Predicting cognitive scores with graph neural networks through sample selection learning

Martin Hanik · Mehmet Arif Demirtaş · Mohammed Amine Gharsallaoui · Islem Rekik

Received: date / Accepted: date

Abstract Analyzing the relation between intelligence and neural activity is of the utmost importance in understanding the working principles of the human brain in health and disease. In existing literature, functional brain connectomes have been used successfully to predict cognitive measures such as intelligence quotient (IQ) scores in both healthy and disordered cohorts using machine learning models. However, existing methods resort to flattening the brain connectome (i.e., graph) through vectorization which overlooks its topological properties. To address this limitation and inspired from the emerging graph neural networks (GNNs), we design a novel regression GNN model (namely RegGNN) for predicting IQ scores from brain connectivity. On top of that, we introduce a novel, fully modular sample selection method to select the best samples to learn from for our target prediction task. However, since such deep learning architectures are computationally expensive to train, we further propose a learning-based sample selection method that learns how to choose the training samples with the highest expected predictive power on unseen samples. For this, we capitalize on the fact that connectomes (i.e., their adjacency matrices) lie in the symmetric positive definite (SPD) matrix cone. Our results on full-scale and verbal IQ prediction outperforms comparison methods in autism spectrum disorder cohorts and achieves a competitive performance for neurotypical subjects using 3-fold cross-validation. Furthermore, we show that our sample selection approach generalizes to other learning-based methods, which shows its usefulness beyond our GNN architecture.

Keywords Regression · Graph Neural Network · Sample Selection · Functional Brain Connectome · Cognitive Score Prediction

1 Introduction

Understanding how the structure of the brain influences cognitive scores such as IQ plays a vital role in understanding the working principles of the human brain. Cognitive scores are indicators of intellectual capacity which were found to be strongly connected to social factors: while high correlation between intelligence scores measured in childhood and educational success were observed in (Colom et al., 2007; Deary et al., 2007), they were also linked to health and mortality (Gottfredson and Deary, 2004; Batty et al., 2007). Motivated by this fact, many studies have investigated how far intelligence quotients (IQ) can be predicted from the structure of the brain. It was found, for example, that cerebral volume positively correlates with cognitive ability (Reiss et al., 1996; Medaniel, 2005). On a finer scale, activity and global connectivity of parts of the brain, especially
of the lateral prefrontal cortex, are linked to IQ (Gray et al., 2003; Woolgar et al., 2010; Cole et al., 2012, 2015).

Against this background, recent works have explored the possibility to predict cognitive ability scores from functional brain connectomes (Pamplona et al., 2015; Dubois et al., 2018; Dadi et al., 2019; Dryburgh et al., 2020; He et al., 2020; Jiang et al., 2020). Conventionally, connectomes are obtained from resting-state MRI and characterize the network structure of the brain; they are modeled as graphs whose nodes represent regions of interest (ROIs) and whose edges correspond to correlations in activity between these ROIs (Sporns et al., 2005). In order to achieve better generalizability across contexts and populations, (Shen et al., 2017) proposed a data-driven protocol for connectome-based predictive modeling of brain–behavior relationships, using cross-validation, to train a linear regression model. Building upon it, (Dryburgh et al., 2020) improved the results by evaluating negative and positive correlations of brain regions separately. They performed their analysis on both neurotypical subjects and subjects with Autism Spectrum Disorder (ASD) in order to investigate how neural correlates of intelligence scores are altered by atypical neurodevelopmental disorders.

Although such works achieved significant success, they mainly relied on classical machine learning approaches, which do not incorporate the graph structure of the connectomes; therefore, the local and global topological properties of the connectomes are not leveraged. (He et al., 2020) introduced graph neural networks (GNNs) (Wu et al., 2021), a subfield of geometric deep learning, where learning is customized to non-Euclidean spaces such as graphs with complex topologies (Dehmamy et al., 2019). GNNs are deep neural networks with graph convolution layers. They have already lead to significant increases in performance over existing methods in many fields. For example, they have been successfully applied to classification tasks on networks (Kipf and Welling, 2017; Qu et al., 2019), image segmentation (Qi et al., 2017), feature matching (Sarlin et al., 2020), few-shot learning (Garcia and Bruna, 2017; Kim et al., 2019), and various graph mining tasks (Schlichtkrull et al., 2018; Yun et al., 2019; Zhang et al., 2019). A very recent review on GNNs in the field of network neuroscience (Bessadok et al., 2021) examined a variety of graph-based architecture tailored for brain connectivity classification, integration, superresolution and synthesis across time and modalities. However, none of the reviewed methods were designed for brain graph regression for cognitive score prediction.

In this paper, we propose the first GNN architecture, namely RegGNN, that is specialized in regressing brain connectomes to a target cognitive score to predict. Our GNN utilizes graph convolutional layers to map input connectomes onto their corresponding cognitive scores, thereby allowing to extract the learned weights to identify the brain connectivities between anatomical regions that fingerprint the target score.

To improve the performance of the GNN, we additionally propose a novel learning-based sample selection method. It is independent from RegGNN and can be used with any architecture or regression learner. The method identifies training samples with the highest predictive power (i.e., those that are most likely to predict unseen test subjects with the lowest error); only these are then used for training. Through this, we eliminate the samples that do not increase—or even decrease—the prediction success of the model and reduce the computational resources needed for training the GNN.

Within our sample selection method, we make use of the fact that the (weighted) adjacency matrix of a functional brain connectome, when modeled as a correlation matrix, is symmetric positive semi-definite; and becomes symmetric positive definite after a simple regularization step (Dodero et al., 2015; Wong et al., 2018; You and Park, 2021). The space of SPD matrices forms a nonlinear manifold (Arsigny et al., 2006), and like (You and Park, 2021), we use a Riemannian geometric structure on it in order to obtain a natural notion of distance between two connectomes as well as tangent matrices that encode the paths that realize this distance.

We summarize the main contributions of our work as follows:

1. We introduce a novel, learning-based sample selection method for graph neural networks that helps to increase accuracy when predicting cognitive scores from connectomes.
2. We propose novel similarity measures between brain connectomes by combining notions from Riemannian geometry and topology of graphs. These measures can be used in other applications whenever we deal with objects that can be interpreted as elements of Riemannian manifolds.
3. We design a pipeline, consisting of RegGNN with sample selection, which outperforms state-of-the-art models in predicting full scale intelligence and verbal intelligence quotients from functional brain connectomes in an autism spectrum disorder cohort and achieves a competitive performance in a neurotypical cohort.
2 Methods

In this section, we detail the architecture of our RegGNN. Furthermore, we introduce our proposed sample selection method and show how we incorporate it into the training process of the GNN. To start with, we recount some facts on the Riemannian geometry of SPD matrices. Furthermore, we recall graph-topological centrality measures. The mathematical notations that we use in the following are summarized in Table 1.

2.1 Preliminaries

The space of $n$-by-$n$ symmetric positive definite matrices $\text{SPD}(n) = \{ P \in \mathbb{R}^{n,n} : P^T = P, \text{ all eigenvalues of } P \text{ are positive} \}$ forms a cone-like manifold in the set of all matrices of the same size (Faraut and Korányi, 1994). Being a manifold, there is a well-defined tangent space at every point $P \in \text{SPD}(n)$, which we denote by $T_P\text{SPD}(n)$. It is a basic fact that each $T_P\text{SPD}(n)$ can be identified with the set of symmetric $n$-by-$n$ matrices. Therefore, in order to avoid later confusion, we call their elements tangent matrices instead of “tangent vectors” (which is the standard term in differential geometry).

As a manifold, $\text{SPD}(n)$ can be endowed with a Riemannian geometric structure (do Carmo, 1992). Such a structure is determined by the choice of a Riemannian metric, i.e., a smoothly varying inner product on the tangent spaces. With its help, we can measure angles between (and norms of) tangent matrices. Consequently, geodesics can be defined as (locally) shortest paths. Like a straight line in Euclidean space, a geodesic $\gamma$ that connects two points $P, Q \in \text{SPD}(n)$ can be represented by a unique tangent matrix $\log(P, Q) \in T_P\text{SPD}(n)$.

In particular, $\log(P, Q)$ points in the direction of $Q$ and has norm (measured in the one induced from the metric) equal to the distance between $P$ and $Q$. Because of this, we can view $\log(P, Q)$ as the linearized “difference” between $Q$ and $P$.

In contrast to Euclidean geometry, tangent matrices from different tangent spaces of a Riemannian manifold cannot be compared directly. Instead, they must be transported along curves to the same tangent space; this process is called parallel translation. This means that although tangent matrices at different points $P \in \text{SPD}(n)$ and $Q \in \text{SPD}(n)$ are symmetric matrices, we must bring them to a common point in order to compare them. The SPD space and parallel translation of vectors are illustrated in Fig. 1.

---

1 It is denoted like this because the corresponding map is called Riemannian logarithm.
for some small $\mu > 0$; see (Dodero et al., 2015) or (Wong et al., 2018). For training RegGNN—but not for the sample selection—we set all negative eigenvalues to zero, as positive correlations have been shown to be more important in brain network analysis (Fornito et al., 2016). Indeed, in our experiments the results improved when negative correlations were ignored. RegGNN receives the regularized, positive adjacency matrix $\mathbf{P}$ of a connectome and predicts the corresponding IQ score from it by applying graph convolutions.

In the literature, there are various implementations of graph convolutions, which mainly differ by the propagation rule. For $i = 0, 1, 2$ let $\mathbf{H}^{(i)}$ denote the activation matrix at the $i$-th layer with initialization $\mathbf{H}^{(0)} := \mathbf{I}$. It is propagated to the next layer according to the general rule $\mathbf{H}^{(i+1)} = g_i(\mathbf{H}^{(i)}, \mathbf{P})$ with functions $g_i : \mathbb{R}^{d_i \times d_i} \times \text{SPD}(d) \to \mathbb{R}^{d_{i+1} \times d_{i+1}}$ for $i = 0, 1$, where $d_0 := d$ and $d_2 := 1$. (The integer $d_1$ can be chosen appropriately—we use a suitable number near $d/2$.) We choose the $g_i$ as proposed in (Kipf and Welling, 2017). Define $\tilde{\mathbf{P}} := \mathbf{P} + \mu \mathbf{I}$ and let $\tilde{\mathbf{D}}$ be the diagonal degree matrix of $\tilde{\mathbf{P}}$, we then formalize $g_i$ as follows:

$$g_i(\mathbf{H}^{(i)}, \mathbf{P}) := \text{ReLU}(\tilde{\mathbf{D}}^{-\frac{1}{2}} \tilde{\mathbf{P}} \tilde{\mathbf{D}}^{-\frac{1}{2}} \mathbf{H}^{(i)} \mathbf{W}^{(i)}),$$

where $\mathbf{W}^{(i)} \in \mathbb{R}^{d_i \times d_i}$ is the learnable weight matrix. We thus use the graph convolution layers to reduce the size of the connectomes and obtain an embedding for the brain graphs into $\mathbb{R}^d$. We apply a dropout layer after the first graph convolution operation for regularization. Finally, the obtained embedding passes through a fully connected layer (linear layer) which produces a continuous scalar output. The goal of the linear layer is to embed the resulting vector containing $d$ features into a scalar value representing the predicted IQ score.

We now recall three basic topological centrality measures for an undirected2 graph $G$: degree centrality, eigenvector centrality, and closeness centrality; we recount them in Appendix A. They measure how far a node is central to the (graph) network in the sense that most of the communication passes through it. A good reference on this is the book (Fornito et al., 2016).

We are now ready to introduce the graph neural network, and afterwards, the sample selection process.

### 2.2 RegGNN

Our GNN for regression, RegGNN, consists of two graph convolution layers and a downstream fully connected layer; a visualization is on the bottom left of Figure 2. In the following we denote the number of ROIs by $d$. Since adjacency matrices of connectomes are $d$-by-$d$ correlation matrices $\mathbf{C}$, they can have zero (but no negative) eigenvalues. Therefore, we can simply regularize them to being symmetric positive definite by adding a small multiple of the identity matrix $\mathbf{I}$, i.e.,

$$\mathbf{P} := \mathbf{C} + \mu \mathbf{I}$$

for some small $\mu > 0$; see (Dodero et al., 2015) or (Wong et al., 2018). For training RegGNN—but not for the sample selection—we set all negative eigenvalues to zero, as positive correlations have been shown to be more important in brain network analysis (Fornito et al., 2016). Indeed, in our experiments the results improved when negative correlations were ignored. RegGNN receives the regularized, positive adjacency matrix $\mathbf{P}$ of a connectome and predicts the corresponding IQ score from it by applying graph convolutions.

In the literature, there are various implementations of graph convolutions, which mainly differ by the propagation rule. For $i = 0, 1, 2$ let $\mathbf{H}^{(i)}$ denote the activation matrix at the $i$-th layer with initialization $\mathbf{H}^{(0)} := \mathbf{I}$. It is propagated to the next layer according to the general rule $\mathbf{H}^{(i+1)} = g_i(\mathbf{H}^{(i)}, \mathbf{P})$ with functions $g_i : \mathbb{R}^{d_i \times d_i} \times \text{SPD}(d) \to \mathbb{R}^{d_{i+1} \times d_{i+1}}$ for $i = 0, 1$, where $d_0 := d$ and $d_2 := 1$. (The integer $d_1$ can be chosen appropriately—we use a suitable number near $d/2$.) We choose the $g_i$ as proposed in (Kipf and Welling, 2017). Define $\tilde{\mathbf{P}} := \mathbf{P} + \mu \mathbf{I}$ and let $\tilde{\mathbf{D}}$ be the diagonal degree matrix of $\tilde{\mathbf{P}}$, we then formalize $g_i$ as follows:

$$g_i(\mathbf{H}^{(i)}, \mathbf{P}) := \text{ReLU}(\tilde{\mathbf{D}}^{-\frac{1}{2}} \tilde{\mathbf{P}} \tilde{\mathbf{D}}^{-\frac{1}{2}} \mathbf{H}^{(i)} \mathbf{W}^{(i)}),$$

where $\mathbf{W}^{(i)} \in \mathbb{R}^{d_i \times d_i}$ is the learnable weight matrix. We thus use the graph convolution layers to reduce the size of the connectomes and obtain an embedding for the brain graphs into $\mathbb{R}^d$. We apply a dropout layer after the first graph convolution operation for regularization. Finally, the obtained embedding passes through a fully connected layer (linear layer) which produces a continuous scalar output. The goal of the linear layer is to embed the resulting vector containing $d$ features into a scalar value representing the predicted IQ score.

---

2. Of course, the centrality measures can also be defined for directed graphs but we do not need this here.

---

### Table 1: Major mathematical notations used in this paper.

| Notation | Dimension | Definition |
|----------|-----------|------------|
| $n_{\text{Train}}$ | $N$ | number of subjects in the training group |
| $n_{\text{Test}}$ | $N$ | number of subjects in the test group |
| $d$ | $N$ | number of brain regions (i.e., ROIs) |
| $n_s$ | $N$ | number of elements in the train-in group |
| $n_h$ | $N$ | number of elements in the holdout group |
| SPD($n$) | $\cdot$ | manifold of $n \times n$ symmetric positive definite matrices |
| $\mathbf{I}$ | $\mathbb{R}^{d \times d}$ | identity matrix |
| $\mathbf{P}^s$ | $\mathbb{R}^{d \times d}$ | functional brain connectome (SPD) of subject $s$ from the train-in group |
| $\mathbf{P}^h$ | $\mathbb{R}^{d \times d}$ | functional brain connectome (SPD) of subject $l$ from the holdout group |
| $S_{i,j}^{s,s}$ | $\mathbb{R}^{d \times d}$ | tangent matrix (symmetric matrix) encoding the geodesic between connectomes $i$ and $j$ from the train-in group $s$ at $\mathbf{P}^s_i$ |
| $S_{i,j}^{h,l}$ | $\mathbb{R}^{d \times d}$ | tangent matrix (symmetric matrix) encoding the geodesic between connectomes $j$ from the train-in group $s$ and $l$ from the holdout group $h$ at $\mathbf{P}^h_j$ |
| $\tilde{S}_{i,j}^{s,s}$ | $\mathbb{R}^{d \times d}$ | parallel translation (symmetric matrix) of $S_{i,j}^{s,s}^{-1}$ to $\mathbf{TSPD}(n)$ |
| $\tilde{S}_{i,j}^{h,l}$ | $\mathbb{R}^{d \times d}$ | parallel translation (symmetric matrix) of $S_{i,j}^{h,l}^{-1}$ to $\mathbf{TSPD}(n)$ |
| $V_{i,j}^s$ | $\mathbb{R}^d$ | feature vector extracted from $S_{i,j}^{s,s}^{-1}$ |
| $V_{i,j}^h$ | $\mathbb{R}^d$ | feature vector extracted from $S_{i,j}^{h,l}^{-1}$ |
| $C_{i,j}^s$ | $\mathbb{R}$ | cognitive score (IQ) of subject $i$ from the train-in group $s$ |
| $C_{i,j}^h$ | $\mathbb{R}$ | cognitive score (IQ) of subject $i$ from the holdout group $h$ |
Fig. 2: Illustration of the proposed sample selection strategy to train our regression graph neural network RegGNN. A) We split the data in training (green) and testing (violet) sets. B-i) After the training set is divided into a train-in (yellow) and a holdout (red) set, we extract tangent matrices for geodesics connecting elements from the train-in set (yellow) and tangent matrices encoding geodesics from elements of the train-in to elements of the holdout group (red). B-ii) The information from the tangent matrices is compressed into vectors through topological feature extraction. B-iii) With linear regression we train a mapping $f$ on the train-in-to-train-in feature vectors (yellow) to learn differences in target score and record for each element $j$ of the train-in group the $k$ elements from the holdout group for whom the predicted difference in target score to $j$ was smallest. B-iv) For each sample in the holdout set, we count how often it was among the top $k$ predictors for a sample from the train-in group. C) After repeating B) in an N-fold cross validation manner, the $k$ samples (blue) with the highest accumulated top-$k$ frequency are selected. After negative correlations have been set to zero the graph neural network is trained (only) on them. Finally, the testing set is used to evaluate the overall performance.

2.3 Learning-based Sample Selection

We now introduce our learning-based sample selection strategy. The underlying idea is the following. Imagine the (rather extreme) case that our subjects are clustered (possibly with outliers) in $k$ tight groups according to their cognitive scores. Then, training a GNN on $k$ representatives, one from each cluster, should yield good results; it should even perform better than a GNN trained on the full data set because it was not “distracted” by outliers during training. Ideally, as representatives we would choose the $k$ most central samples of each group, i.e., those with the smallest average difference in cognitive score to the other samples. Now, since we want to predict cognitive scores from connectomes, we do not know the differences beforehand. On the other hand, existing studies validated the relationship between brain connectivity patterns and brain behavior and cognition. For instance, recent papers (Pamplona et al., 2015; Shen et al., 2017; Dubois et al., 2018; Dadi et al., 2019; Dryburgh et al., 2020; Jiang et al., 2020; He et al., 2020) have shown that the cognitive ability of a person can be predicted quite accurately from the human connectome, indicating that
the brain cognitive and behavior are encoded in its connectivity to a measurable degree. Such prediction would have been elusive if similar data inputs (here brain connectomes) cannot be mapped to similar outputs (here cognitive scores). Consequently, we assume that similar brain connectivity networks are correlated in cognition whereas brain connectomes that vary in topological patterns might elicit different cognitive scores. Such hypothesis might seem somewhat reductionist as there are many other factors that contribute to molding and predicting brain cognition such as genetics and epigenetics (Goldberg and Weinberger, 2004; Deary et al., 2006; Reichenberg et al., 2009). However, such factors remain out of the scope of this study. Therefore, our idea is to learn the differences between the connectomes to identify those “representatives”. Our experiments below show that this idea of—to represent predicted local aggregations of data by (few) representatives and training only with them—generalizes well to real data.

Implementing the idea, we represent differences between connectomes by tangent matrices and assume that the difference in IQ between two subjects depends linearly on (notions deduced from) the tangent matrix log(\(P_i, Q_j\)) that encodes the geodesic between the corresponding connectomes \(P, Q \in \text{SPD}(d)\). This model is flexible, but at the same time allows for fast computations. Our sample selection method learns this linear map, which we call \(f\) in the following, via regression and uses it to identify the \(k\) samples with the lowest predicted average difference in target score to all other samples. As motivated above, we assume that they are representative of the whole set but do not contain (most of) the outliers that hinder successful training of the GNN. The structure and terminology of our method are inspired by the work of (Errica et al., 2019).

The sample selection method consists of four steps, which are visualized in part B of Fig. 2. Given a connectome data set, these are repeated in a nested \(N\)-fold cross-validation manner to make our selection of samples more robust. In cross-validation, we split the data into two groups: a training subset which we call train-in group, and a validation subset which we call holdout group; we perform different train-in and holdout group splits so that each sample from the training set will be in the train-in group exactly \(N-1\) times. We denote the (constant) sizes of the train-in and the holdout sets by \(n_s\) and \(n_h\), respectively.

1) Riemannian tangent matrix derivation. For each pair of regularized connectomes \(P_i^s, P_j^s \in \text{SPD}(d)\) in the train-in group, we compute the tangent matrix that encodes the geodesic between them and parallel translate it to \(T_{\text{SPD}}(d)\); we denote the resulting symmetric \(d\)-by-\(d\) matrix by \(S_{i,j}^{s,s}\). As a result, we obtain a set of \(n_s (n_s - 1)/2\) tangent matrices in \(T_{\text{SPD}}(d)\) that represent the pairwise differences between the connectomes from the train-in group. Analogously, we get a tangent matrix \(S_{j,l}^{s,h} \in T_{\text{SPD}}(d)\) for each pair with one sample \(P_i^s\) from the train-in and another sample \(P_j^h\) from the holdout group; this results in another set consisting of \(n_s n_h\) tangent matrices. The latter are the outgoing “difference matrices” from the train-in into the holdout set.

2) Topological feature extraction from tangent matrices. The tangent matrices are still rather high dimensional, which leads to long computation times. Thus, we suggest to extract topological features in order to encode the information in more compact form. We select degree, closeness, and eigenvector centrality as well as combinations of them as our candidates for feature extraction. Note that in our case a tangent matrix represents the “difference” between two connectomes. The above features thus encode information on linearized changes in node connectivity. To the best of our knowledge, this is the first time that these notions were used in conjunction. As a result, from all in-group tangent matrices \(S_{i,j}^{s,s}\) as well as outgoing tangent matrices \(S_{j,l}^{s,h}\) we obtain feature vectors \(v_{i,j}^{s,s}\) and \(v_{i,j}^{s,h}\), respectively.

3) Learning a linear regression mapping for predictive sample selection. We learn the linear map \(f\) via regression by training to map the vectors \(v_{i,j}^{s,s}\) corresponding to samples \(i\) and \(j\) from the train-in group to the absolute difference in target score \(|IQ_i - IQ_j|\) between them. We then apply the learned linear regression mapping \(f\) to the vectors \(v_{i,j}^{s,h}\) to predict the differences in target score between all samples \(j\) from the train-in and samples \(l\) from the holdout group.

4) Frequency map. We record for each holdout sample \(P_k^h\) the \(k\) subjects from the train-in group with the smallest predicted difference under \(f\) and increment a frequency map (i.e., a counter) that is initialized at the start of the sample selection process. The frequency value of a subject is then the number of times it was one of the top \(k\) predictive samples. These frequencies give an approximated ranking whereby the top samples are closest to the largest number of other samples in (predicted) target score.

After the cross-validation is finished, we extract the top \(k\) samples\(^3\) with the highest cumulative frequencies. We expect these samples to have the highest represen-
Sample selection for IQ prediction on connectomes using GNNs

Copyright (c) 2021 by the authors. Published by the American Association for Artificial Intelligence. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the Association.

Sample selection for IQ prediction on connectomes using GNNs 7
tative power as they consistently predicted samples in
different holdout groups with low error.

2.4 Training Process

In the following, we explain how we integrate the sam-
ple selection method into the training process of Reg-
GNN. The whole pipeline is shown in Fig. 2.

Given the data set of connectomes our proposed training pipeline consists of the following steps A-C.

A- Training-test split. First, we split the data set
into a training and a test set. The test set is used only
for the final evaluation of RegGNN.

B- Learning-based sample selection. Then, we
select the top \( k \) samples with the highest representative
power from the training set by applying the sample selec-
tion method from Sec. 2.3.

C- RegGNN architecture for regression. Fi-
ally, we train RegGNN on the top \( k \) samples using
cross-validation to evaluate model generalizability against
perturbations of training and testing data distributions.
The final testing is done on the unseen test set.

2.5 Data and Methodology

We used the pipeline from Sec. 2.4 to predict the full
scale intelligence quotient (FIQ) and the verbal intelli-
gence quotient (VIQ) from brain connectomes for both
neurotypical (NT) subjects as well as subjects with
autism spectrum disorder (ASD). In the following, we
summarize these experiments.

Dataset. We used samples from the Autism Brain
Imaging Data Exchange (ABIDE) Preprocessed dataset
(Craddock et al., 2013) for our experiments. It contains
data from 16 imaging sites, preprocessed by five differ-
tent teams using four pipelines: the Connectome Compu-
tation System (CCS), the Configurable Pipeline for the
Analysis of Connectomes (CPAC), the Data Processing
Assistant for rs-fMRI (DPARSF) and the NeuroImag-
ing Analysis Kit. The preprocessed data sets are avail-
able online\(^4\). To account for possible biases due to differ-
cences in sites, we used randomly sampled subsets of the
available data for both cohorts; the same sets were also
used by (Dryburgh et al., 2020). The NT cohort con-
sisted of 226 subjects (with mean age \( = (15 \pm 3.6) \)),
while the ASD cohort was made up of 202 subjects
(with mean age \( = (15.4 \pm 3.8) \)). FIQ and VIQ scores
in the NT cohort have means 111.573 \( \pm 12.018 \) and
112.787 \( \pm 12.056 \), whereas FIQ and VIQ scores in the
ASD cohort have means 106.102 \( \pm 15.045 \) and 103.005
\( \pm 16.874 \), respectively. The brain connectomes were ob-
tained from resting-state functional magnetic resonance
imaging using the parcellation from (Tzourio-Mazoyer
et al., 2002) with 116 ROIs. The functional connectomes
are represented by 116-by-116 matrices, whose entry in
row \( i \) and column \( j \) is the Pearson correlation between
the average rs-fMRI signal measured in ROI \( i \) and ROI
\( j \).

Software. All experiments are done in Python 3.7.10.
We used Scikit-learn 0.24.2 (Pedregosa et al., 2011)
for machine learning models and PyTorch Geometric
1.6.3 (Fey and Lenssen, 2019) for graph neural network
implementations. For Riemannian geometric computa-
tions in the SPD space, we used the SPD class from the
Morphomatics package of (Ambellan et al., 2021). To
extract the graph topological features from the tangent
matrices we used NetworkX (Hagberg et al., 2008).

Parameter settings. We trained our method with Adam optimizer (Kingma and Ba, 2017) for 100 epochs
with a learning rate of 0.001 and weight decay at 0.0005
based on our empirical observations. Furthermore, we
set \( d_1 := 64 \) in RegGNN. The dropout rate after the
first graph convolutional layer was set to 0.1. To regu-
larize the adjacency matrices, we used \( \mu = 10^{-10} \) in (1).
In order to explore the parameter space for the number
of selected training samples \( k \), we varied it between 2
and 15.

Evaluation and comparison methods. To test
the generalizability and robustness of our method, we
used 3-fold cross-validation on both NT and ASD co-
horts separately for both FIQ and VIQ prediction. We
report the mean absolute error (MAE) and the root
mean squared error (RMSE) for all methods. For the
sample selection methods, we additionally give the mean,
standard deviation, minima and maxima over all tested
\( k = 2, \ldots, 15 \) to test our sample selection methods sen-
sitivity to the selection of \( k \).

To benchmark against our method, we chose state-
of-the-art methods from both deep learning and ma-
chine learning. The first baseline was CPM (Shen et al.,
2017), which was specifically designed for behavioral
score prediction on brain connectomes; the second be-
ing PNA (Corso et al., 2020), which outperformed com-
mon GNNs on both artificial and real-world benchmark
regression tasks (but has not been applied to brain con-
nectomes yet). PNA comes with principal neighborhood
aggregation layers that are defined similarly to graph
convolution operations. They are designed to increase
the amount of information that is used from the lo-
cal neighborhoods in the graphs. In our experiments,
we inserted PNA layers in our RegGNN architecture.
We implemented both a simpler setup with sum ag-
gregation and identity scaling only (denoted by PNA-

\(^4\) http://preprocessed-connectomes-project.org/abide/
S), as well as various aggregation (sum, mean, var and max) and scaling (identity, amplification and attenuation) methods (denoted by PNA-V) as detailed in the paper of (Corso et al., 2020). The code of both CPM\(^5\) and PNA\(^6\) is available online.

In order to assess the effect of the sample selection method, we also always trained each architecture on all samples as a baseline.

**Evaluation of the sample selection.** For each architecture, we compared several methods that can be used as measure of difference in the sample selection (viz., Sec. 2.3 part (ii)) to train the linear mapping \(f\).

The first class of methods was the proposed one: we encoded the differences via tangent matrices in the SPD space. To identify a good choice for handling the information that is contained in the tangent matrices, we compared several methods. As one option, we trained \(f\) on the vectorized upper triangular part (including the diagonal) of the tangent matrix; this method is denoted by (tm). Since the matrices are symmetric, ignoring the lower part speeds up computations while not losing information. Further, we used degree centrality, eigenvector centrality, and closeness centrality (see App. A), and applied them to the tangent matrices; they are denoted by (dc), (ec), and (cc), respectively. Note that during the process, the topology of each connectome is not altered. The mapping \(f\) was then trained on the resulting centrality vectors. Additionally, we tested whether the concatenation of the above centrality measures into a single vector is even more informative. To this end, we used both an unscaled and a scaled version, denoted by (cnu) and (cns) respectively. The unscaled version was generated by simple concatenation of the three feature vectors. However, as the three centrality measures have different ranges, we additionally tested scaling each feature vector first. For this, we used min-max scaling. Remember that min-max scaling of a vector \(v\) is defined element-wise by

\[
\tilde{v}_i := \frac{v_i - \max(v)}{\max(v) - \min(v)}.
\]

Each centrality vector was scaled before concatenating, which then gave a vector with elements in \([0, 1]\) as data for the regression.

We complemented these methods with two baselines. In order to check whether the additional directional information that the tangent matrices contain helps, we also tested whether it suffices to train \(f\) on the Riemannian geometric distances \(d(P^*_i, P^*_j)\) between the connectomes from the train-in group alone; this method is denoted by (g). To assess whether we improve by using the manifold structure of the SPD space at all, we trained \(f\) on the Euclidean absolute distance between the upper triangular parts \(\hat{P}^*_i, \hat{P}^*_j\) of each pair of connectomes \(P^*_i, P^*_j\), i.e., on the scalars \(\|\hat{P}^*_i - \hat{P}^*_j\|_F\) (F standing for the Frobenius norm); we denote this method by (a).

We report the \(p\)-value between the best performing sample selection method MAE and the baseline MAE according to a t-test for all architectures.

### 3 Results and Discussion

The results for the NT and ASD cohorts are shown in Tables 2 and 3, respectively. We observe that while the state-of-the-art machine learning model CPM surpasses naive applications of GNNs in the form of PNA, our RegGNN, paired with sample selection, outperforms CPM in all tasks according to both MAE and RMSE with the exception of the NT (FIQ) task. Improvements by our method are especially visible in the ASD cohort. Interestingly, we see that the results of all methods are worse on the ASD cohort compared to the NT cohort. This was also observed by (Dryburgh et al., 2020). We hypothesize that the difficulty of predicting IQ scores in ASD cohort might be caused by the inter-subject heterogeneity that is characteristic for ASD (Tordjman et al., 2018). Another factor may be that ASD samples from ABIDE are biased towards high-functioning individuals (Craddock et al., 2013).

We observe further that sample selection improved the results for RegGNN in all tasks except ASD (VIQ), and for CPM in the ASD cohort. For PNA based architectures, there are drastic improvements in NT (FIQ) and ASD (VIQ) and incremental improvements in the remaining other tasks. An exception to this is the PNA-V setup for ASD (FIQ), where most models with sample selection perform worse than the one trained on all samples. This might be partly explained by the more complicated structure of PNA with various aggregation models, which might demand more samples for correct training.

For all models, we see that the minimum MAE over \(k\) is lower than the MAE version that was trained on the full data sets even when the mean MAE across \(k\) is higher in all tasks. This indicates that improvements are highly likely with fine-tuning of parameter \(k\).

Our experiments did not reveal a clear trend for the value of \(k\) for which the minimum was attained. Nevertheless, our observations show that the proposed RegGNN network is more stable to changes in the parameter \(k\). Calculating for each architecture the average

---

\(^5\) https://github.com/esfinn/cpm_tutorial

\(^6\) https://github.com/lukecavabarrett/pna/
of the standard deviations (std) of the mean absolute error\footnote{The standard deviations include variations over different \( k \).} (see the results table) over all feature extraction methods, we first note that the averages for RegGNN are 0.455, 0.145, 0.877, 0.369 for NT (FIQ), NT (VIQ), ASD (FIQ) and ASD (VIQ), respectively. While RegGNN therefore shows small variation with respect to \( k \), CPM is highly sensitive to the changes of this parameter with averages of 2.901, 2.613, 1.803 and 2.754 respectively. This is approximately a 2 to 10 fold increase in variability. Consequently, RegGNN can better capture the brain graph structure, whereas CPM treats graphs as flattened vectors without preserving their topological features.

Improvements to the performance of CPM are not statistically significant (\( p = 0.87 \) for ASD (FIQ), \( p = 0.15 \) for ASD (VIQ)). Similarly, we observed that improvements to the performance of RegGNN are only statistically significant in NT (VIQ) task (\( p < 0.05 \) for NT (VIQ), \( p = 0.98 \) for NT (FIQ), \( p = 0.64 \) for ASD (FIQ)). The performance increases for PNA models are more consistent, as PNA-S improved significantly in three out of four tasks (\( p < 0.01 \) for NT (FIQ), \( p = 0.11 \) for NT (VIQ), \( p < 0.05 \) for ASD (FIQ), \( p < 0.01 \) for ASD (VIQ)), and PNA-V improved significantly in two out of four tasks (\( p < 0.05 \) for NT (FIQ), \( p = 0.21 \) for ASD (FIQ)).

Fig. 3: The top three most relevant brain regions for each task according to the learned weights extracted from the last layer of the trained RegGNN. The colors indicate the task. NT: neurotypical subjects. ASD: autism spectrum disorder subjects. FIQ: fluid intelligence quotient. VIQ: verbal intelligence quotient.
NT (VIQ), \( p = 0.11 \) for ASD (FIQ), \( p < 0.01 \) for ASD (VIQ)).

Within the sample selection pipelines, the best performing methods always utilize the Riemannian geometric structure of the SPD space with respect to MAE, apart from PNA-V results for the NT (VIQ) task. In the majority of cases, the methods that rely on tangent matrices perform best with the vectorized version of the whole tangent matrix being the best method in NT (VIQ) and ASD (FIQ). We also see that the three centrality measures and concatenated versions perform well consistently. Our results do not reveal any finer pattern among the sample selection measures, but we can conclude that using Riemannian structure of connectomes to estimate their predictive power outperforms methods that do not leverage these geometric properties. Thus, other metrics should also be considered when deciding for one. The tangent matrix method seems to perform very well but is also the most time consuming since no dimension reduction is performed. On the contrary, computing centrality measures significantly reduces the size of the matrices which speeds up the process sufficiently. In our experiments, we observed that training linear regression models using tangent matrices took up to 16 times more time compared to training models using centrality measures. To understand the latter methods and how they work better, it would be helpful to analyze their behavior on the tangent matrices mathematically. In contrast to their use for adjacency matrices of graphs, this is, to the best of our knowledge, unknown. It is thus an interesting venue for future work.

An important advantage of using sample selection in training graph neural networks is the decrease in the computational power needed for the training process. Using the computational power more efficiently leads to shorter training times on fixed amount of data, which opens up opportunities to train more complex models on more data or in shorter amounts of time. While the exact time required for sample selection is heavily dependent on the hardware used and varies based on the model architecture, number of epochs in training, number of training samples, and the number \( k \) of samples to select, our observations during the experiments show that sample selection reduces the training time by 20% on average. Therefore, usage of our sample selection pipeline can enable the use of deeper neural network architectures on connectomes and provides a topic of interest in future work.

So far, we evaluated our method on a young population; however, our RegGNN demonstrated its generalizability by the utilized cross-validation strategy and across both NC and ASD brain connectivity datasets. The proposed model can be easily used to map a particular brain connectivity population (e.g., elderly population) to target scores to predict. To proliferate replication studies on other cohorts, we publicly shared our RegGNN source code\(^8\).

**Explainability and biomarker discovery.** In order to identify the brain regions of interest that influence the prediction most, we extracted for each of the four tasks the learned weights of the RegGNN utilizing the best-performing sample selection method. The weights came from the fully connected layer that maps a 116-dimensional vector to the output score. (Remember that the input vector represents the learned embedding of the graph.) Thanks to the end-to-end network training, the backpropagation process as well as our network design (which preserves the structure of the connectome in both the first and second layer), the learned weights in the final fully connected layer quantified the importance of its nodes in the target prediction task. Hence, a node with a higher weight in the fully connected layer is more influential for the prediction of the output score.

In Fig. 3, we show the regions of interest with the 3 highest weights averaged over \( k = 2, \ldots, 15 \); underlying is the AAL parcellation atlas (Tzourio-Mazoyer et al., 2002).\(^9\) For the FIQ prediction task in the NT cohort, we see that the left superior dorsal frontal gyrus (SFG-dor.L), right superior frontal medial gyrus (SFGmed.R), and right cerebellum 6 (CRBL6.R) have the highest weights. For the VIQ prediction task in the same cohort, left hippocampus (HIP.L), left heschl gyrus (HES.L), and left cuneus (CUN.L) possess the highest weights. In the ASD cohort, the highest weights for FIQ prediction are left insula (INS.L), left calcarine cortex (CALL), and right pallidum (PAL.R), while the highest weights for VIQ prediction have the left superior frontal medial gyrus (SFGmed.L) left middle occipital gyrus (MOGL.L), and left cuneus (CUN.L).

According to our results, the more important regions of interest in IQ prediction lie in the left hemisphere of the brain. Our findings are in line with other studies, that found that the insula shows greater activity in various cognitive tasks (Critchley et al., 2000) and that the surface areal change in the left cuneus correlates strongly with full IQ, especially in perceptual tasks in young adults with very low birth weight (Skranes et al., 2013). Furthermore, we observe that the left cuneus was influential in predicting VIQ in both cohorts. Finally, as (Dryburgh et al., 2020), our experiments indicate that the middle frontal gyrus is a significant region in IQ prediction.

---

\(^8\) [https://github.com/basirilab/RegGNN](https://github.com/basirilab/RegGNN)

\(^9\) The brain networks were visualized with the BrainNet Viewer (Xia et al., 2013).
A highly interesting question for future work is to investigate why the sample selection method improves the prediction, i.e., why there seem to be clusters within the data that can be represented by central samples. This is a challenging question that most likely requires the development of new analytical tools. Nevertheless, we think that it will be worth the effort as common structures and connections between these central samples could give us a lot more insights into the interplay between the connectivity structure of the brain and cognitive ability.

4 Conclusion

In this work, we applied RegGNN, a new graph neural network, to connectome data of neurotypical subjects and subjects with autism spectrum disorder to predict full scale and verbal intelligence quotients. We trained it using a novel sample selection method, which tries to identify samples within the training set that are expected to better predict the cognitive scores of new subjects. This enabled us to train the network with only 15 samples or less, while the testing performance was on par or even better than state-of-the-art methods for cognitive score prediction from connectomes. Both the sample selection and RegGNN are easy to implement in open access software and can be used in clinical practice.

Declarations

Funding This work was funded by generous grants from the European H2020 Marie Skłodowska-Curie action (grant no. 101063403, http://basira-lab.com/normets/) to I.R. and the Scientific and Technological Research Council of Turkey to I.R. under the TUBITAK 2232 Fellowship for Outstanding Researchers (no. 118C288, http://basira-lab.com/reprime/).

However, all scientific contributions made in this project are owned and approved solely by the authors. M.A.G is funded by the same TUBITAK 2232 Fellowship. M.H. is funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany’s Excellence Strategy ? The Berlin Mathematics Research Center MATH+ (EXC-2046/1, project ID: 390685689).

Conflicts of interest/Competing interests The authors declare that they have no conflict of interest.

Ethics approval Not applicable.

Consent to participate Not applicable.

Consent for publication All authors agreed to the publication of this article.

Availability of data and material The ABIDE data that was used in this work is available online; the link is given in the text.

Code availability All methods were implemented in Python. Our RegGNN code is available at https://github.com/basiralab/RegGNN.

Author Contributions Author contributions included designing the GNN and sample selection method as well as the experimental setup (all authors), implementation of the methods and experiments (M.A.D. and M.A.G.), writing the manuscript (M.H.) and revising it critically for important intellectual content (all authors), and approval of final version to be published and agreement to be accountable for the integrity and accuracy of all aspects of the work (all authors).

A Topological Centrality Measures

Let $A$ be the (weighted) adjacency matrix of $G$, $V$ the set of vertices of $G$, and $v \in V$. The degree centrality $D(v)$ of $v$ is defined by

$$D(v) := \sum_{u \in V, w \notin v} A_{uv},$$

i.e., it assigns to each node its weighted sum of neighbors.

Let $x$ be the unit norm eigenvector of $A$ that corresponds to the largest eigenvalue $\lambda_1$ and has only non-negative entries. The eigenvector centrality $E(v)$ of $v$ is the $v$-th entry of $x$; that is,

$$E(v) := \frac{1}{\lambda_1} \sum_{w \in V} A_{vw} x_w,$$

s.t. $\|x\|_2 = 1$ and $x_w \geq 0$ for all $w \in V$.

It measures, in a relative sense, how influential a node is in the network. Intuitively, a high score means that a node has many neighbors that themselves have high eigenvector centrality scores.

Let $l_{vw}$ be the length of the shortest path between two nodes $v$ and $w$, and $n = |V|$. The closeness centrality $C(v)$ of $v$ is defined by

$$C(v) := \frac{n - 1}{\sum_{w \in V, w \neq v} l_{vw}},$$

i.e., as the inverse of the average distance of $v$ to all other nodes.

References

Abbas, K., Liu, M., Venkatesh, M., Amico, E., Kaplan, A. D., Ventresca, M., Pessoa, L., Harezlak, J., and Goñi, J. (2021). Geodesic distance on optimally regularized functional connectomes uncovers individual fingerprints. Brain Connect., 0(0):null.

Ambellan, F., Hanik, M., and von Tycowicz, C. (2021). Morphometrics: Geometric morphometrics in non-Euclidean shape spaces. https:// morphometrics.github.io/.

Arsigny, V., Fillard, P., Pennec, X., and Ayache, N. (2006). Log-Euclidean metrics for fast and simple calculus on diffusion tensors. Magn. Reson. Med., 56(2):411–421.

\[ With a slight abuse of notation, we identify nodes $v \in V$ and the integers we assign to them in order to construct $A$, e.g., we write $A_{vw}$ for the entry that corresponds to the edge between nodes $v$ and $w$.\]
Batty, G. D., Deary, I. J., and Gottfredisorn, L. S. (2007). Premorbid (early life) IQ and later mortality risk: Systematic review. Ann. Epidemiol., 17(4):278–288.

Bessadok, A., Mahjoub, M. A., and Rekil, I. (2021). Graph neural networks in network neuroscience. arXiv preprint arXiv:2106.03535.

Cole, M. W., Ito, T., and Braver, T. S. (2015). Lateral prefrontal cortex contributes to fluid intelligence through multineuronal connectivity. Brain Connect., 5(8):497–504.

Cole, M. W., Yarkoni, T., Repovš, G., Anticevic, A., and Braver, T. S. (2012). Global connectivity of prefrontal cortex predicts cognitive control and intelligence. J. Neurosci., 32(26):8988–8999.

Cole, R., Escorial, S., Shih, P. C., and Privado, J. (2007). Fluid intelligence, memory span, and temperament difficulties predict academic performance of young adolescents. Pers. Individ. Differ., 42(8):1503 – 1514.

Corso, G., Cavalleri, L., Beani, D., Lió, P., and Velicóvic, P. (2020). Principal neighbourhood aggregation for graph nets. arXiv preprint arXiv:2004.05718.

Craddock, C., Benhajali, Y., Chu, C., Chouinard, F., Evans, A., Jakab, A., Khodrakpam, B. S., Lewis, J. D., Li, Q., Milham, M., et al. (2013). The neuro preprocessor preprocessing initiative: open sharing of preprocessed neuroimaging data and derivatives. Front. Neuroinform., 7.

Critchley, H. D., Daly, E. M., Bullmore, E. T., Williams, S. C. R., Van Amelsvoort, T., Robertson, D. M., Rowe, A., Phillips, M., McAlonan, G., Howlin, P., and Murphy, D. G. M. (2000). The functional neuroanatomy of social behaviour: Changes in cerebral blood flow when people with autistic disorder process facial expressions. Brain, 123(11):2203–2212.

Dadi, K., Rahim, M., Abraham, A., Chyzhyk, D., Milham, M., Thirion, B., Varoquaux, G., Initiative, A. D. N., et al. (2019). Benchmarking functional connectome-based predictive models for resting-state fmri. NeuroImage, 192:115–134.

Deary, I. J., Spinath, F. M., and Bates, T. C. (2006). Genetics of intelligence. Eur. J. Hum. Genet., 14(6):690–700.

Deary, I. J., Strand, S., Smith, P., and Fernandes, C. (2007). Intelligence and educational achievement. Intelligence, 35(1):13–21.

Dehmamy, N., Barahasi, A.-L., and Yu, R. (2019). Understanding the representation power of graph neural networks in learning graph topology. In Wallach, H., Larochelle, H., Beygelzimer, A., d' Alché-Buc, F., Fox, E., and Garnett, R., editors, Advances in Neural Information Processing Systems, volume 32. Curran Associates, Inc.

do Carmo, M. P. (1992). Riemannian Geometry; 2nd ed. Birkhäuser, Boston, MA.

Dodero, L., Minh, H. Q., Biaggio, M. S., Murino, V., and Sona, D. (2015). Kernel-based classification for brain connectivity graphs on the Riemannian manifold of positive definite matrices. In 2015 IEEE 12th International Symposium on Biomedical Imaging (ISBI), pages 42–45.

Dryburgh, E., McKenna, S., and Rekil, I. (2020). Predicting full-scale and verbal intelligence scores from functional connectomic data in individuals with autism spectrum disorder. Brain. Imaging. Behav., 14:1769–1778.

Dubois, J., Galdi, P., Paul, L. K., and Adolphs, R. (2018). A distributed brain network predicts general intelligence from resting-state human neuroimaging data. Philos. Trans. R. Soc. B, 373(1756):20170284.

Errica, F., Podda, M., Bacciu, D., and Micheli, A. (2019). A fair comparison of graph neural networks for graph classification. arXiv preprint arXiv:1912.09899.

Faraut, J. and Korán, J. (2006). Analysis on symmetric cones. Oxford University Press, New York, USA.

Fey, M. and Lenssen, J. E. (2019). Fast graph representation learning with pytorch geometric.

Fornito, A., Zalesky, A., and Bullmore, E. (2016). Fundamentals of brain network analysis. Academic Press.

Deary, I. J., Spinath, F. M., and Bates, T. C. (2006). Genetics of intelligence predicts health and longevity, but why? Curr. Dir. Psychol. Sci., 15(1):1–4.

Gray, J. R., Chabris, C. F., and Braver, T. S. (2003). Neural mechanisms of general fluid intelligence. Nat. Neurosci., 6(3):316–322.

Hagberg, A. A., Schult, D. A., and Swart, P. J. (2008). Exploring network structure, dynamics, and function using networkx. In Varoquaux, G., Vaught, T., and Millman, J., editors, Proceedings of the 7th Python in Science Conference, pages 11 – 15. Pasadena, CA, USA.

He, T., Kong, R., Holmes, A. J., Nguyen, M., Sabuncu, M. R., Eckhoff, S. B., Bzdok, D., Feng, J., and Yeo, B. T. (2020). Deep neural networks and kernel regression achieve comparable accuracies for functional connectivity prediction of behavior and demographics. NeuroImage, 206:116276.

Jiang, R., Calhoun, V. D., Fan, L., Zuo, N., Jung, R., Qi, S., Lin, D., Li, J., Zhuo, C., Song, M., et al. (2020). Gender differences in connectome-based predictions of individualized intelligence quotient and sub-domain scores. Cereb. Cortex, 30(3):888–900.

Kim, J., Kim, T., Kim, S., and Yoo, C. D. (2019). Edge-labeling graph neural network for few-shot learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 11–20.

Kingma, D. P. and Ba, J. (2017). Adam: A method for stochastic optimization.

Kipf, T. N. and Welling, M. (2017). Semi-supervised classification with graph convolutional networks.

Mcdaniel, M. (2005). Big-brained people are smarter: A meta-analysis of the relationship between brain volume and intelligence. Intelligence, 33:337–346.

Moakher, M. (2005). A differential geometric approach to the geometric mean of symmetric positive-definite matrices. SIAM J. Matrix Anal. Appl., 26(4):735–747.

Pamplona, G. S. P., Santos Neto, G. S., Rosset, S. R. E., Rogers, B. P., and Salman, C. E. G. (2015). Analyzing the association between functional connectivity of the brain and intellectual performance. Front. Hum. Neurosci., 9:61.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. J. Mach. Learn. Res., 12:2825–2830.

Pennecc, X., Fillard, P., and Ayache, N. (2006). A Riemannian framework for tensor computing. Int. J. Comput. Vis., 66(1):41–66.

Qi, X., Liao, R., Jia, J., Fidler, S., and Urtasun, R. (2017). 3D graph neural networks for rgbd semantic segmentation. In Proceedings of the IEEE International Conference on Computer Vision, pages 5199–5208.

Qu, M., Bengio, Y., and Tang, J. (2019). Gmnn: Graph markov neural networks. In International conference on machine learning, pages 5241–5250. PMLR.
Reichenberg, A., Mill, J., and MacCabe, J. H. (2009). Epigenetics, genomic mutations and cognitive function. *Cogn. Neuropsychiatry*, 14(4-5):377–390.

Reiss, A. L., Abrams, M. T., Singer, H. S., Ross, J. L., and Denckla, M. B. (1996). Brain development, gender and iq in children: a volumetric imaging study. *Brain*, 119(5):1763–1774.

Sarlin, P.-E., DeTone, D., Malisiewicz, T., and Rabinovich, A. (2020). Superglue: Learning feature matching with graph neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4938–4947.

Schlichtkrull, M., Kipf, T. N., Bloem, P., Van Den Berg, R., Titov, I., and Welling, M. (2018). Modeling relational data with graph convolutional networks. In *European semantic web conference*, pages 593–607. Springer.

Shen, X., Finn, E., Scheinost, D., Rosenberg, M., Chun, M., Papademetris, X., and Constable, R. (2017). Using connectome-based predictive modeling to predict individual behavior from brain connectivity. *Nat. Protoc.*, 12(2):506–518.

Skranes, J., Løhaugen, G. C., Martinussen, M., Häberg, A., Brubakk, A.-M., and Dale, A. M. (2013). Cortical surface area and iq in very-low-birth-weight (vLBW) young adults. *Cortex*, 49(8):2264–2271.

Sporns, O., Tononi, G., and Kötter, R. (2005). The human connectome: A structural description of the human brain. *PloS Comput. Biol.*, 1(4).

Tordjman, S., Cohen, D., Anderson, G., Bothol, M., Cantonno, R., Coulon, N., and Roubertoux, P. (2018). Repint of “reframing autism as a behavioral syndrome and not a specific mental disorder: Implications of genetic and phenotypic heterogeneity”. *Neurosci. Biobehav. Rev.*, 89:132–150.

Tzourio-Mazoyer, N., Landeau, B., Papathanassiou, D., Crivello, F., Etard, O., Delcroix, N., Mazoyer, B., and Joliot, M. (2002). Automated anatomical labeling of activations in spm using a macroscopic anatomical parcellation of the mni mri single-subject brain. *NeuroImage*, 15(1):273–289.

Wong, E., Anderson, J. S., Ziellinski, B. A., and Fletcher, P. T. (2018). Riemannian regression and classification models of brain networks applied to autism. In *Connectomics in NeuroImaging*, pages 78–87, Cham. Springer International Publishing.

Woolgar, A., Farr, A., Cusack, R., Thompson, R., Nimmo-Smith, I., Torralva, T., Roca, M., Antoun, N., Manes, F., and Duncan, J. (2010). Fluid intelligence loss linked to restricted regions of damage within frontal and parietal cortex. *Proc. Natl. Acad. Sci. U.S.A.*, 107(33):14999–14902.

Yamin, M. A., Tessadori, J., Akbar, M. U., Dayan, M., Murino, V., and Sona, D. (2020). Geodesic clustering of positive definite matrices for classification of mental disorder using brain functional connectivity. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pages 1–5.

You, K., and Park, H.-J. (2021). Re-visiting Riemannian geometry of symmetric positive definite matrices for the analysis of functional connectivity. *NeuroImage*, 225:117464.

Yun, S., Jeong, M., Kim, R., Kang, J., and Kim, H. J. (2019). Graph transformer networks. *arXiv preprint arXiv:1911.06455.*

Zhang, C., Song, D., Huang, C., Swami, A., and Chawla, N. V. (2019). Heterogeneous graph neural network. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 793–803.
Table 2: Comparison of regression methods on the NT cohort. The best performing method for each architecture is bold while the second best is underlined. The mean ± standard deviation as well as minima and maxima over k = 2, . . . , 15 (in brackets) are given. The overall best performing method according to mean error and the best sample selection performance are indicated in blue. Abbreviations are: (a) absolute Euclidean distance, (g) geometric Log-Euclidean distance, (tm) full tangent matrix, (dc) degree centrality, (ec) eigenvector centrality, (cc) closeness centrality, (cnu) concatenation unscaled, (cns) concatenation scaled.
### Table 3: Comparison of regression methods on the ASD cohort. The best performing method for each architecture is bold while the second best is underlined. The mean ± standard deviation as well as minima and maxima over \( k = 2, \ldots, 15 \) (in brackets) are given. The overall best performing method according to mean error and the best sample selection performance are indicated in blue. Abbreviations are: (a) absolute Euclidean distance, (g) geometric Log-Euclidean distance, (tm) full tangent matrix, (dc) degree centrality, (ec) eigenvector centrality, (cc) closeness centrality, (cnu) concatenation unscaled, (cns) concatenation scaled.

| Method | Mean MAE ± std (min, max) | Mean RMSE ± std (min, max) |
|--------|-----------------------------|-----------------------------|
| CPM    | 11.161                      | 13.901                      |
| CPM (a) |
| CPM (g) |
| CPM (tm) |
| CPM (dc) |
| CPM (cc) |
| CPM (cnu) |
| CPM (cns) |
| PNA-S  | 11.455                      | 14.000                      |
| PNA-S (a) |
| PNA-S (g) |
| PNA-S (tm) |
| PNA-S (dc) |
| PNA-S (cc) |
| PNA-S (cnu) |
| PNA-S (cns) |
| PNA-V  | 11.196                      | 12.148                      |
| PNA-V (a) |
| PNA-V (g) |
| PNA-V (tm) |
| PNA-V (dc) |
| PNA-V (cc) |
| PNA-V (cnu) |
| PNA-V (cns) |
| RegGNN | 11.952                      | 14.001                      |
| RegGNN (a) |
| RegGNN (g) |
| RegGNN (tm) |
| RegGNN (dc) |
| RegGNN (cc) |
| RegGNN (cnu) |
| RegGNN (cns) |
| CPM    | 11.171                      | 13.844                      |
| CPM (a) |
| CPM (g) |
| CPM (tm) |
| CPM (dc) |
| CPM (cc) |
| CPM (cnu) |
| CPM (cns) |
| PNA-S  | 11.952                      | 14.001                      |
| PNA-S (a) |
| PNA-S (g) |
| PNA-S (tm) |
| PNA-S (dc) |
| PNA-S (cc) |
| PNA-S (cnu) |
| PNA-S (cns) |
| PNA-V  | 11.196                      | 12.148                      |
| PNA-V (a) |
| PNA-V (g) |
| PNA-V (tm) |
| PNA-V (dc) |
| PNA-V (cc) |
| PNA-V (cnu) |
| PNA-V (cns) |
| RegGNN | 11.952                      | 14.001                      |
| RegGNN (a) |
| RegGNN (g) |
| RegGNN (tm) |
| RegGNN (dc) |
| RegGNN (cc) |
| RegGNN (cnu) |
| RegGNN (cns) |