When are Deep Networks better than Decision Forests at small sample sizes, and how?

Adway Kanhere*, Noga Mudrik*, Haoyin Xu, Joshua Vogelstein
Biomedical Engineering, Johns Hopkins University

Abstract

- Forests have empirically dominated tabular data scenarios, where the relative position of features is irrelevant.
- In contrast, networks typically dominate other methods on large sample size structured data scenarios, where the relative position of features is key for sample identification.
- The relationship between the internal representations that the two approaches learn has not yet been made explicit, to our knowledge
- We illustrate the conceptual commonalities of their representations on three different classification tasks.

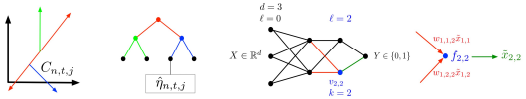

Fig 1: Representations of forests (left) and networks (right)

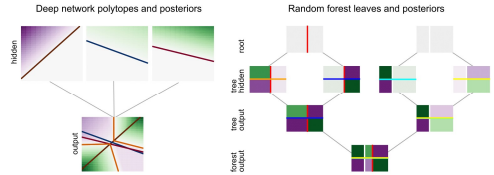

Fig 2: Visualizations of the polytope compositions for networks (left) and forests (right)

Vision Tasks

- We experimented with multi-class classifications on the CIFAR 10 & CIFAR 100 datasets using 1-layer, 2-layer, 5-layer CNNs, pretrained ResNet-18, and Random forests.
- The 3-class and 8-class training sets are from the CIFAR-10 and the 90- class training sets from the CIFAR-100 dataset
- Final metrics reported were Cohen's Kappa and Expected calibration error.

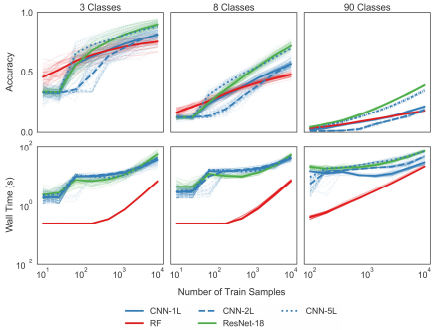

Fig 3: Performance of forests and networks on multiclass CIFAR 10 classifications. The left two columns use CIFAR-10, while the rightmost uses CIFAR-100

Audition Tasks

- We performed benchmarks on the FSD Kaggle 18K dataset using the same models as Vision tasks.
- Various sample sizes and training sample combinations were selected during model training.
- To preprocess the auditory files for networks, we used the short-time Fourier transform to convert the 8 kHz raw auditory signals into mel-spectrograms
- The final metrics reported were Cohen's Kappa and Expected calibration error (ECE)

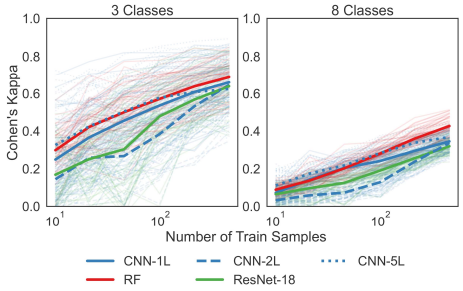

Fig 4: Performance of forests and networks on multiclass FSDKaggle18 classifications using mel-spectrograms

Tuning hyper-parameters using Bayesian Optimization

- With increasing problem complexity, tuning parameters of algorithms is time-consuming and usually employs too many evaluations.
- Factorial based optimization such as grid search or global optimization techniques such as DIRECT or genetic algorithms become impractical.
- Bayesian optimization allows a smarter way to tune parameters by building a smooth surrogate model of the objective function.
- We perform hyper-parameter tuning for vision and audition for maximizing accuracy during each trial.
- After tuning, the best parameter sets were used to retrain the models from scratch and classify on the test data.

Parameters	Range
Learning rate	10^{-6} – 0.4
Momentum	0 – 1
Epochs	15 – 40
Optimizer	SGD , Adam

Fig 5: Table of parameters and their ranges that have been used for Bayesian hyper-parameter optimization

Tabular Tasks

- Three models were tested for the tabular data: Random-forest (by xgboost), GBDT (by xgboost), and TabNet [5], which is a high-performance and interpretable canonical deep tabular data learning architecture.
- After the hyper-parameters of all models were tuned on a validation set, the models were trained and then tested on a held-out test set.
- This evaluation was performed on different samples sizes of a large number of real-world datasets, each with unique properties and features. For each model, the Expected calibration error (ECE) as well as Cohen's Kappa were calculated. The training time was measured as well.
- As for the current results, while the training time of the TabNet network was significantly higher than that of the other models, its performance, with respect to both ECE and Cohen's Kappa, was better as well. For all models – the generalization ability has increased as the number of samples increased.

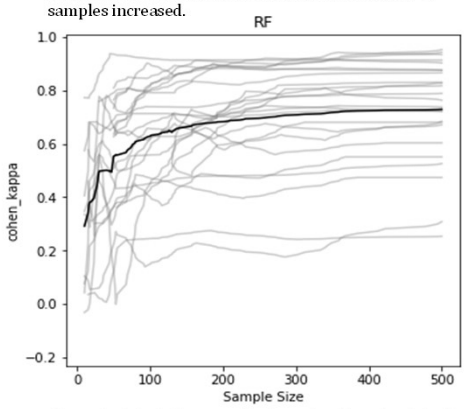

Figure 6: Cohen's Kappa results for random forest. Each bright curve is the result on one data set, samples across several sample size. The black curve is the average performance

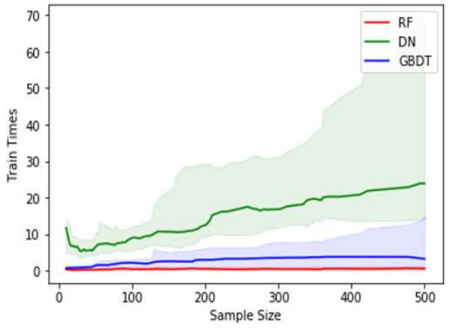

Figure 7: Training times with confidence intervals for the 3 tabular data models. The training time for TabNet (DN) was much higher than that of the other models, especially in large sample size

Conclusions

- In this study we compared the performance of three types of models, random forests, gradient boosting, and deep neural networks, on three types of tasks—vision, audition, and tabular.
- For the vision tasks, the performance decreased for all models as the number of classes increased. In general, the accuracy of all models improved as the number of training samples increased. The wall time of random forest was much lower than the corresponding times of the other checked models.
- For the audition tasks, the performance of different models highly depended on the number of training samples, and the variance of the Cohen's Kappa metric, over different dataset, was high for each of the models. For 3 classes, RF and CNN-2L reached similar performance and outperformed the other models for all sizes of training data. However, in the case of 8 classes, the CNN-5L outperformed the other models for small training sample sizes and RF outperformed the other models for higher training sample sizes.
- For the tabular data, the training times of the DN were much higher than the training time of GBDT and RF. For all models, as the number of training sample sizes increases, both the ECE and Cohen's Kappa metric increases.
- We hope that our results will lead to a better understanding of the mechanisms according to which a model excels on one type of data and not on the other. This way, we will be able to use different models in a more efficient and effective manner, thus improving performance and interpretability.

Code Paper

References

1. Jerome H. Friedman. Greedy function approximation: A gradient boosting machine. The Annals of Statistics, 29(5):1189–1232, 2001.
2. Rich Caruana and Alexandru Niculescu-Mizil. An empirical comparison of supervised learning algorithms. In Proceedings of the 23rd International Conference on Machine Learning, ICML '06, pages 161–168, New York, NY, USA, 2006. ACM.
3. Rich Caruana, Nikos Karampatziakis, and Ainur Yessenalina. An empirical evaluation of super-vised learning in high dimensions. In Proceedings of the 25th international conference on Machine learning, pages 96–103, New York, New York, USA, July 2008. ACM.
4. Manuel Fernández-Delgado, Eva Cernadas, Senén Barro, and Dinani Amorim. Do we need hundreds of classifiers to solve real world classification problems? Journal of Machine Learning Research, 15(90):3133–3181, 2014.
5. Arik, S.Ö., & Pfister, T. (2021). TabNet: Attentive Interpretable Tabular Learning. AAAI.

Acknowledgements

The authors thank Dr. Jeremias Sulam for his recommendations and to all the members of the NeuroData lab for their support.