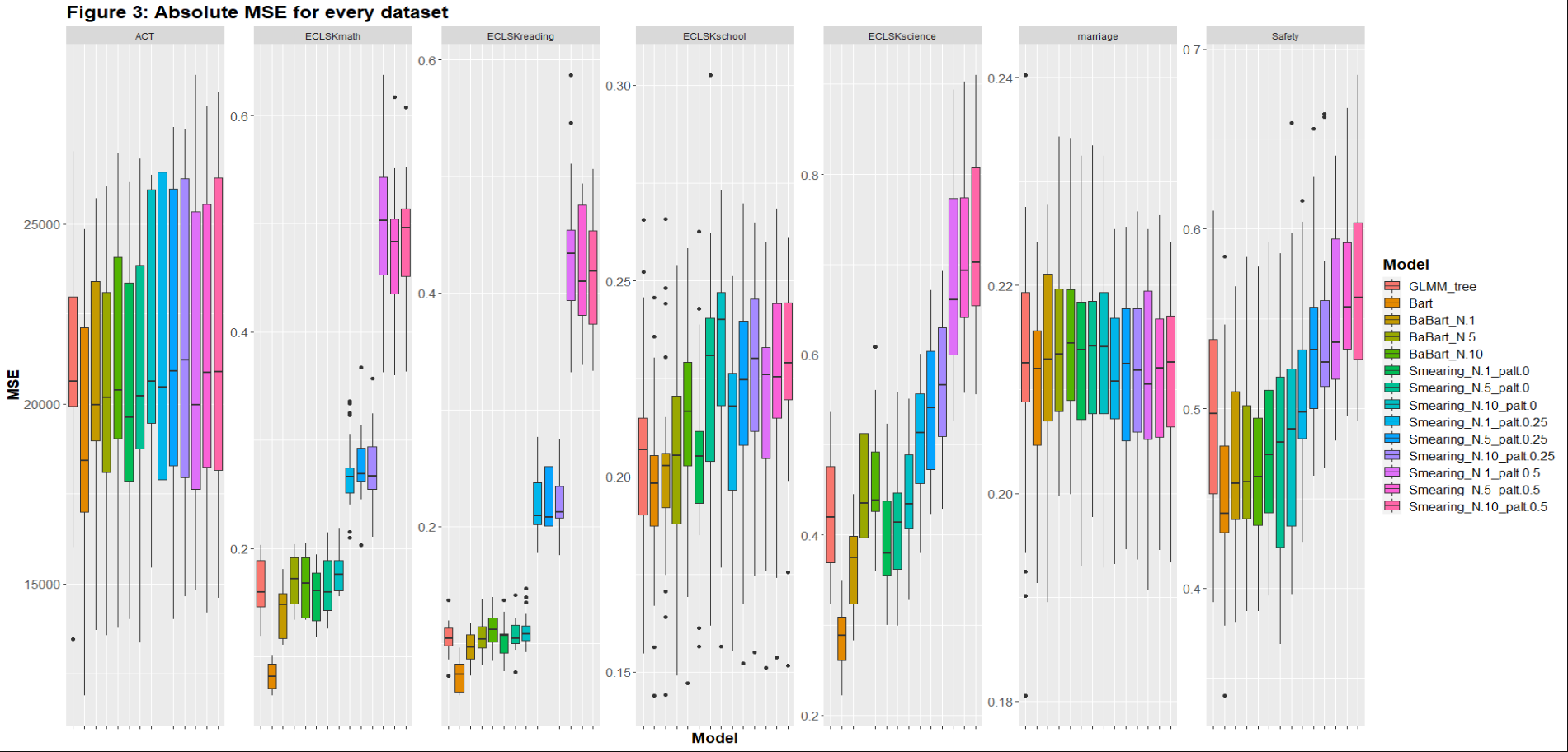
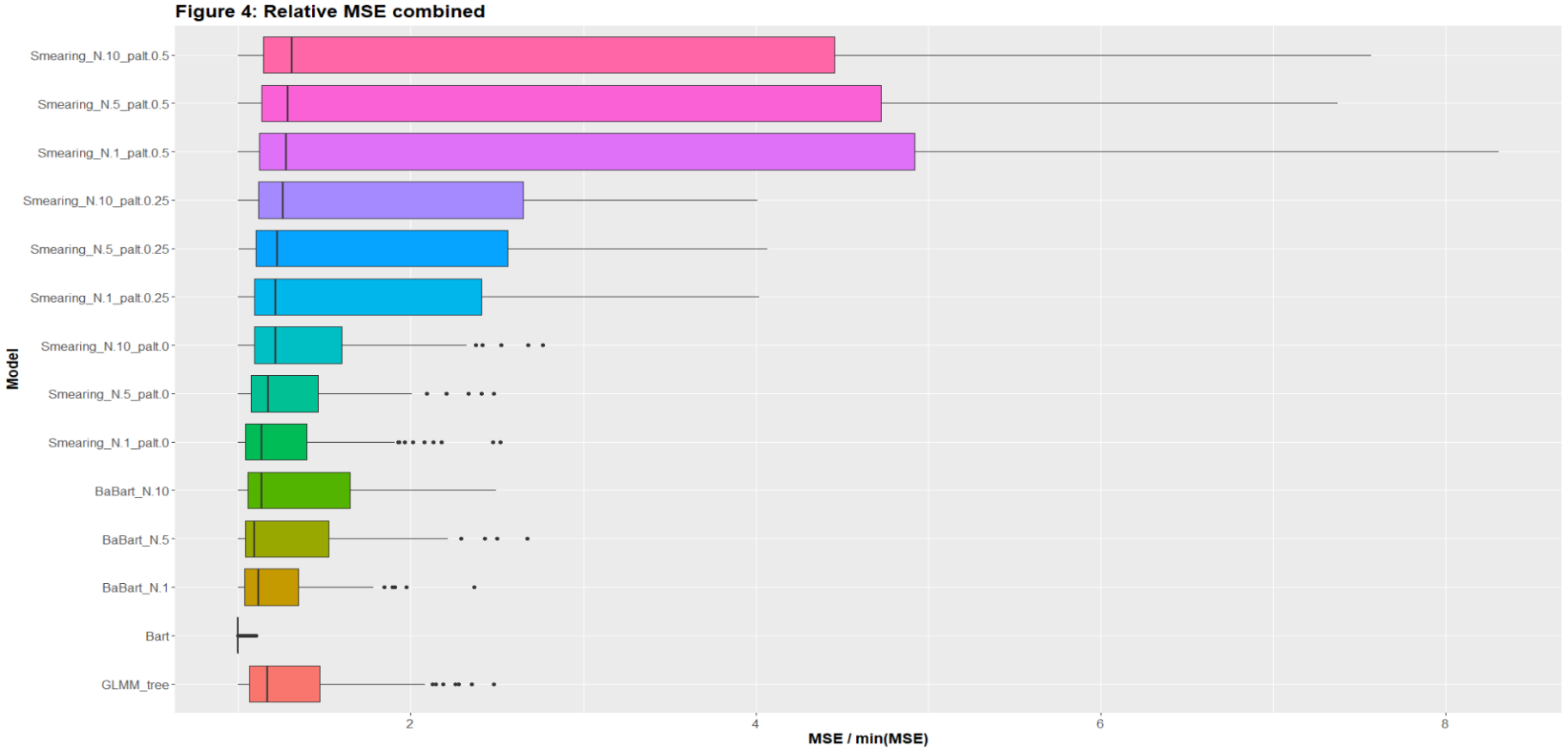
**Results**

The tree size of each model is presented in Figure 1. In the born-again models, it is generally observed that the tree size decreases as Ngen increases. BART-based BA GLMM trees have smaller tree size than regular GLMM trees on the ACT, Math, Reading, and Science datasets. However, on the Safety, Marriage, and ECLSK-school datasets, the tree size of BART-based BA GLMM trees is larger.

As shown in Figure 2, regular GLMM trees, BART-based BA GLMM trees with Ngen = 10, and Smearing-based BA GLMM trees with Ngen = 10 and palt = 0 have the lowest tree size on average, with regular GLMM trees having the lowest median.

The absolute mean squared error (MSE) of each model is depicted in Figure 3. BART models consistently have the lowest MSE. Additionally, BART-based BA GLMM trees tend to have lower MSE than regular GLMM trees. Smearing-based BA GLMM trees typically have the highest MSE, except on the ACT and marriage datasets where the MSE is similar across all models. In the other datasets, increasing palt also increases the MSE.

Figure 4 shows that on average, Ngen = 1 results in the lowest MSE. As palt increases, the MSE tends to also increase. These results suggest that optimizing Ngen and palt may be important for achieving the lowest MSE in these models.



**Discussion**

The results provide mixed support for **H01**, which proposed that BART-based BA GLMM trees would have equivalent tree sizes to GLMM trees on the same datasets. As shown in Figure 1, BART-based GLMM trees did indeed have smaller tree sizes than GLMM trees for the ACT, math, reading, and science datasets. However, BART-based GLMM trees had larger tree sizes than GLMM trees for the school, marriage, and safety datasets.

There is not sufficient evidence to reject **H02**, which proposed that BART-based BA GLMM trees would have equivalent MSE to GLMM trees on the same datasets. As shown in Figures 3 and 4, the MSE of BART-based BA GLMM trees was slightly lower than the MSE of GLMM trees, but the difference was not statistically significant.

There is also not enough evidence to reject **H03** as stated, which proposed that BART-based BA GLMM trees would have equivalent tree sizes to smearing-based BA GLMM trees on the same datasets. However, Figures 1 and 2 suggest that low values of Ngen and high values of palt would have led to rejection of H3, as BART-based BA GLMM trees had smaller tree sizes than smearing-based BA GLMM trees when Ngen ≠ 1 and palt = 0.25 or palt = 0.50. When using Ngen ≠ 1 and palt = 0.00, the smearing-based BA GLMM tree appears to overfit, as it has low MSE but a large tree size. The tree size decreases as Ngen increases, consistent with the findings of Breiman and Shang (1996) that Ngen should be large to prevent overfitting. The effect of palt on tree size is weaker, but palt = 0 results in the smallest tree size. These results lead us to reject **H05**, which proposed that Ngen and palt would have no effect on tree size.

There is not sufficient evidence to reject **H04**, which proposed that BART-based BA GLMM trees would have lower MSE than smearing-based BA GLMM trees on the same datasets. As shown in Figures 3 and 4, when using palt = .00, the MSE of BART-based BA GLMM trees was equivalent to that of smearing-based BA GLMM trees. However, when using palt = .25 or palt = .50, the MSE increased significantly. This finding is in contrast to the results of Breiman and Shang (1996), who found that using palt = .25 or palt = .50 resulted in the lowest MSE. It is worth noting that their study fitted BA CART trees rather than BA GLMM trees, which may account for the discrepancy. The MSE remained roughly equal when varying Ngen. However, the drastic effect of palt on the MSE of smearing-based BA GLMM trees leads us to reject **H06**.

**Limitations**

*Relative MSE*

As stated before, I used relative MSE to compare the MSE obtained in different datasets. Relative MSE, as defined by Chipman et. al (2010) to be MSE/min(MSE), makes the assumption that for every split, every possible value of MSE is known.

*Variance*

As can be seen in Figure 1, there is high variance in the obtained tree sizes. This is partly due to the fact that on the datasets ECLSKschool and ECSLKmath BA GLMM trees are created that have zero nodes. This happens because the GLMM tree makes its prediction using only the multilevel variance. This could skew the results in Figure 2, making it look like some BA GLMM trees have lower tree size on average than can actually be expected.

**References**

Bodó, B., & Janssen, H. (2021). Here Be Dragons–Maintaining Trust in the Technologized Public Sector. *Amsterdam Law School Research Paper*, (2021-23).

Breiman, L., & Shang, N. (1996). Born again trees. *University of California, Berkeley, Berkeley, CA, Technical Report*, *1*(2), 4.

van Bruxvoort, X., & van Keulen, M. (2021). Framework for Assessing Ethical Aspects of Algorithms and Their Encompassing Socio-Technical System. *Applied Sciences*, *11*(23), 11187.

Carnegie, N. B., (2020, May 13). *Introduction to Bayesian Additive Regression Trees for Causal Inference* [Targeted Learning Webinar Series]. Putnam Data Sciences. <https://www.youtube.com/watch?v=9d5-3_7u5a4&t=2093s>

Chipman, H. A., George, E. I., & McCulloch, R. E. (2010). BART: Bayesian additive regression trees. *The Annals of Applied Statistics*, *4*(1), 266-298.

**Christodoulou, E., Ma, J., Collins, G. S., Steyerberg, E. W., Verbakel, J. Y., & Van Calster, B. (2019). A systematic review shows no performance benefit of machine learning over logistic regression for clinical prediction models. *Journal of clinical epidemiology*, *110*, 12-22.**

Craven, M., & Shavlik, J. (1995). Extracting tree-structured representations of trained networks. *Advances in neural information processing systems*, *8*.

Dorie, V., Hill, J., Shalit, U., Scott, M., & Cervone, D. (2019). Automated versus do-it-yourself methods for causal inference: Lessons learned from a data analysis competition. *Statistical Science*, *34*(1), 43-68.

Fair Trials and European Digital Rights (EDRi), (2022, March 1). *Civil society calls on the EU to ban predictive AI systems in policing and criminal justice in the AI Act*. Retrieved from <https://edri.org/our-work/civil-society-calls-on-the-eu-to-ban-predictive-ai-systems-in-policing-and-criminal-justice-in-the-ai-act/>

Fernández-Delgado, M., Cernadas, E., Barro, S., & Amorim, D. (2014). Do we need hundreds of classifiers to solve real world classification problems?. *The journal of machine learning research*, *15*(1), 3133-3181.

Fokkema, M. (2017). Fitting prediction rule ensembles with R package pre. *arXiv preprint arXiv:1707.07149*.

Fokkema, M., Edbrooke-Childs, J., & Wolpert, M. (2021). Generalized linear mixed-model (GLMM) trees: A flexible decision-tree method for multilevel and longitudinal data. *Psychotherapy Research*, *31*(3), 329-341.

Fokkema, M., Smits, N., Zeileis, A., Hothorn, T., & Kelderman, H. (2018). Detecting treatment-subgroup interactions in clustered data with generalized linear mixed-effects model trees. *Behavior research methods*, *50*(5), 2016-2034.

Gacto, M. J., Soto-Hidalgo, J. M., Alcalá-Fdez, J., & Alcalá, R. (2019). Experimental study on 164 algorithms available in software tools for solving standard non-linear regression problems. *IEEE Access*, *7*, 108916-108939.

Hammer, S. M., Katzenstein, D. A., Hughes, M. D., Gundacker, H., Schooley, R. T., Haubrich, R. H., ... & Merigan, T. C. (1996). A trial comparing nucleoside monotherapy with combination therapy in HIV-infected adults with CD4 cell counts from 200 to 500 per cubic millimeter. *New England Journal of Medicine*, *335*(15), 1081-1090.

Hooker, G., Mentch, L., & Zhou, S. (2021). Unrestricted permutation forces extrapolation: variable importance requires at least one more model, or there is no free variable importance. *Statistics and Computing*, *31*(6), 1-16.

Hox, J. J., Moerbeek, M., & Van de Schoot, R. (2017). *Multilevel analysis: Techniques and applications*. Routledge.

Huisman, P. (2020). Hoe de toeslagenaffaire kon gebeuren. *Management Kinderopvang*, *26*(2), 36-37.

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An introduction to statistical learning* (Vol. 112, p. 18). New York: springer.

Kaur, H., Nori, H., Jenkins, S., Caruana, R., Wallach, H., & Wortman Vaughan, J. (2020, April). Interpreting interpretability: understanding data scientists' use of interpretability tools for machine learning. In *Proceedings of the 2020 CHI conference on human factors in computing systems* (pp. 1-14).

Lax, J. R., & Phillips, J. H. (2009). How should we estimate public opinion in the States?. *American Journal of Political Science*, *53*(1), 107-121.

Lundberg, S. M., & Lee, S. I. (2017). A unified approach to interpreting model predictions. *Advances in neural information processing systems*, *30*.

Kaur, H., Nori, H., Jenkins, S., Caruana, R., Wallach, H., & Wortman Vaughan, J. (2020, April). Interpreting interpretability: understanding data scientists' use of interpretability tools for machine learning. In *Proceedings of the 2020 CHI conference on human factors in computing systems* (pp. 1-14).

**Klievink, A. J. (2021). *Hollen én stilstaan: hoe data en digitalisering de overheid veranderen* (Doctoral dissertation, Leiden University).**

Mulligan, G. M., McCarroll, J. C., Flanagan, K. D., & Potter, D. (2016). Findings from the Third-Grade Round of the Early Childhood Longitudinal Study, Kindergarten Class of 2010-11 (ECLS-K: 2011): First Look. NCES 2016-094. *National Center for Education Statistics*.

R Core Team (2021). R: A language and environment for statistical computing*. R Foundation for Statistical Computing,* Vienna, Austria. URL https://www.R-project.org/.

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016, August). " Why should i trust you?" Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 1135-1144).

Rudin, C. (2019). Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, *1*(5), 206-215.

**Semenova, L., Rudin, C., & Parr, R. (2019). A study in Rashomon curves and volumes: A new perspective on generalization and model simplicity in machine learning. *arXiv preprint arXiv:1908.01755*.**

Sparapani, R., Spanbauer, C., & McCulloch, R. (2021). Nonparametric machine learning and efficient computation with bayesian additive regression trees: the BART R package. *Journal of Statistical Software*, *97*, 1-66.

Tan, Y. V. (2018). *Novel Applications and Extensions for Bayesian Additive Regression Trees (BART) in Prediction, Imputation, and Causal Inference* (Doctoral dissertation).

Varshney, K. R., & Alemzadeh, H. (2017). On the safety of machine learning: Cyber-physical systems, decision sciences, and data products. *Big data*, *5*(3), 246-255.

Vincent Dorie (2021). dbarts: Discrete Bayesian Additive Regression Trees Sampler. R package version 0.9-20. URL https://CRAN.R-project.org/package=dbarts

Wexler, R. (2017). When a computer program keeps you in jail. *New York Times*.

Wundervald, B., Parnell, A., & Domijan, K. (2022). Hierarchical Embedded Bayesian Additive Regression Trees. *arXiv preprint arXiv:2204.07207*.

**Notes**

**Vragen Marjolein:**

* Wat voor statistieken kan ik het beste gebruiken om de hypotheses te testen?

**Voordelen:**

BART can overcome assumptions about depth of trees and shrinkage because prior incourages small trees certain level of shrinkage towards 0, but it is not a fixed value compared to boosted trees where you can set max depth to f.e. 3 splits.

Computational benefits from avoiding CV -> more data informed approach

Embeds what is normally an algorithmic approach in a likelihood framework to produce coherent uncertainty intervals, unusual for machine learning approaches. (?)

**BART and causal inference: Why?**

More thorough control for confounding than with traditional parametric models. Normally there are assumptions about the confounders.

Causal effects are easily shown (?)