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

words

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 traits, and compared this to traditional, baseline, performance on these tasks. We compared aggregation strategies with respect to baseline performance, the number of perturbed samples, and the distribution of the target variables. Finally, we compared the aggregation of unstable derivatives to aggregation obtained through the traditional acquisition of repeated measurements.

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 consists of two distinct collections containing i) a single session of data from 100 individuals (D100; $100 \times 1 \times 1$), or ii) two sessions which have each been subsampled into two components from 25 individuals (D25; $25 \times 2 \times 2$). In both cases, the derived brain networks were generated with a probabilistic structural connectome estimation pipeline[2] using a fixed random seed, and MCA perturbations were added to all Python-implemented operations throughout the pipeline. Each sample was simulated 20 times, resulting in $2,000$ unique graphs per dataset. Further information on the processing and curation of this dataset can be found here[2].

While the D100 collection enabled the exploration of subsampling and aggregation methods in a typical learning context for neuroimaging[3,4], D25 allowed for both the evaluation of these techniques in a small-sample setting and for their comparison to traditional approaches for data augmentation through repeated measurement, either across session or subsampling.

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[5–8], was selected and turned into a binary target by dividing participants into adult (> 18) and non-adult groups (D100: 68% adult, D25: 40% adult).

Preprocessing

The brain networks being used for classification were represented as symmetric 83×83 adjacency matrices. 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 or 15 dimensions of highest variance for D100 and D25, respectively. The number of components was chosen to capture approximately 90% of the variance present within D100. This was reduced due to limitations in sample size when dividing D25 into training, testing, and validation sets; approximately 85% of variance was explained through 15 components for D25.

Feature Agglomeration – the number of features in each connectome were reduced by combining edges according to maximum similarity/minimum variance using agglomerative clustering[9]. The number of resulting features was 20 and 15 for D100 and D25, respectively, 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[10]:

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.

Random Forest – 100 decision trees were fit using balanced class weights, each splitting the dataset according to a maximum of 4 or 3 features per node for D100 and D25, respectively (corresponding to the square root of 20 and 15).

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[11].

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 D100 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 across neuroimaging datasets[5,12–18].

Dataset Sampling

As a chief purpose of this manuscript involves the comparison of various forms of aggregation across simulated pipeline outputs, the datasets and classifiers were sampled, 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. For the repeated measures evaluation using D25, a single observation (first session, first subsample) per individual was selected.

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 in the case of MCA simulation experiments for both D100 and D25, and 10 times for repeated session or subsample experiments on D25. In all cases, the number of resamplings was 5× the number of unique observations per network, resulting in 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. This method was not used in the D25 repeated measurement setting due to a small and even number of samples.

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[19] 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. This approach was not used in the D25 repeated measurement setting, due to the small number of samples.

Mega-analysis – all observations of each network were used simultaneously for classification. 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 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 $2,200$ and $2,520$ models trained and evaluated for D100 and D25, respectively.

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 or network 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.

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

$$G = 1 - |F1_{test} - F1_{validation}| \quad (1)$$

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 value of the difference in performance was used in Eq. 1 resulting in a score which penalizes 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 using the Wilcoxon Signed-Rank Test.

Results

complete table for each of measure and model performance table, as well as and complete significance table, can be found in the dataset available through GitHub/Zenodo.

Data Resampling Improves Classification

Figure 1:

- svm and lrc performance improve on validation, test, and increase their generalizability. This global increase means that not only did the performance increase on both train and test, but the discrepancy between the performance on each was reduced.

- knn and adaboost both have minimal change to or reduced performance on the training set, though better performance on test, suggesting a reduction in overfitting. In the case of knn this leads to an increased generalizability, though in Adaboost it does not; the latter suggests a shift from under-performance to over-performance on the test set, rather than a more generalized performance.

- the reduced performance from knn and mega is likely due to the hyper-parameterization of the model, in which a number of neighbours must be specified and this value was scaled linearly with the number of MCA samples. It is possible that a better method exists for scaling the number of neighbours in this model, eliminating or reversing this effect.

- rf, the simple ensemble model, doesn't experience a change in performance or with dataset sampling methods, but the resulting models are less generalizable,

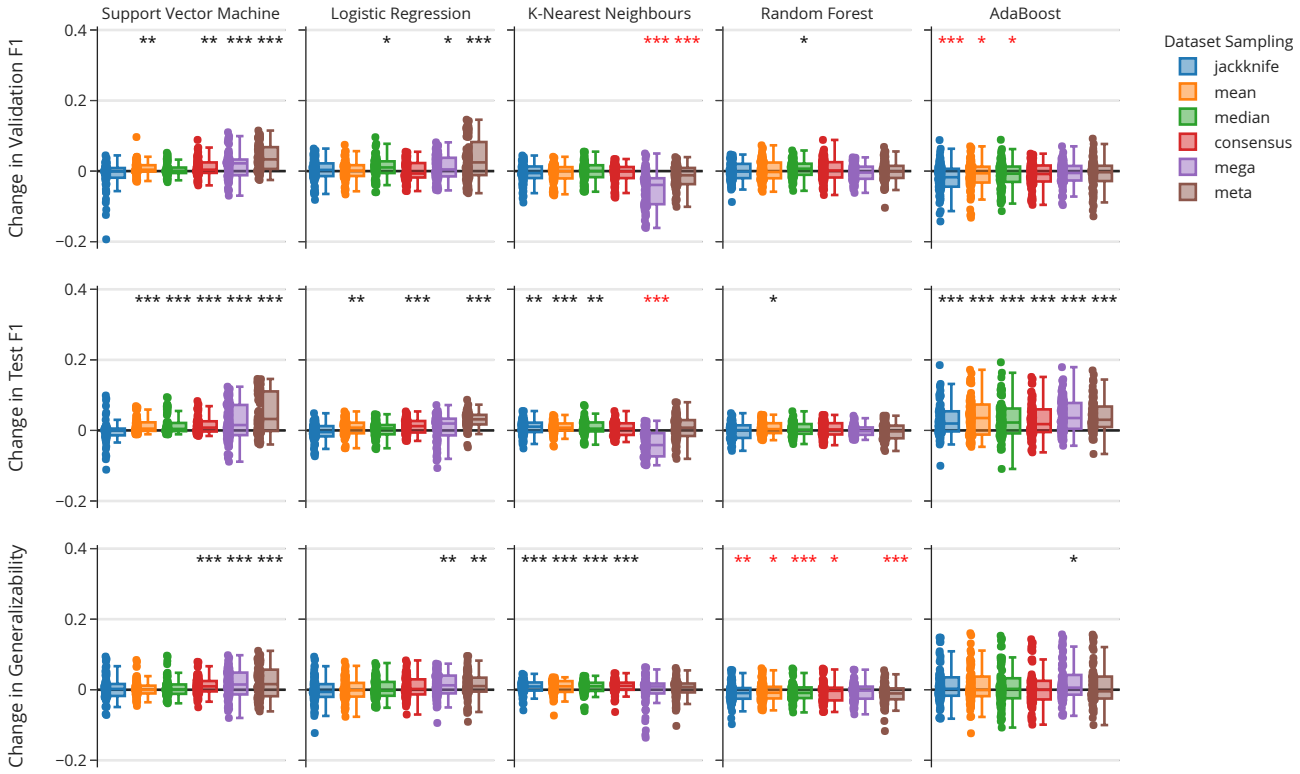

Figure 1. Relative change in classifier performance with respect to classifier type and dataset sampling strategies as measured by change in F1 score on the validation set (top) or test set (middle), as well as the generalizability of performance (bottom). Each star annotation indicates an order of magnitude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

meaning there is a larger discrepancy between validation and test performance.

- while mega and meta have a slight edge, there is no significant difference in sampling approaches.

- While certain combinations of preprocessing, dimensionality reduction, and classifiers performed better than others, there was no significant relationship between any dataset resampling methods and preprocessing or dimensionality reduction techniques.

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

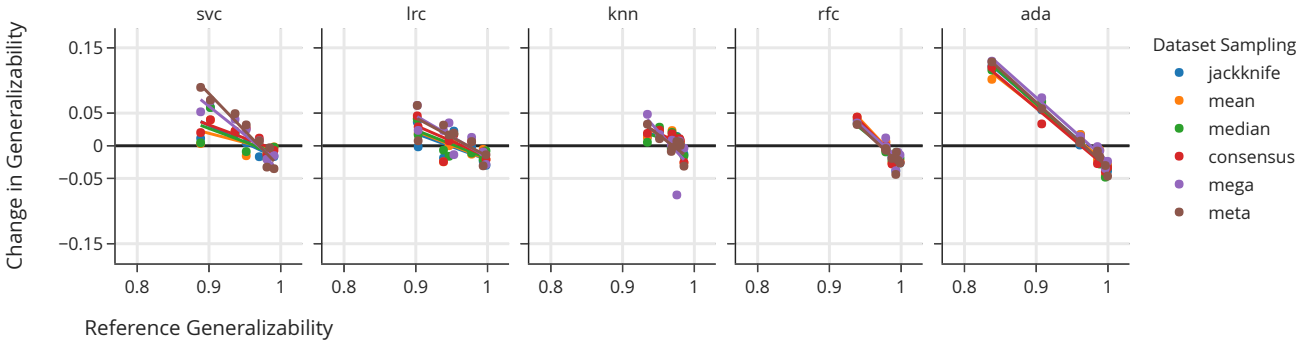

Figure 2. Change in the generalizability of classifiers with respect to the reference generalizability. Each data point represents the mean change in generalizability for all models using the same preprocessing and dimensionality reduction techniques for a given classifier and dataset sampling strategy.

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

- class imbalance... “Data is expensive, we don’t want to waste it, and biological oversampling (without a technique such as MCA) is hard... together, that’s why we went with balancing weights over undersampling”

- consider MCA as an oversampling technique which does not compromise the quality of samples or rely on interpolation in high dimensional data

- The first pass, we swept over both classifiers and targets, but I was lazy and didn’t make sure we had good performance, so results were all of high variance.

- The second pass, we made sure that we had good baseline performance in unique settings, but this required the addition of another degree of freedom through pre-processing, so the results when we mapped over all the combinations were still high variance and in some cases complete garbage.

- The third pass, we removed the degrees of freedom of target and classifier by fixing a single target to a single classifier. This led to better baseline performance for each task, but, of course, lead to the issue of not being able to actually draw conclusions on either target or classifier.

- For the fourth pass, I’m proposing to fix a single target, removing this degree of freedom entirely, but selecting one with both high performance and some variance across the set of classifiers used, so that we can model a relationship between model performance and impact of aggregation. If we wanted to extend this to other targets, too, I think we agree that it is useless to apply this on chance-classifiers; thus excluding BMI entirely, given the very high variance/generally weak relationship. Thus it would really only add sex to the mix, and wouldn’t particularly answer any questions about how consistent this is over other target variables, basically because

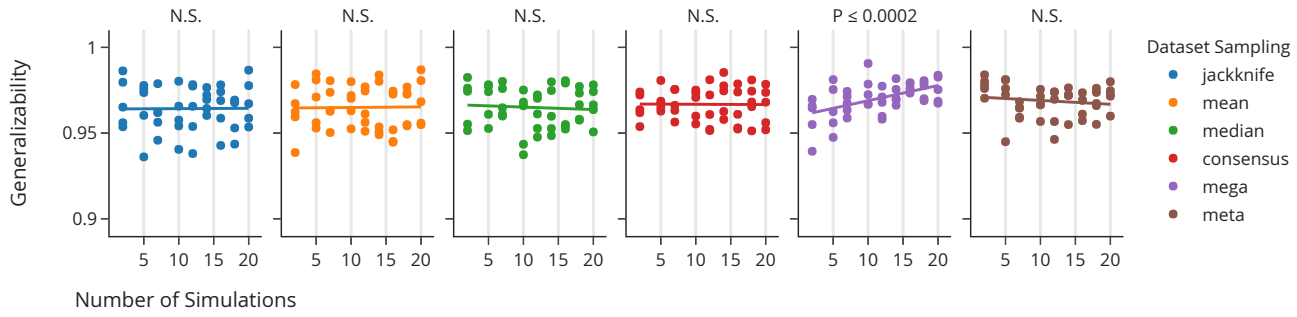

Figure 3. The generalizability of classifiers using each dataset sampling technique with respect to the number of MCA simulations. Each number of simulations was sampled a single time, to avoid artificial skewing of the dataset due to the inclusion of "higher" or "lower" quality samples; a single drawing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

"2 data points isn't a trend", despite complicating the picture further.

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. 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] 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.

[4] 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.

[5] 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.

[6] 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.

[7] 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.

[8] 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.

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

[10] 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.

[11] 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.

[12] 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.

[13] 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.

[14] 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.

[15] 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.

[16] 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.

[17] 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.

[18] 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.

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