# Perturbation-Enabled Data Augmentation Improves the Generalizability of Classifiers in Network Neuroscience

Gregory Kiar[1], Alan C. Evans[1], Tristan Glatard[2]

**Abstract**

Machine learning models are commonly applied to human brain imaging datasets in an effort to associate function or structure with behaviour, health, or other individual attributes. Such models often rely on low-dimensional maps relating brain regions, generated by complex processing pipelines. However, the numerical instabilities inherent to pipelines limits the fidelity of these estimates, and results in bias-rich derivatives serving as inputs to these machine learning models. This work seeks to take advantage of numerical instabilities in pipelines towards reducing the bias in networks used by machine learning models. We found that resampling brain networks across a series of numerically perturbed outcomes led to more consistently generalizable performance in all tested classifiers, preprocessing strategies, and dimensionality reduction techniques when tasked with an age classification task. Importantly, this finding does not hinge on a large number of perturbed networks in order to exhibit improved performance, suggesting that even minimally perturbing a dataset reduces bias in the resulting models.

**Keywords**

Stability — Network Neuroscience — Neuroimaging — Machine Learning — Generalizability

[1] *Montréal Neurological Institute, McGill University, Montréal, QC, Canada*;

[2] *Department of Computer Science and Software Engineering, Concordia University, Montréal, QC, Canada.*

## Introduction

- machine learning has become commonplace for the identification and characterization of individual biomarkers.
- in neuroimaging, models accept processed imaging data and seek to relate structure or function to phenotypes.
- the development of biomarkers through this approach has been widely applied, such as linking sex, age, bmi, etc., to brain structure.
- however, these models are sensitive the data used for training, and performance out-of-sample often degrades significantly.
- this truth of machine learning is not helped by the fact that bias-rich estimates are used as inputs to the machine learning models, as scientific pipelines at best provide “estimates” of structure or function, rather than capturing true and unobstructed signal.
- the impact that numerical instabilities play in determining the results of pipelines have been clearly demonstrated across operating systems, data perturbations, and pipeline perturbations through MCA
- a benefit of characterizing instabilities through perturbation approaches is that they result in a series of estimates

of derivatives for each sample in a dataset.

- here, we take advantage of a range of possible – and equally plausible – outputs from a MCA experiment on structural connectomes estimation pipeline to augment out dataset.
- we classify brain networks based on participant age, and resample the samples used for this task based on the augmented dataset.
- we evaluate the impact of data augmentation through MCA, explore the relationship between the change in performance and baseline performance for the classification task, and identify any relationship between these potential benefits and the number of MCA simulations performed per sample.

## Materials & Methods

The objective of this study was to evaluate the impact of aggregating collections of unstable brain networks towards learning robust brain-phenotype relationships. We sampled and aggregated simulated networks within individuals to learn relationships between brain connectivity and individual an trait, in this case age, and compared this to traditional baseline performance on this tasks. We compared aggregation strategies with respect to baseline validation performance, performance out-of-sample, and generalizability.

All developed software and analysis resources for this project have been made available through GitHub at `https://github.com/gkpapers/2020AggregateMCA`.

### Dataset

An existing dataset containing Monte Carlo Arithmetic (MCA) perturbed structural human brain networks was used for these experiments[1]. The perturbations introduced for the generation of brain networks in this dataset were at the level of machine-error, simulating expected error over a typical pipeline execution. This dataset contains a single session of data from 100 individuals ($100 \times 1 \times 1$). The derived brain networks were generated with a probabilistic structural connectome estimation pipeline[2] using a fixed random seed, and Monte Carlo Arithmetic (MCA) perturbations were added to all Python-implemented operations throughout the pipeline[3,4]. Each sample was simulated 20 times, resulting in $2,000$ unique graphs. Further information on the processing and curation of this dataset can be found here[5].

This collection enabled the exploration of subsampling and aggregation methods in a typical learning context for neuroimaging[6,7]. Exploring the relationship between the number of simulations and performance further allows for MCA-enabled resampling to be evaluated as a method of dataset augmentation.

As the target for classification, individual-level phenotypic data strongly implicated in brain connectivity was desired. Participant age, which has consistently been shown to have a considerable impact on brain connectivity[8–11], was selected and turned into a binary target by dividing participants into adult ($> 18$) and non-adult groups (68% adult).

### Preprocessing

Prior to being used for this task, the brain networks being were represented as symmetric $83 \times 83$ adjacency matrices, sampled upon the Desikan-Killiany-Tourville[12] anatomical parcellation. To reduce redundancy in the data, all edges belonging to the upper-triangle of these matrices were preserved and vectorized, resulting in a feature vector of $3,486$ edges per sample. All samples were preprocessed using one of four standard techniques:

**Raw** The raw streamline count edge-weight intensities were used as originally calculated.

**Log Transform** The log10 of edge weights were taken, and edges with 0 weight prior to the transform were reset to 0.

**Rank Transform** The edges were ranked based on their intensity, with the largest edge having the maximum value. Ties

were settled by averaging the rank, and all ranks were finally min-max scaled between 0 and 1.

**Z-Score** The edge weights were z-scored to have a mean intensity of 0 and unit variance.

## Machine Learning Pipelines

The preprocessed connectomes were fed into pipelines consisting of two steps: dimensionality reduction and classification. Dimensionality reduction was applied using one of two methods:

**Principal Component Analysis** The connectomes were projected into the 20 dimensions of highest variance. The number of components was chosen to capture approximately 90% of the variance present within the dataset.

**Feature Agglomeration** The number of features in each connectome were reduced by combining edges according to maximum similarity/minimum variance using agglomerative clustering[13]. The number of resulting features was 20, to be consistent with the number of dimensions present after PCA, above.

After dimensionality reduction, samples were fed into one of five distinct classifiers as implemented through scikit learn[14]:

**Support Vector Machine** The model was fit using a radial basis function (RBF) kernel, L2 penalty, and a balanced regularization parameter to account for uneven class membership.

**Logistic Regression** A linear solver was used due to the relatively small dataset size. L2 regularization and balanced class weights were used, as above.

**K-Nearest Neighbour** Class membership was determined using an L2 distance and the nearest 10% of samples, scaling with the number of samples used for training.

**Random Forest** 100 decision trees were fit using balanced class weights, each splitting the dataset according to a maximum of 4 features per node (corresponding to the rounded square root of 20 total features).

**AdaBoost** A maximum of 50 decision trees were fit sequentially such that sample weights were iteratively adjusted to prioritize performance on previously incorrectly-classified samples, consistent with[15].

The hyperparameters for all models were refined from their default values to be appropriate for a small and imbalanced dataset. The performance for all pipeline combinations of preprocessing methods, dimensionality reduction techniques, and models using the reference (i.e. unperturbed) executions in the dataset ranged from an F1 score of 0.64–0.875 with a mean of 0.806; this evaluation was performed on a consistent held-out test set which was used for all experiments, as described in a following section. This set of models was chosen as it includes i) well understood standard techniques, ii) both parametric and non-parametric methods, iii) both ensemble and non-ensemble methods, and iv) models which have been commonly deployed for the classification neuroimaging datasets[8,16–22].

## Dataset Sampling

A chief purpose of this manuscript involves the comparison of various forms of aggregation across equivalently-simulated pipeline outputs. Accordingly, the dataset was resampled prior to dimensionality reduction and classifiers were trained, evaluated, and combined according to the following procedures:

**Reference** Networks generated without any MCA perturbations were selected for input to the models, serving as a benchmark.

**Jackknife** The datasets were repeatedly sampled such that a single randomly chosen observation of each unique network was selected (i.e. derived from the same input datum). This resampling was performed 100 times, resulting in the total number of resamplings being 5× larger than the number of unique observations per network, ensuring a broad and overlapping sampling of the datasets.

**Median** The edgewise median of all observations of the same network were used as the samples for training and evaluation.

**Mean** Similar to the above, the edgewise mean of all observations for each network were computed and used as input data to the classifiers in both collections.

**Consensus** A distance-dependent average network[23] was computed across all observations of each network. This data-aware aggregation method, developed for structural brain network analysis, preserves network properties often distorted when computing mean or median networks.

**Mega-analysis** All observations of each network were used simultaneously for classification, increaseing the effective sample size. Samples were organized such that all observations of the same network only appeared within a single fold for training and evaluation, ensuring double-dipping was avoided.

**Meta-analysis** Individual classifiers trained across jackknife dataset resamplings, above, were treated as independent models and aggregated into an ensemble classifier. The ensemble was fit using a logistic regression classifier across the outputs of the jackknifed classifiers to learn a relationship between the predicted and true class labels.

The robustness and possible benefit of each subsampling approach was measured by evaluation on a subset of all MCA simulations, including 9 distinct numbers of simulations, ranging from 20 to 2 simulations per sample. Combining the dataset sampling methods, the set of simulations, preprocessing strategies, dimensionality reduction techniques, and classifier models, there were $2,200$ models trained and evaluated.

### Training & Evaluation

Prior to training models on the brain networks, 20% of subjects were excluded from each dataset for use as an out-of-sample test dataset for all experiments. With the remaining 80% of subjects, cross validation was performed following a stratified grouped $k$-fold approach ($k = 5$). In this approach, samples were divided into training and validation sets such that the target variable was proportionally represented on each side of the fold (stratified), conditional upon all observations from the same individual, relevant for the mega-analysis dataset sampling method, falling upon the same side of the fold (grouped). This resulted in 5 fold-trained classifiers per configuration, each trained on 64% of the samples and validated on 16%, prior to each being tested on the remaining 20% of held-out samples. All random processes used in determining the splits used the same seed to remove the effect of random sampling.

Classifiers were primarily evaluated on both the validation and test (out-of-sample) sets using F1 score, a standard measure for evaluating classification performance. The generalizability of predictions was defined as:

$$G = 1 - |F1_{test} - F1_{validation}| \tag{1}$$

where a score of 1 (maximum) indicates the equivalent performance across both the validation and test sets, and a lower score (minimum of 0) indicates inconsistent performance. The absolute change in performance was used in Eq. 1, resulting in a score which penalizes spurious over-performance similarly to under-performance, as all inconsistency is undesirable when applying a classifier out-of-sample.

Differences in F1 score and Generalizability were used to measure the change in performance between the reference and other dataset sampling techniques, and statistical comparisons were made through Wilcoxon Signed-Rank Tests.

## Results

The figures and findings presented in this section represent a summary of the complete experiment table which consists of performance measures and metadata for all $2,200$ models tested. The complete performance table alongside the table of significant differences, are made available through the GitHub repository.

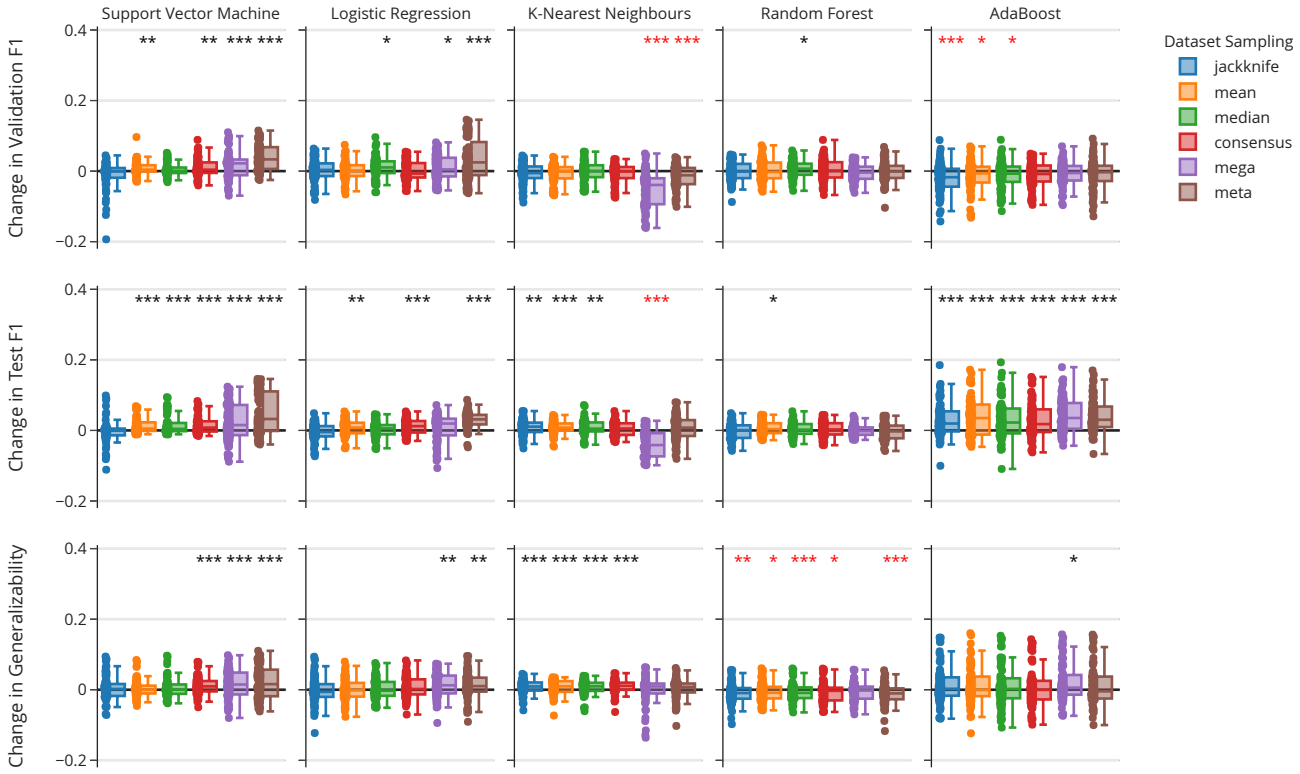


**Figure 1.** Relative change in classifier performance with respect to classifier type and dataset sampling strategies as measured by change in F1 score on the validation set (top) or test set (middle), as well as the generalizability of performance (bottom). Each star annotation indicates an order of magnitude of statistically significant change, beginning at 0.05 for one star and decreasing from there, with those in black or red indicating an increase or decrease due to resampling, respectively.

## Data Resampling Improves Classification

The change in performance for each model and dataset sampling technique is shown in Figure 1. The change in performance was measured as a change in F1 score on the validation set, the change in F1 score on the test set, and the change in overall generalizability, a measure which summarizes the similarity between validation and test performance for a given model.

The support vector machine and logistic regression models improve across each of these three measures for a variety of dataset sampling techniques, meaning that the addition of the MCA-perturbed samples improves the training, testing, and overall generalizability of the classifiers.

Distinctly, k-nearest neighbours (KNN) and AdaBoost classifiers experience minimal change in validation and often see their perfomance decline. However, the improvement of these classifiers on the test set suggests that resampling reduced overfitting in these classifiers. In the case of KNN, this translates to imrpoved generalizability, while in the case of AdaBoost generalizability was largely unchanged, suggesting that the model went from underperforming to overperforming after dataset resampling. The unique decline in performance when using the mega-analytic resampling technique on KNN classifier is suggestive of poor hyperparameterization, as there

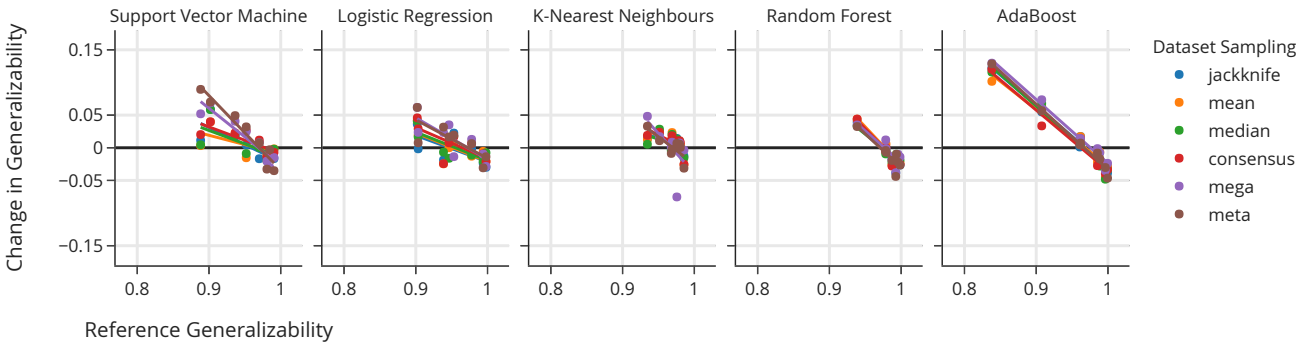


**Figure 2.** Change in the generalizability of classifiers with respect to the reference generalizability. Each data point represents the mean change in generalizability for all models using the same preprocessing and dimensionality reduction techniques for a given classifier and dataset sampling strategy.

is a strong relationship between the number of samples in the dataset and the $k$ parameter of the model. At present this parameter was scaled linearly with the number of MCA simulations used, however, it is both possible that an improved scaling function exists or that the model performance degrades with large sample sizes making it a poor model choice given this resampling technique.

The random forest classifiers uniquely did not see a significant change in validation or testing performance across the majority of resampling techniques. However, these classifiers did experience a significant decrease in the generalizability of their performance, meaning that there was a larger discrepancy between training and testing performance in many cases. This distinction from the other models is likely due to the fact that random forest is a simple ensemble technique which takes advantage of training many independent classifiers and samples them to assign final class predictions. It is likely that this approach forms more generalizable predictions generally, and thus the addition of more data does not improve performance further. While AdaBoost is also an ensemble method, the iterative training of models based on sample difficulty allows for the added variance in those samples to play an increasingly central role in the construction of class relationships.

Across all classifier types, it was found that both mega- and meta-analytic approaches outperformed other methods slightly, though this was not statistically significant. Additionally, while certain combinations of preprocessing, dimensionality reduction, and classifiers performed more harmoniously than others, there was no significant relationship between the performance of any single resampling method and preprocessing or dimensionality reduction technique. The above results show that dataset augmentation through MCA-perturbed pipeline outputs may be an effective way to improve the performance and generalizability of non-ensemble classifiers tasked with modeling brain-phenotype relationships, both within and out of sample.

### Model Improvement Scales With Generalizability

- Perturbation-enabled dataset resampling improves generalizability
- Less generalizable models improve more
- With the excecption of KNN, for which all models already exhibited high performance, there was a significant relationship between the baseline generalizability and the improvement from perturbations.
- The models which decrease in generalizability all have high generalizability scores ($> 0.935$), and this is likely

the paired result of "removing good luck" while the other is "removing bad luck"

### Mega-Analysis Improves With Samples

- While we previously note an increase in the generalizability when resampling, there is no relationship between the number of independent samples used and performance in most cases,
- however, in the case of the mega-analytic approach, there is a significant relationship between the number of samples used and generalizability.
- mega is the only approach that changes the number of samples being used by the classifiers, and this relationship is consistent to an increase one would expect when increasing the number of samples in their experiment.

## Discussion

- MCA augmentation is a good thing for performance, increasing our ability to train, test, and generalize performance in standard models.
- RF models benefitted the least from MCA, likely because they already combine a series of simple and independent representations towards making their decisions.
- Adaboost performance increased dramatically on the testing dataset, however, the lack of generalizability increase suggests the model went from underperforming to overperforming with the addition of MCA.
- learning is typically easier in a balanced class setting, and MCA could potentially be used as an oversampling method to increase the balance of datasets. The advantages of an approach like this are that all results are biologically plausible, and it would be possible in context when simulation methods specific to the data modality or processing technique are unavailable.
- While our work demonstrates that poorer performing classifiers benefit more from this resampling, the limit of that is unclear. An avenue for future work includes exploring the performance space more broadly, and identifying a relationship between baseline performance and impact of perturbation-based augmentation. It is unlikely that a model classifying with an F1 score of 0.95 would benefit the same as one with 0.75, or one performing near chance.
- Further work should situate this result relative to increased data collection. This was not performed here, despite the existence of an MCA-perturbed repeated-measures dataset in the previous study, as the sample size of 25 individuals was too small to meaningfully train and evaluate models in a paradigm consistent with other evaluatiions.
- This work demonstrates the benefit of performing perturbation experiments goes beyond stability evaluation, and that these methods support data augmentation strategies that significantly improve our ability to model brain-phenotype relationships.

### Data & Code Availability

The perturbed connectomes were publicly available data resource previously produced and made available by the authors[1]. They can be found persistently at `https://doi.org/10.5281/zenodo.4041549`, and are made available through The Canadian Open Neuroscience Platform (`https://portal.conp.ca/search`, search term "Kiar"). All software developed for processing or evaluation is publicly available on GitHub at `https://github.com/gkpapers/2020AggregateMCA`. Experiments were launched on Compute Canada's HPC cluster environment.

### Author Contributions

GK was responsible for the experimental design, data processing, analysis, interpretation, and the majority of writing.

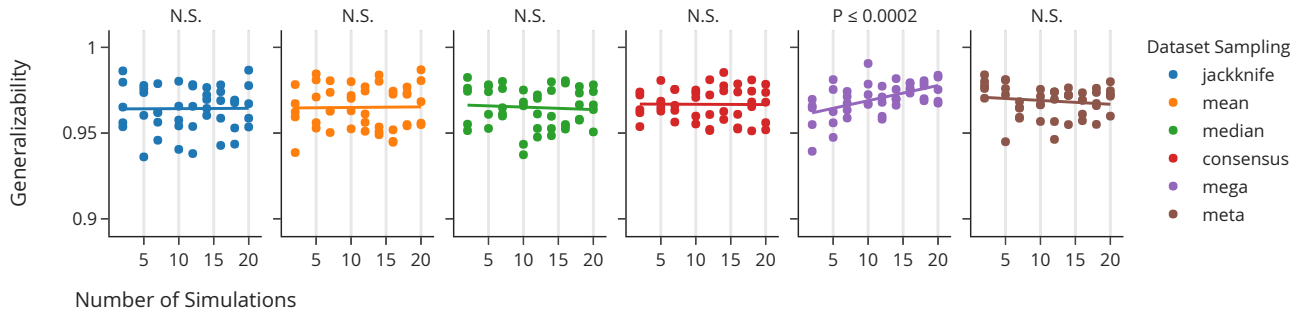


**Figure 3.** The generalizability of classifiers using each dataset sampling technique with respect to the number of MCA simulations. Each number of simulations was sampled a single time, to avoid artificial skewing of the dataset due to the inclusion of "higher" or "lower" quality samples; a single drawing of each split mimics a true perturbation experiment context.

All authors contributed to the revision of the manuscript. TG and ACE contributed to experimental design, analysis, interpretation. The authors declare no competing interests for this work. Correspondence and requests for materials should be addressed to Gregory Kiar at `gregory.kiar@mail.mcgill.ca`.

## Acknowledgments

This research was financially supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) (award no. CGSD3-519497-2018). This work was also supported in part by funding provided by Brain Canada, in partnership with Health Canada, for the Canadian Open Neuroscience Platform initiative.

## References

[1] G. Kiar, "Numerically perturbed structural connectomes from 100 individuals in the NKI rockland dataset," Apr. 2020.

[2] E. Garyfallidis, M. Brett, B. Amirbekian, A. Rokem, S. van der Walt, M. Descoteaux, I. Nimmo-Smith, and Dipy Contributors, "Dipy, a library for the analysis of diffusion MRI data," *Front. Neuroinform.*, vol. 8, p. 8, Feb. 2014.

[3] C. Denis, P. de Oliveira Castro, and E. Petit, "Verificarlo: Checking floating point accuracy through monte carlo arithmetic," *2016 IEEE 23nd Symposium on Computer Arithmetic (ARITH)*, 2016.

[4] D. S. Parker, *Monte Carlo Arithmetic: exploiting randomness in floating-point arithmetic*. University of California (Los Angeles). Computer Science Department, 1997.

[5] G. Kiar, Y. Chatelain, P. de Oliveira Castro, E. Petit, and others, "Numerical instabilities in analytical pipelines lead to large and meaningful variability in brain networks," *bioRxiv*, 2020.

[6] S. I. Dimitriadis, M. Drakesmith, S. Bells, G. D. Parker, D. E. Linden, and D. K. Jones, "Improving the reliability of network metrics in structural brain networks by integrating different network weighting strategies into a single graph," 2017.

[7] C. R. Buchanan, C. R. Pernet, K. J. Gorgolewski, A. J. Storkey, and M. E. Bastin, "Test–retest reliability of structural brain networks from diffusion MRI," *Neuroimage*, vol. 86, pp. 231–243, Feb. 2014.

[8] T. B. Meier, A. S. Desphande, S. Vergun, V. A. Nair, J. Song, B. B. Biswal, M. E. Meyerand, R. M. Birn, and V. Prabhakaran, "Support vector machine classification and characterization of age-related reorganization of functional brain networks," *Neuroimage*, vol. 60, no. 1, pp. 601–613, Mar. 2012.

[9] K. Wu, Y. Taki, K. Sato, S. Kinomura, R. Goto, K. Okada, R. Kawashima, Y. He, A. C. Evans, and H. Fukuda, "Age-related changes in topological organization of structural brain networks in healthy individuals," *Hum. Brain Mapp.*, vol. 33, no. 3, pp. 552–568, Mar. 2012.

[10] S. Y. Bookheimer, D. H. Salat, M. Terpstra, B. M. Ances, D. M. Barch, R. L. Buckner, G. C. Burgess, S. W. Curtiss, M. Diaz-Santos, J. S. Elam, B. Fischl, D. N. Greve, H. A. Hagy, M. P. Harms, O. M. Hatch, T. Hedden, C. Hodge, K. C. Japardi, T. P. Kuhn, T. K. Ly, S. M. Smith, L. H. Somerville, K. Uğurbil, A. van der Kouwe, D. Van Essen, R. P. Woods, and E. Yacoub, "The lifespan human connectome project in aging: An overview," *Neuroimage*, vol. 185, pp. 335–348, Jan. 2019.

[11] T. Zhao, M. Cao, H. Niu, X.-N. Zuo, A. Evans, Y. He, Q. Dong, and N. Shu, "Age-related changes in the topological organization of the white matter structural connectome across the human lifespan," *Hum. Brain Mapp.*, vol. 36, no. 10, pp. 3777–3792, 2015.

[12] A. Klein and J. Tourville, "101 labeled brain images and a consistent human cortical labeling protocol," *Front. Neurosci.*, vol. 6, p. 171, Dec. 2012.

[13] J. H. Ward, "Hierarchical grouping to optimize an objective function," pp. 236–244, 1963.

[14] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, and Others, "Scikit-learn: Machine learning in python," *the Journal of machine Learning research*, vol. 12, pp. 2825–2830, 2011.

[15] Y. Freund and R. E. Schapire, "A Decision-Theoretic generalization of On-Line learning and an application to boosting," *J. Comput. System Sci.*, vol. 55, no. 1, pp. 119–139, Aug. 1997.

[16] B. Tunç, B. Solmaz, D. Parker, T. D. Satterthwaite, M. A. Elliott, M. E. Calkins, K. Ruparel, R. E. Gur, R. C. Gur, and R. Verma, "Establishing a link between sex-related differences in the structural connectome and behaviour," *Philos. Trans. R. Soc. Lond. B Biol. Sci.*, vol. 371, no. 1688, p. 20150111, Feb. 2016.

[17] X. Zhu, X. Du, M. Kerich, F. W. Lohoff, and R. Momenan, "Random forest based classification of alcohol dependence patients and healthy controls using resting state MRI," *Neurosci. Lett.*, vol. 676, pp. 27–33, May 2018.

[18] S. Payabvash, E. M. Palacios, J. P. Owen, M. B. Wang, T. Tavassoli, M. Gerdes, A. Brandes-Aitken, D. Cuneo, E. J. Marco, and P. Mukherjee, "White matter connectome edge density in children with autism spectrum disorders: Potential imaging biomarkers using Machine-Learning models," *Brain Connect.*, vol. 9, no. 2, pp. 209–220, Mar. 2019.

[19] N. A. Crossley, A. Mechelli, J. Scott, F. Carletti, P. T. Fox, P. McGuire, and E. T. Bullmore, "The hubs of the human connectome are generally implicated in the anatomy of brain disorders," *Brain*, vol. 137, no. Pt 8, pp. 2382–2395, Aug. 2014.

[20] B.-Y. Park, J. Seo, J. Yi, and H. Park, "Structural and functional brain connectivity of people with obesity and prediction of body mass index using connectivity," *PLoS One*, vol. 10, no. 11, p. e0141376, Nov. 2015.

[21] D. R. Nayak, R. Dash, and B. Majhi, "Brain MR image classification using two-dimensional discrete wavelet transform and AdaBoost with random forests," *Neurocomputing*, vol. 177, pp. 188–197, Feb. 2016.

[22] E. Tolan and Z. Isik, "Graph theory based classification of brain connectivity network for autism spectrum disorder," in *Bioinformatics and Biomedical Engineering*. Springer International Publishing, 2018, pp. 520–530.

[23] R. F. Betzel, A. Griffa, P. Hagmann, and B. Misic, "Distance-dependent consistency thresholds for generating group-representative structural brain networks," *bioRxiv*, 2018.