

DIPLOMA THESIS

LINKING CONNECTIVITIES AND GENE EXPRESSION PATTERNS IN THE MOUSE BRAIN

September 30, 2022

Tilman Hinnerichs
Matrikelnummer: 4643427
Technische Universität Dresden

Tutor: Dr. Nico Scherf

MPI for CBS

Summer semester 2022

proper title?

Abstract

Contents

1	Introduction	1
2	Literature overview	3
2.1	Gene expression databases and prediction	3
2.2	Finding spatial patterns in gene expression in mice brains	4
2.3	Structural, functional and effective connectivity prediction	5
3	Materials and methods	8
3.1	Problem description	8
3.1.1	Spatial gene expression prediction	8
3.1.2	Preserving dimensionality reduction in brains	8
3.1.3	Structural, functional and effective connectivity prediction	8
3.2	Datasets and preprocessing	9
3.2.1	Spatial gene expression values in mouse brains	10
3.2.2	Protein-protein interaction graph	14
3.2.3	Connectivity data	14
3.2.3.1	Structural connectome data	15
3.2.3.2	Functional connectivity data	15
3.2.3.3	Effective connectivity	17
3.3	Model	18
3.3.1	Feature generation and protein representation	18
3.3.2	Graph convolutional neural networks	19
3.3.2.1	Graph convolutional neural layers (GCNConv)	20
3.3.2.2	Graph attention networks (GATConv)	21
3.3.2.3	Training deeper graph convolutional neural networks (GENConv)	22
3.3.2.4	Graph Kernels going beyond the Weisfeiler-Leman algorithm (KerGNN)	23
3.3.3	Dimensionality reduction techniques	24
3.3.3.1	Principal component analysis (PCA)	24
3.3.3.2	t-SNE	25
3.3.3.3	UMAP	25
3.3.3.4	Parametric UMAP	26
3.3.4	Hyperparameter tuning	27
3.4	Evaluation and metrics	27
4	Results	30
4.1	Gene expression prediction	30
4.2	Dimensionality reduction and its combination with different graphs structures	32
4.2.1	On the validity of EMBSIM and EMBVAR	32
4.2.2	Enhancing neural embedding methods with GCNs	34
4.3	On the linkage of connectivities and gene expression patterns	40

5 Discussion	43
5.1 Evaluating different feature types for gene expression prediction	43
5.2 Gene expression patterns and their relation to GCNs	43
5.3 Connectivity prediction	44
6 Conclusion	45

To be sorted somewhere

- Variability and different interpretations of different graph convolutional neural filters (Kipf and Welling, 2016; Li et al., 2020; Feng et al., 2022) etc.
- DeepGOPlus for feature generation (Kulmanov and Hoehndorf, 2019)
- discussion of different PPI network databases (Szklarczyk et al., 2014)
- discussion of potential databases associating gene expression data with their spatial distribution (Hawrylycz et al., 2011)
- discussion of best neural learning/graph convolutional methods (Paszke et al., 2019; Fey and Lenssen, 2019)
- how to handle highly imbalanced data, metrics, preprocessing, sampling, modification of loss function (Jeni et al., 2013a) and optimization over them (with Adam(Kingma and Ba, 2015))
- maybe introduction of PhenomeNET for MP/GO for more sophisticated protein representation (Hoehndorf et al., 2011; Ashburner et al., 2000; Carbon et al., 2020; Smith and Eppig, 2009) and derive features from DL2vec (Chen et al., 2020; Mikolov et al., 2013)
- evaluation of „Using ontology embeddings for structural inductive bias in gene expression data analysis“(Trebacz et al., 2020)
- take some ideas from Zitnik and Leskovec (2017) with title „Predicting multicellular function through multi-layer tissue networks“. (OhmNet)
- potentially group results based on InterPro(Blum et al., 2020) families eventually
- choice of model organism?!

1 Introduction

General thread for introduction and motivation:

- Gene expression patterns are difficult to analyze in humans → take mouse as model organisms
- why do we study sane mice and not a
- The brain is a multi-level system in which the high-level functions are generated by low-level genetic mechanisms. Thus, elucidating the relationship among multiple brain levels via correlative and predictive analytics is an important area in brain research. Currently, studies in multiple species have indicated that the spatiotemporal gene expression patterns are predictive of brain wiring. Specifically, results on the worm *Caenorhabditis elegans* have shown that the prediction of neuronal connectivity using gene expression signatures yielded statistically significant results.
- no in-depth analysis of mouse brain genetic patterns and their relation to different connectivity patterns has been made yet
- Why are we concerned with gene expression prediction and what could it tell us?
 - (Twine et al., 2011) show the importance of gene expression patterns, by linking gene expression aberration with increase in Alzheimer's disease
 - studies have shown circadian patterns of gene expression in human brain and the disruption of those in depressive disorder (Li et al., 2013)
 - first understand sane brain and its circuits, before tackling pathological data
- why is finding (low-dimensional) patterns important here?
- What is structural and functional connectivity and what are associated hypotheses?
 - (Fornito et al., 2015) elaborate on the connectomics of brain disorders and its complexity in connectivity. Understanding how brain networks respond to pathological perturbations is crucial for understanding brain disorders and behavior
- Why is finding a link or to connectivity from gene expression desirable?
- Why do we think that GCNs could help finding such patterns?
 - Guilt by association over gene networks (Oliver, 2000; Gillis and Pavlidis, 2012) in genetic networks
 - protein function prediction from PPI networks (Vazquez et al., 2003)
 - GCNs have been applied successfully to variety of tasks over different types of graphs.
- what are our contributions?
 - Showed that graph convolution over PPI graphs helps finding patterns in gene expression data

- Contributed an implementation of our method
- contributed an implementation of KerGNN to PyTorch Geometric
- built an open framework for parametric UMAP in torch to integrate non-UMAP graphs
- what is the outline of this script?
- implementation done in Python, pytorch and Pytorch Geometric and available at Github link

General Introduction of the Research Study

Research problem or Questions with Sub-Questions

Reasons or Needs for the Research Study/Motivation for my research

Definition and explanation of Key Terminology

Context of Research Study within th Greater Discipline

- Introduction to mouse brains as model organisms for insights into human brain
- Works on mouse brain in general and potential tasks
- works on gene expression in mouse brains
 - traditional approaches
 - importance of gene expression patterns in mouse brains
- neural networks for this purpose
 - how were
- gene expression for general tissue

2 Literature overview

2.1 Gene expression databases and prediction

Research in gene expression prediction and profiling has a long history in bioinformatics and systems biology, but was almost exclusively linked to cancer research. Moreover, with the rise of machine learning, and more specifically (deep) neural networks and its variants, this field became increasingly data reliant. The Human Genome Project (Watson, 1990), launched in 1990 and declared finished in 2003 while the first gapless assembly was finished in 2022, also sparked various works in relating these genetic representations to other tissue- and individual-specific properties and traits.

For comparison of gene expression profiling works there exist multiple prominent variables. Most significantly, the chosen organism is a crucial choice for both data availability and predictive complexity. Second, the chosen tissue is naturally important for the proposed hypotheses, especially with respect to tissue definitive cancer research, and its potential ability to generalize without transfer learning. While gene expression pattern analysis approaches frequently focus on tissues like *mamma*, (primarily female) breast, (Herschkowitz et al., 2007), liver (Flores-Morales et al., 2002), and skeletal muscle (Lecker et al., 2004) for exploration of diseases like cancer and atrophy, respectively, in humans.

However, the nervous system is often investigated separately as it bears different molecular processes and structure, anatomy and cell life cycles, while brain and spinal cord are even based in a separate nutritional circuit for mammals. Moreover, gene expression determination in the human brain may almost certainly remain an deadly intervention for most brain tissues, hence allowing only for careful extraction of specific tissues in living organisms. Also this disallows for *in-vivo* extraction of vital brain regions and structures, e.g. the brainstem. Furthermore, the human brain's gene expression patterns are varied and diversified (Ramasamy et al., 2014), aligning with its anatomical and embryogenesis complexity, and its compartments are exceptionally and deeply connective and collaborative (Fornito et al., 2015). Both also hold for invertebrates, i.e. insects. Thus, full genetic profiles of expression are mandatory for a full understanding of the mammalian and invertebrate brain and primary nervous system, respectively. By the strong intervention of the tissue extraction, full genome atlases are fit together from various experiments on multiple individuals.

The human brain is among the most intricate and complicated networks we do know of, and is far from being fully understood. Additionally, full transcriptomic atlases of human brains are difficult to collect while raising decisive privacy concerns. Yet, there were multiple efforts and projects with rather small sample sizes. A detailed elaboration on dataset and organism choice, and their respective properties may be found in Section 3.2.

However, there have been works on numerous works for other tissues and other organisms. modENCODE Consortium et al. (2010) correlate activity patterns in the regulatory network within *Drosophila*, proposing their model for identification of functional elements "modENCODE". As this work was published back in 2010, the approach relies purely on statistical correlation and covariance. Chikina et al. (2009) follow a similar approach in *C. elegans* predicting tissue-specific gene expression in 2006 utilizing support-vector machines (SVM)(Noble, 2006).

More modern, data-oriented machine learning models such as (deep) neural networks (NN) were applied successfully to similar problems. Aromolaran et al. (2020) achieved to predict essential genes based on their respective sequence and functional features profiting off NNs, while transcriptomic interaction prediction was done based on functional gene data using deep learning in Yang et al. (2019) in *Drosophila* over different tissues.

Within humans, as mentioned previously, gene expression was primarily used for cancer and disease research. Schulte-Sasse et al. (2021) and Wang et al. (2021) were the first to apply graph convolutional neural networks to the task of gene expression prediction within humans. While Schulte-Sasse et al. (2021) was applied on data from The Cancer Genome Atlas (TCGA)(Tomczak et al., 2015) across multiple tissues, Wang et al. (2021)'s MOGONET is proposed as a general framework for gene expression prediction with example computations on ROSMAP dataset and TCGA. Both approaches implement the original formulation of GCNs(Kipf and Welling, 2016), which we will discuss in more detail in Section 3.3.2, over protein-protein interaction networks and accomplish outstanding performances and both measure biomarker importance for prediction in order to leverage explainability. The authors thereby exploit the "guilt by association" principle (Oliver, 2000; Gillis and Pavlidis, 2012) over gene networks, adding background knowledge such as biological interaction and pathways.

Crucial for almost all classification tasks in machine learning is the choice of entity representation. In the mentioned works molecular (Schulte-Sasse et al., 2021; modENCODE Consortium et al., 2010; Noble, 2006) and phenotypical (Wang et al., 2021; Chikina et al., 2009) features were used for expression prediction, but never both combined. The combination of phenotypical and molecular features over GCNs was proven to raise predictive performance in drug-target interaction prediction (Hinnerichs and Hoehndorf, 2021) but remains an open challenge for this very task.

2.2 Finding spatial patterns in gene expression in mice brains

In this subsection we will constrain the issue of gene expression analysis to both "spatial patterns", mammals and the tissues of the brain, which we study in this work. The term *spatial patterns* is rather vague and allows for various interpretations, both discrete and continuous, which will form the classes for the following literature review.

In Pavlidis and Noble (2001) is the first first review paper on regional variation in genetic expression in mouse brain, up to our knowledge. Zapala et al. (2005) is also among the earliest works, showing that local structures beared "transcriptional imprint" that coincide with the embryological origin of the examined regions. However, they only were able to identify up to 24 neural tissues. They further conclude that this may be important for functional collaboration within the adult mouse brain. The authors measure pairwise correlation show the existence of clusters over a heatmap.

The Allen Institute Brain Atlas (AIBA), is a collection various atlases such as Allen Mouse Brain Atlas (AMBA)(Lein et al., 2006; Daigle et al., 2018), the Allen Mouse Brain Connectivity Atlas (AMBCA) (Oh et al., 2014; Harris et al., 2019) and the Allen Mouse Brain Common Coordinate Framework (CCFv3) (Wang et al., 2020) to name only the ones related to adult mice's brains. As it was the first coherent collection of spatially resolved expression values, mapping 2D expression

images consistently to 3D coordinates, the AMBA has sparked a range of publications. Within Lein et al. (2006), the Allen Institute also published the "Allen Reference Atlas"(ARA) proposing a number morphological and histologically induced sub-regions of the brain and hence a precisely defined parcelation. Moreover, they propose the ARA *ontology*, a semantic hierarchy, providing a hierarchical cluster of all sub-structures and map them back to their coordinates with the CCF.

Bohland et al. (2010) advance clustering of such expressions under usage of singular value decomposition (SVD) within mice, combined with an extensive analysis of similarities to neuroanatomy. Likewise, Takata et al. (2021) propose a flexible annotation atlas of the mouse brain, introducing a flexible ontology construction framework which may be used on the transcriptomic data such as the AMBA, leveraging anatomic structure and axonal projection data. Here, FAA focuses on consistent and reproducible regions-of-interest (ROIs) definition for other downstream tasks such as resting-state functional connectivity annotation. Further, this ontology may be seen as a pattern within mouse brain, while it may only detect connected structures.

The authors of Valk et al. (2020) analyze structural covariance of cortical thickness within primate brains, namely macaques, and its correlation to each cortical layers transcriptome. Further, transcriptomic variation was related to a continuum of functions by mapping them the brain anatomy, inducing a *continuous*, functional parcelation of the primates brain. Further, this study suggests a relation of functional and transcriptomic links. Similarly, Zeng et al. (2015) propose a range of deep learning methods for capturing spatiality of gene expression within the mouse brain. A notable addition is the work of Kelly and Black (2020), presenting an R package for simulating gene expression from graph structures over general biological pathways, that may be and was applied to (mammalian) brains prospectively.

While also focused on mouse brains, Partel et al. (2020) submits a novel database based on their own *in-situ* sequencing data, and a consecutive spatial gene expression analysis pipeline, and relates the results to tissue morphology and hence indirectly to the AMBA. Similarly to our proposed approach, brain parcelation are present as *n*-dimensional, continuous embeddings, representing closeness in gene expression space. Due to the similarity in the pipeline especially in their visualization utilizing UMAP, we will use the generated images of this work for a brief comparison in Section 4.

Up until now, GCNs were not applied to this issue.

2.3 Structural, functional and effective connectivity prediction

In this section we will examine related work on brain connectivity prediction from transcriptomic data. We hereby separate axonal and functional connectivity due to their differing associated hypotheses.

The relation of gene expression patterns and *structural* connectivity was studied numerous times over various model organisms, especially *C. elegans*, *Mus musculus*, but also humans. We will categorize existing literature with respect to the underlying organism.

Kaufman et al. (2006) and Varadan et al. (2006) were among the earlier works on this research field, followed by Arnatkevičiūtė et al. (2018), showing the relation of axonal connectivity and gene expression within *C. elegans*. Further, that relation was shown by Rubinov et al. (2015), Fakhry et al. (2015) and Fulcher and Fornito (2016) for the mouse brain, while Parkes et al. (2017) and Goel et al.

Name a few
"sparked"
research
works on
this

(2014) proved a correlation within human brains.

Fakhry and Ji (2015) is among the earlier works focusing on the predictive power across different mouse brain regions. They applied non-machine learning, computational models for axonal connectivity prediction in adult mouse brains. Similarly, Roberti et al. (2019) uses transcriptomic information to anatomical connectivity patterns and gene expression of neurons using (shallow) neural networks. Yield a 85% accuracy in prediction of unconnected and connected regions. Both shall serve as a baseline performance in chapter 4. Only recently, Wang et al. (2022) proposed a novel-network based method integrating molecular-based gene association networks such as protein-protein interaction networks with brain connectome data. They further link these gene expression patterns to four brain diseases, including Alzheimer's disease, Parkinson's disease, major depressive disorder and autism.

The correlation of *functional* connectivities and transcriptomic data is much more complex in nature than the previous task. We will again classify approaches by their respective model organism.

Whitfield et al. (2003) were one of the first to link transcriptomic data with behavior and hence functional patterns in individual honey bees back in 2003. The authors show that changes in the messenger RNA were connected to behavior and how changes to RNA directly influenced the other. Rankin (2002) first developed the idea of combining behavioral analyses of *C. elegans* with their genetics. Further, Sun and Hobert (2021) only recently described the distinct functional states and the corresponding distinct molecular states within the transcriptome. While honey bees and nematodes are rather simple model organisms, enabling both full transcriptomic analyses of the organisms, and their bearing and actions. However, "behavior" may be ambiguous and vague for such taxonomically distant animals, from the viewpoint of humans, and may only be linked to very basic meta-tasks such as basic routing, orientation and basic social interaction.

Research on humans further indicates correlation of transcriptomic patterns and "neural dynamics", concluded from e.g. fMRI data (Richiardi et al., 2015; Diez and Sepulcre, 2018; Vértes et al., 2016) or electrocorticography (Betzel et al., 2019). We refer to Fulcher et al. (2021) for an extensive overview on the link to axonal and functional connectivity. Further, Zerbi et al. (2021) propose a computational model calibrated over 16 autistic mouse models, that reveals a range of functional connectivity subclasses and -types, based on spatial gene expression.

Third, we study the relation of *effective connectivity* and genetic expression patterns. While effective differs from functional connectivity (Bauer et al., 2018), there is only little study on the influence of gene expression on causal relations. The rather point-wise studies such as Tan et al. (2012) show that disturbances in e.g. *COMT*, *AKT1* and *DRD2* are related to variation in effective connectivity. Moreover Hamida et al. (2018) showed that *Gpr88*-knockout is related to changes to effective connectivity in the corticolimbic system. While these works motivate only little experiments towards this relation, one of the most well-renowned researcher on effective connectivity, Karl J. Friston, wrote in Friston (2002) on "The Disconnection Hypothesis". It states that pathological cognitive behavior, is expressed in abnormal, especially *effective*, connections. These however, are determined by (1) structural plasticity, i.e. neurogenesis and gene expression, (2) synaptic plasticity, i.e. functional connectivity, which is in return linked to the transcriptome (see above paragraph on FC). Friston (2011) further provides a splendid overview over the relation of functional and effective connectivity. Lepperød

et al. (2018) run a first computational model for describing effective connectivities in brains.

Gene
expression
and effective
connectivity

Brief Overview of Literature Reviewed, Discussed and applied

Study Model and Process Aligning with literature reviewed

Hypotheses and justifications tied to prior sections and statements

The Scope of the study with theoretical assumptions and limitations

3 Materials and methods

In this study, we utilized and incorporated various approaches from other works and applied them to diverse datasets. The following section will give a brief overview over all modules of the proposed models, while the entire computational methods will be presented and described in the results section (Section 4). We further introduce the goals and scopes of our respective research questions and on our evaluation metrics for this purpose.

3.1 Problem description

Here we give a brief introduction to each of the three tackled issues and further summarize data properties, challenges and goals of each problem in this section and Section 3.2.

3.1.1 Spatial gene expression prediction

Firstly, the issue of spatial gene expression prediction is concerned with the following problem: Within a given structure or at a specific coordinate, and for a given gene, we want to determine whether the latter is expressed or not. While there are also ways to quantify the expression within a region, we only care about the *quality*, i.e. whether the is expressed or not. We treat all structure-gene pairs without a known expression as negatives, thus handling the dataset in a closed-world manner, and accordingly formulate the problem as a binary classification task. Naturally, as shown in Section 2.2, mammalian brains show a high correlation of gene expression and neuroanatomic substructures, suggesting importance of nearby and adjacent structures to the considered one. While spatial prediction and conditionals are hard to infuse into models, we will therefore start by predicting gene expression within single structures, constructing train and validation set over genes, or train from related or proximate regions.

3.1.2 Preserving dimensionality reduction in brains

The second studied issue is the task of preserving dimensionality reduction. Here, all regions or 3D-voxels are associated with a vector embedding of fixed dimensionality n , representing each section with the features of choice. As we are concerned with the influence and patterns of gene expression we will only use and consider embeddings of transcriptome for this task. The eventual task is to find a mapping $f_{\text{emb}} : \mathbb{R}^n \rightarrow \mathbb{R}^k$ with $k < n$ reducing each structure's representation dimensionality to k , such that the relative, pairwise distances are preserved. Hence, if two sections share similar expressions, they should be described similarly in \mathbb{R}^k , invariant to spatial distance. As we seek to visualize such embeddings for quality assessment of the embeddings and characterization of marked regions, a mapping into color-space, i.e. choosing $k \in \{1, 2, 3\}$, appears natural.

3.1.3 Structural, functional and effective connectivity prediction

Eventually, we want to forecast the brain's connectome in our third formulation: Given two structures we want to predict whether there is a connection either of axonal or functional or both types. Therefore, similarly to the question of dimensionality, regions and voxels, respectively, are represented by their

expression characteristics in vector space. Likewise to other works, introduced and explained in Section 4, and to the issue described in 3.1.1, our analysis is invariant to the eventual "strength" of the connection, but focuses on the quality of connectivity. Hence, we define a cut-off threshold converting the issue to a binary classification task, whereas non-positive and unknown links are treated as negatives.

Assumptions of study method and study design with implied

3.2 Datasets and preprocessing

As human brains are among the most complex in structure and connectivity within nature, a full transcriptomic atlas may be very valuable for the research community and our experiments in this work. However, full transcriptomic atlases of homo sapiens are ethically difficult to gather. Additionally, as a valuable, public genetic atlas of deceased relatives may provide highly critical information about the remaining, living ones, such as genetic diseases, genetic markers for correlating with addiction and other social behavior, or ancestry in general, this raises tremendous privacy concerns. As we aim to investigate transcriptomic patterns in the brain and their relation to structural and functional connectivity as a generalized, organism-invariant methodology, we also want our experiments to be as understandable and replicable as possible. However, there have been multiple initiatives towards collaborative and open human brain data, such as the Allen Human Brain Atlas (AHBA) (Hawrylycz et al., 2011) also published by the Allen Institute and the Human Brain Atlas (HBA) (Roland et al., 1994). While both are almost complete, e.g. AHBA considers over 20000 genes, but these were collected from just 6 human individuals. In combination with the Human Connectome Project (HCP) (Van Essen et al., 2013) this atlas provides a valuable, matched data resource. As rodents, and more specifically mice, are more simplistic and well studied in behavior and due to their taxonomic proximity to humans serve as model organisms for diverse genetic, social and medical experiments, we opted for mice as the study organism. The ultimate goal still shall be the further understanding of brains of our species.

Furthermore, immense effort was put into enormous projects and databases for model invertebrates, namely *Drosophila* (specifically *Drosophila melanogaster*, also called *fruit fly*) and *Caenorhabditis elegans* (short: "*C. elegans*", colloquially also called *roundworm*) with the two projects "Virtual Fly Brain" (Milyaev et al., 2012) (<https://virtualflybrain.org/>) and "Wormbase" (Lee and Sternberg, 2003; Davis et al., 2022) (<https://wormbase.org>), respectively. Yet, we wanted to stay within the same taxonomic phylum leading our choice towards mouse brains.

We further use five different graphs in our setup, which shall be described and our choice motivated in the following sections:

1. Structure – gene expression data and the corresponding structure ontology/hierarchy,
2. Protein-protein interaction graph,
3. Structural connectivity and axonal projection data,
4. Functional connectivity data, and

5. Effective connectivity data.

We provide visualizations and plots for each graph to provide a better intuition respectively. Links to all datasets and download scripts are available in the <https://github.com/neural-data-science-lab/Treasure-Gene-expression-regions> Github repository.

3.2.1 Spatial gene expression values in mouse brains

The AMBA(Lein et al., 2006) is a genome-wise and comprehensive, digital 3D-map of spatial gene expression of the adult mouse central nervous system (CNS). AMBA utilizes *in situ* hybridization (ISH) for expression measurement and is publicly and freely available. In literature there are several approaches, compromising sequencing depth, accuracy, throughput and spatial resolution. Generally, one can identify two classes of gene expression measurement while preserving spatial information. The first approach is to store spacial coordinates first, followed by a the sequencing of single-cell RNA where Achim et al. (2015) and Chen et al. (2017) propose the mapping and the Geo-seq protocol for this method. The second method includes the usage of "barcodes", decoded in the tissue sample, while running a parallel analysis of numerous mRNAs (Ke et al., 2013; Moffitt et al., 2016). Here, AMBA and Partel et al. (2020)'s dataset are constructed using the first and the latter method, respectively. Sample extraction is very invasive and hence samples are collected from multiple individuals. Please see Ng et al. (2007) for more information and an overview on applied 3D reconstruction and registration of ISH images.

In AMBA over 17000 genes were measured in voxels in resolutions of $10\mu m$, $25\mu m$, $50\mu m$ or $100\mu m$. An "average brain", i.e. averaging expression intensities of each voxel among all genes, is shown in Figure 1a and 1b as coronal and sagittal cross-sections. The most important supportive dataset for AMBA is the Common Coordinate Framework Version 3 (CCFv3)(Wang et al., 2020) structuring and mapping all brain voxels into identifiable sub-structures, based on their morphology. Unfortunately, functional connectivity (FC) measurement is often measured with respect to regions-of-interest (ROIs) that may coincide with these structures. Therefore, full brain FC datasets on voxel level may not be publicly available in the same resolution. Hence, for a full integration of all datasets, we will use the CCFv3 grouped voxel for structures, thus redefining all three problem descriptions from voxel to structure level, i.e. a set of voxels. A visualization of these structures based on the CCFv3 annotation volume is depicted in Figure 1c and 1d.

Expression values in AMBA are provided per structure as expression densities, intensities and energies. Let g be the considered gene, d the considered division or substructure, $P_{g,d}$ be the set of pixels within an ISH image and $P_{g,d,e}$ the set of expressing pixels, displaying expression of g in d . Both $P_{g,d}$ and $P_{g,d,e}$ are given as measured expression intensities for the respective pixels. Thus, the three measurements are defined as shown in Table 1.

As expression energy provides an in-division normalization within itself, is invariant to structure size and is dependent on the actually measured intensities, it appears to be the natural choice. It was further used by all previously mentioned studies over AMBA as ground-truth; we will hence do likewise, and thus use expression value and expression energy interchangeably.

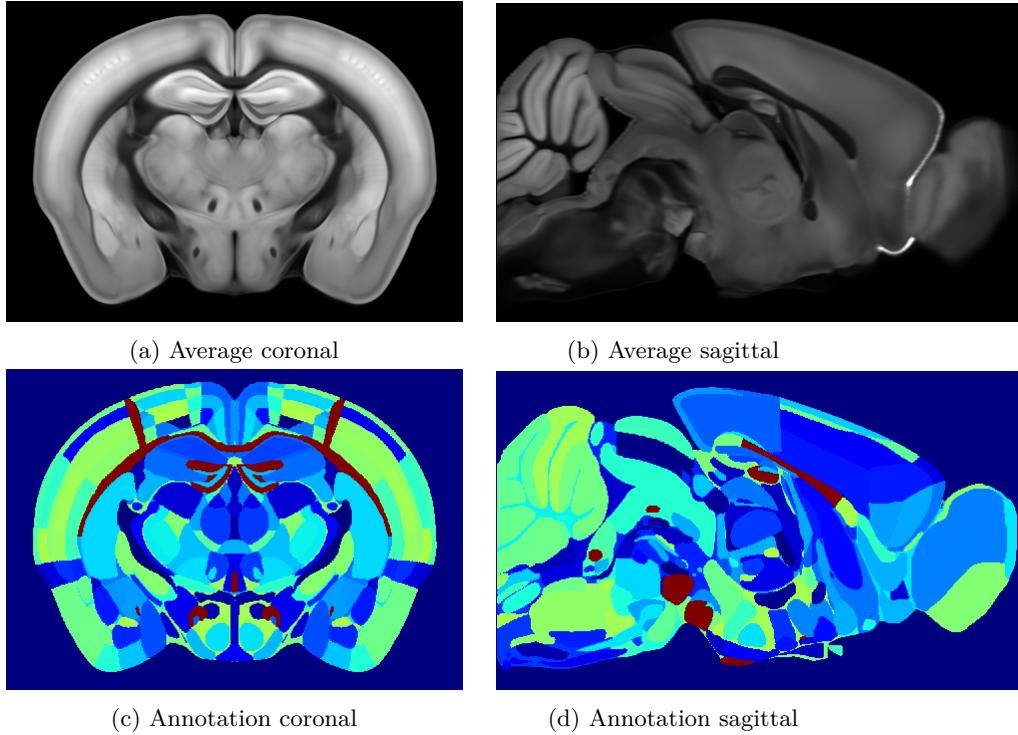


Figure 1: (a)-(b): Coronal and sagittal view on the "average brain", an averaged map of all gene expression values gathered from AMBA; (c)-(d): Coronal and sagittal view on the annotation atlas, partitioning the reference brain into ≈ 900 sub-structures.

Metric name	Formula	Description
Expression density	$E_d(g, d) := \frac{ P_{g,d,e} }{ P_{g,d} }$	sum of expressing pixels / sum of all pixels in division
Expression intensity	$E_i(g, d) := \frac{\sum_{p_e \in P_{g,d,e}} p_e}{ P_{g,d,e} }$	sum of expressing pixel intensity / sum of expressing pixels
Expression energy	$E_e(g, d) := \frac{\sum_{p_e \in P_{g,d,e}} p_e}{ P_{g,d} }$	sum of expressing pixel intensity / sum of all pixels in division

Table 1: Formulas and descriptions for the calculation of the three expression values provided by AMBA. The respective descriptions are provided by the official Allen Mouse Brain Documentation, which we note as a reference for further information on these metrics.

We perform a gene-set enrichment analysis (GSEA) on the expression values. GSEA is a method facilitating recognition of classes that are expressed frequently, and thus may be over-represented in the expression data. We hereby follow the works of Subramanian et al. (2007) and Kuleshov et al. (2016) for this purpose, and integrate them into our preprocessing pipeline.

With S the set of structures and G the set of genes, we are now able to construct a matrix $M_{GE} \in \mathbb{R}^{S \times G}$. For simplicity, we will only consider structures with at least one non-zero gene expression value and genes that have a non-zero expression energy in at least one structure. This leaves us with $|S| = 843$ structures and $|G| = 16679$ genes. Note further, that the measured pixel-wise intensities are not normalized and hence the matrix entries, i.e. expression energies, may not be within the interval $[0, 1]$. Hence, we will first normalize the matrix. However, the briefly put process of normalization remains non-trivial, not within its computational complexity, but within its neuro-biological implications and constraints. More specifically, there may be "effective" and "ineffective" genes. "Effective genes" may have a high impact on other downstream processes with even a few low expression intensities on one side, while "ineffective" genes have low impact albeit high expression intensities and high spread among the considered substructure. Summing up, the expression entries in M_{GE} may not be proportional for its actual importance.

An example may be genes describing and piloting cell proliferation. As cell division is naturally lowered in brains, in comparison to e.g. bone marrow, due to the density of neural cells, those genes may not be expressed as much. A significant increase of such may hence indicate functional deviant purpose or even pathological, i.e. cancerous, tissues, but works on low expression levels thus representing an "effective" gene. On the other hand, genes managing nutritional supply, may be highly expressed in all cells at all times, over-ruling the above "effective gene" in its expression intensities, thus constituting an "ineffective" gene.

Thus, there exist three different normalization schemes for this matrix:

1. A global normalization by dividing all matrix values by its global maximum,
2. a row-wise or per-structure normalization scheme, and
3. a column-wise or per-gene normalization.

While the global normalization remains the most common, it potentially leads to even lower values for "effective" and preserves over-represented and exaggerated expression energies for "ineffective" genes. Further per-structure, i.e. row-wise, normalization allows for highlighting of expression values that are significantly lower or higher among a single structure. Unfortunately, a row-wise normalization scheme will not break the above bias, but migrates the issue from the global to a structure-level scale. The third normalization scheme helps us to identify values associated with a specific gene, which are highly expressed, significantly lower or higher among the samples, i.e. *relative* expression intensities. This scheme helps us solving the issue of invariance to effectiveness of genes and allows for a fair representation of the transcriptome. The overall distribution of all all normalization schemes are depicted in Figure 2a to 2f.

Especially for our binary classification task of gene expression prediction (see Section 3.1.1), we need mapping of those entries to $\{0, 1\}$. Moreover, neural networks prefer inputs in the interval $[0, 1]$

Edit figure
caption

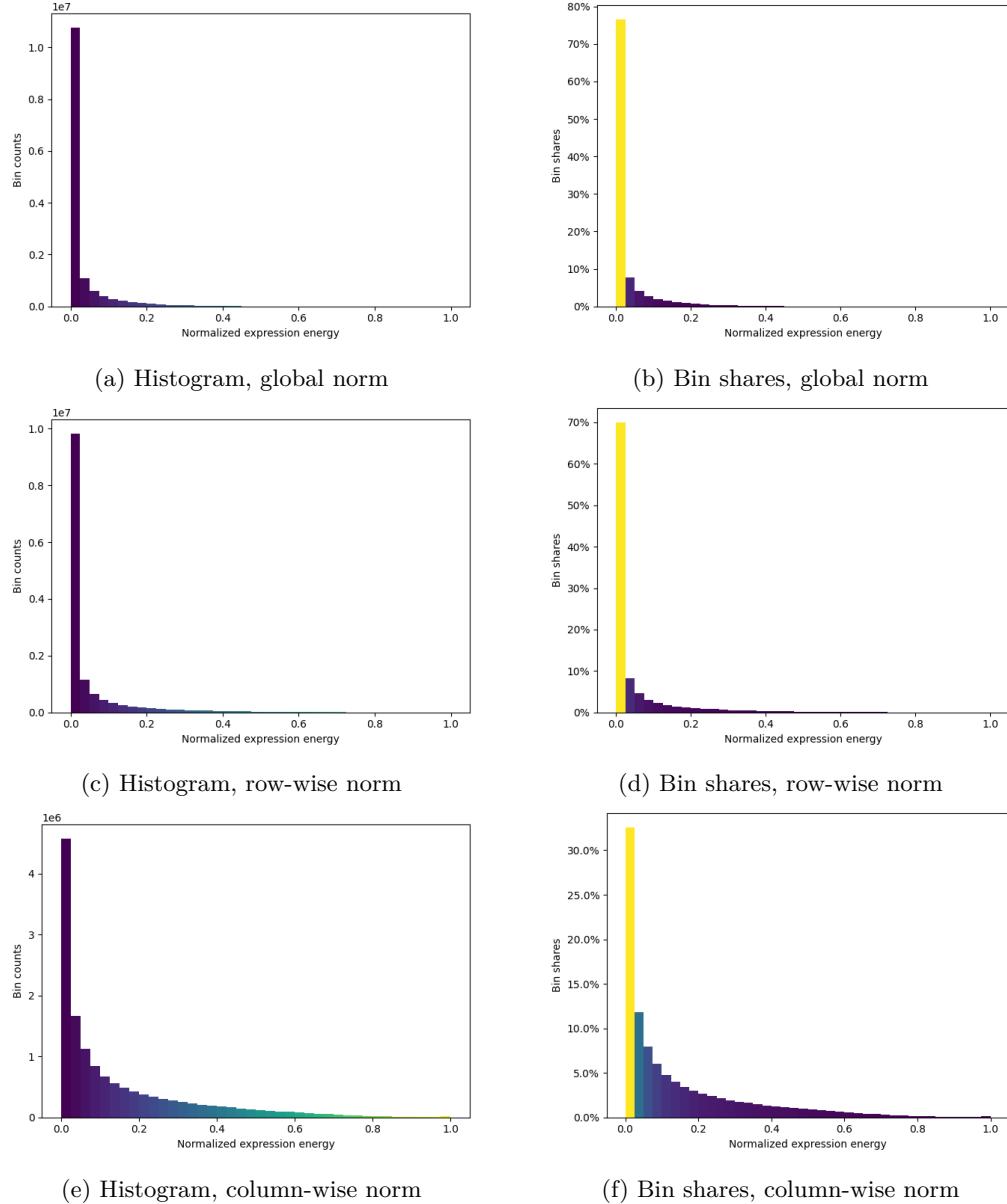


Figure 2: This figure shows the corresponding distributions of gene expression values for the respective normalization schemes, i.e. global, column-wise and row-wise, each associated with differing biologic interpretations. The expression matrix is determined by $|S| = 843$ structures and $|G| = 16679$ corresponding genes and the pairwise spatial gene expression values.

2a, 2c and 2e: Color-coded distribution of normalized expression energies; color describes the individual expression values. Note that values are on different scales i.e. 10^7 for 2a and 2c, and 10^6 for 2e.

2b, 2d and 2f: Color-coded proportional distribution of expression energies. While showing the same distribution, colors describe the several shares of each bin. Note the different scales. We clearly see a shift of the average for the column-wise normalization in comparison to the other two approaches.

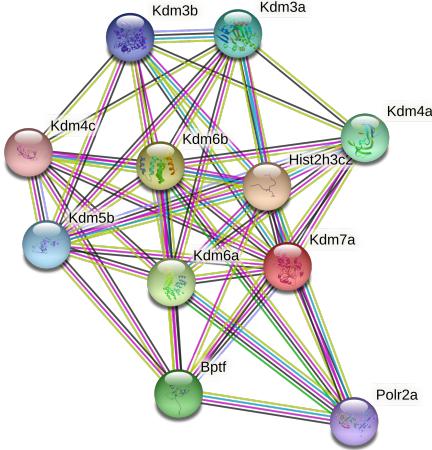


Figure 3: Protein-protein interaction graph for *Kdm7a*, an histone demethylase required for brain development, that is closely related to e.g. *Hist2h3c2*, a protein playing central role in transcription regulation and thus proliferation activity indicating e.g. malign tumors. For more information about *Kdm7a* and the used color encoding for vertices and edges, please see string-db.org/10090.ENSMSUPO00000002305

as we want to use M_{GE} as input for the problems described in Sections 3.1.2 and 3.1.3. Thus, we threshold the resulting normalized expression energies, by applying a fixed cut-off threshold $t \in [0, 1]$. We experiment with various expression thresholds as described in results Section 4.

3.2.2 Protein-protein interaction graph

As shortly sketched in the introduction (see Section 1), we want to apply graph convolutional neural networks over protein-protein interaction (PPI) graphs and identify patterns within. However, there are numerous databases for PPI graphs publicly and freely available. Common sources are CPDB(Lo et al., 2009), iRefIndex(Razick et al., 2008), MultiNet(Sengupta et al., 2023) and STRING(Szklarczyk et al., 2014). Due to our experience with the database and its success in other related tasks (Schulte-Sasse et al., 2021; Wang et al., 2021; Hinnerichs and Hoehndorf, 2021), we will use STRING (Version 10) as ground truth data. As STRING provide probabilities and hence confidence scores for each interaction, we re-use the recommended threshold of 0.7 in order to retrieve only high-confidence interactions. STRING database contains 300000 interactions between over 20000 genes gathered from other databases and literature for *Mus musculus*. An example interaction graph for *Kdm7a*, an histone demethylase required for brain development, and its interactors is shown in Figure 3.

3.2.3 Connectivity data

Understanding neural circuits and the wiring within our brains remains an unsolved, yet fundamental task to understand the processes of cognition. In history there were multiple approaches to measurement of such wiring schemes. We follow the classification of "Networks of the Brain"(Sporns, 2016), distinguishing three different brain connectivities:

1. Structural connectivity,
2. Functional connectivity, and
3. Effective connectivity.

The following definitions and descriptions are summarized from Sporns (2016)(see pp. 37-40) as it forms the most fundamental and influential standard literature up to this date. Please see this book for further information and more in-depth analysis of this topic and their cognitive and neurological implications.

Structural connectivity hereby links to physical or anatomical connections in form of linking neural elements, i.e. neurons. While this wiring measure may be relatively static on shorter time scales, "local" structural connectivity within regions may be dynamic or plastic over hours, days and months, due to our learning process. This is described in more detail in Section 3.2.3.1.

Functional connectivity (FC) tries to measure the deviation from statistical independence between distributed and often spatially remote neuronal units (Friston et al., 1993, 1994). This data may be recorded from different sources measuring neuronal activity as elaborated further in Section 3.2.3.2. Unlike structural connectivity, FC is highly time dependent. However, an observed statistical dependence between two nodes does not allow the inference of a causal interaction between them.

The third type of connectivity is *effective connectivity*, describing the *causal* effects between neural elements as detailed in Friston (1994) and Friston and Büchel (2000). It models the actual collaboration of and dependencies between structures as noted in 3.2.3.3.

3.2.3.1 Structural connectome data

We chose the AMBA partly with respect to its neat integration with the Allen Mouse Brain Connectivity Atlas (AMBCA)(Oh et al., 2014), published by the Allen institute, as a publicly available source for axonal connectivity data. The Allen Mouse Brain Connectivity Atlas is a high-resolution map of neuronal connections in the adult mouse brain, obtained from EGFP-expressing (enhanced green fluorescent protein) vectors following structural connections. As with AMBA, AMBCA tracing images were mapped back to 3D reference space using the CCF(v3) coordinates. Likewise, we are again able to group voxels to structures, parcelating the mouse brain and organized by the CCF ontology. All data was downloaded over the AllenSDK API. For a better understanding we provide a visualization of the measured EGFP data and the associated masks in Figure 4.

Let S be the set of structures, then $M_{\text{struc}} \in \mathbb{R}^{|S| \times |S|}$ denotes the structural connectivity matrix. We normalize all injections by their respective origin intensity projection. Second, we threshold the intensities with $t = 0.1$ as the connectivity distribution is quite similar the one for AMBA (see 2) and eventually symmetrize the matrix. We eventually obtain $|S| = 147$ structures and $\sum_{m \in M_{\text{struc}}} m = 2624$ functional connections between them, depicted in Figure 5a.

3.2.3.2 Functional connectivity data

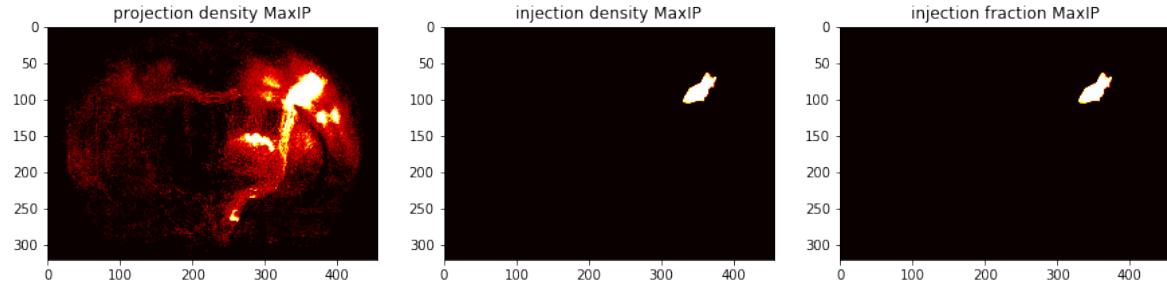


Figure 4: Test, what is MaxIP? (Maximum intensity projection)

The functional connectivity data was rather difficult to obtain in comparison to all other data sources. While previously described data sources are either published by the same institute or provide a inherently provide a wide range of mappings to other databases, there are very few datasets of functional connectivity data publicly and freely available. Moreover, these rare datasets should be related to non-pathological, adult mouse brains, further diminishing resulting numbers.

Functional connectivity may be measured using different data sources such as electrocorticography, fMRI or others. In most studies of this kind the latter is utilized as ground truth and thus shall be both used here and described briefly. Functional magnetic resonance imaging (fMRI) is non-invasive way of detecting joint neural activity the brain. The most frequent approach is measurement of blood oxygen-level dependent signal (BOLD signal), capturing the relative levels of hemoglobin-bound oxygen in the blood. The underlying assumption presumes a correlation of used oxygen and neural activity within some given region creating a unique signal. Observation of oxygen usage hence forms an indirect and delayed measurement. Moreover, we observe all regions at the same time, while acquiring BOLD signals in response to various tasks. A temporally proximate similar characteristic signal within two regions may indicate a collaboration of those two regions for that specific task. However in this publication, we will focus on resting-state fMRI (rs-fMRI), where no specific task is given. This is due to the over-expression of motoric collaborations and exploration of the *default mode network* (Sporns, 2016). Therefore it is more reliable in the investigation of the brains hierarchy and functional organization. Hence, a further constraint of rs-fMRI was added to the list of requirements to the dataset.

However, we were eventually able to obtain resting-state, functional connectivities from AIDAmri (Pallast et al., 2019), publishing both their pipeline for Atlas-Based Imaging Data Analysis (AIDA) and the associated dataset containing raw functional MRI data. The authors further capture the brains of 7 individuals as resting-state fMRI prior and after a stroke inducing intervention. Their follow-up software AIDAconnect (not published as a paper, please see [for more information](#)) first maps the AIDAmri data to CCF regions and second measures the associated change in both structural and functional connectivities pre and post stroke. One contribution of this work was to make this pipeline usable and provide *working* scripts following their approach. We would like to thank the AswendtLab (Website) for their support and help in order to get their pipeline running. While the authors capture axonal projections in AIDAmri, too, they measure far less structures than AMBCA on a smaller sample size, hence leading to our preference of AMBCA for this very study.

[Github link](#)

Let n be the number of observed individuals, then $M'_{\text{func}} \in \mathbb{R}^{n \times |S| \times |S|}$ denotes the functional connectivity matrix. We threshold the pair-wise connectivity by $t = 0.5$ obtaining a binary matrix. In order to combine the matrices over all individuals, we follow "The Handbook of Functional Connectivity" (Nieto-Castanon, 2020) by over averaging the respective connectivities, defined by

$$M_{\text{func}} := (M_{\text{func}})_{i,j} = \left(\left(\frac{1}{n} \sum_{k \in [n]} M'_{\text{func},k,i,j} \right) > t_{\text{func}} \right)_{i,j} \quad (1)$$

with $>: \mathbb{R} \times \mathbb{R} \rightarrow \{0, 1\}$ and $t_{\text{func}} := 0.5$. We symmetrize the resulting matrix. This leaves us with $|S| = 49$ structures and $\sum_{m \in M_{\text{func}}} m = 278$ connections between them. A visualization is shown in Figure 5b.

3.2.3.3 Effective connectivity

The third considered connection type is *effective connectivity* describing the causal relationship between different structures. In general, effective connectivity may be derived from various neural activity data likewise to functional connectivity, i.e. fMRI data over BOLD signals or electrocorticography. As already processed and available, we use the raw BOLD signals from AIDAmri (Pallast et al., 2019). In the literature there are various ways of measuring causality between signals, still constituting an active research topic. We follow the approach of *instrumental variables* described in Angrist and Pischke (2009) for econometrics and famously elaborated in detail in Pearl (2009).

Let n be the number of individuals, then $M'_{\text{eff}} \in \mathbb{R}^{n \times |S| \times |S|}$ denotes the effective connectivity matrix. Our pipeline returns the respective pairwise beta values of each pairwise beta regression of structure related signals, which are stored in M'_{eff} . Beta values may be interpreted similarly to covariance matrices, whereas their magnitude and quality is more expressive. As we are only interested in the interaction itself, but not the corresponding inhibition or activation, we calculate the absolute values and normalize for each individual. Likewise to our FC preprocessing, we average the causal connections along all individuals and symmetrize the resulting matrix. This may be summarized in the following equation:

$$\begin{aligned} m_{\max} &:= \max(|M'_{\text{eff}}|) \\ t_{\text{eff}} &:= 0.1 \\ M_{\text{eff}} &:= (M_{\text{eff}})_{i,j} \\ &= \left(\left(\frac{1}{n} \sum_{k \in [n]} \frac{|M'_{\text{eff},k,i,j}|}{m_{\max}} \right) > t_{\text{eff}} \right)_{i,j} \end{aligned}$$

with $>: \mathbb{R} \times \mathbb{R} \rightarrow \{0, 1\}$. This results in $|S| = 49$ structures and $\sum_{m \in M_{\text{eff}}} m = 492$ causal connections between them. A visualization of measured *effective connectivities* is shown in Figure 5c.

Do figure
description

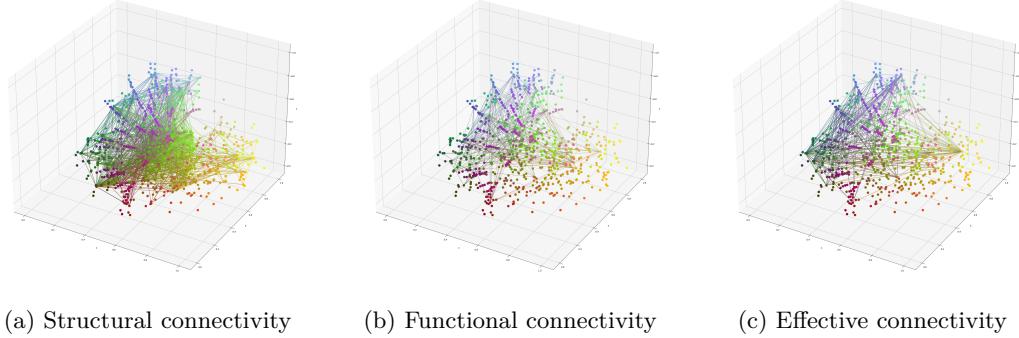


Figure 5: Test

Connectivity type	Number of present structures	Number of links
Structural connectivity	147	2624
Functional connectivity	49	278
Effective connectivity	49	492

Table 2: Overview on sizes of respective connectivity datasets

3.3 Model

Explanation of Measurement, Definitions, Indexes, Reliability and Validity of study method and study design

Description of Analytical Tehcniques to be Applied and justification for them

Reliability and validity of internal/external design and related subtypes

3.3.1 Feature generation and protein representation

For the prediction gene expression values in the first task (see Section 3.1.1), we need embeddings of each gene potentially representing its role in downstream tasks. Hence, we decided to exploit protein-function data based on both bottom-up and top-down i.e. molecular and phenotypical background data. For the bottom-up representations we utilize the DeepGOPlus (Kulmanov and Hoehndorf, 2019) embeddings, that are based on a prediction of protein-function based on their corresponding amino-acid sequence. For top-down and phenotypical features, we use DL2vec (Chen et al., 2020) over PhenomeNET(Hoehndorf et al., 2011) an aggregation of various ontologies such as Gene Ontology (GO) (Ashburner et al., 2000; Carbon et al., 2020), Mammalian Phenotype Ontology (MPO) (Smith and Eppig, 2009) and Uberon (Mungall et al., 2012), a multi-species anatomy ontology. DL2vec hereby formulates random walks over these ontologies, interpreted as knowledge graphs, and embeds them under usage of Word2Vec (Mikolov et al., 2013).

For the other two tasks (see Section 3.1.2 and Section 3.1.3), we will use the raw gene expression values parsed from AMBA to represent each substructure.

3.3.2 Graph convolutional neural networks

Graphs are one of the most expressive and common data structures for structured data. Especially in bio- and neuroinformatics, pairwise relations like in protein-protein interaction graphs, molecular graphs, biomedical ontologies, and biochemical knowledge graphs are widely spread. Considering graphs we distinguish between various types that allow application of differing algorithms. Especially with respect to the four graphs mentioned in Section 3.2, we will provide a brief classification.

Definition 1 (Homogeneous Graph) A (homogeneous) graph G is given by a tuple $G := (V, E)$ with V the set of vertices and E the set of edges. A graph G is called

- **directed** if $E \subseteq V \times V$, and
- **undirected** if $E \in 2^V$.

Variations of graphs are shown in the table below.

Graph type	Definition
(Undirected) Hypergraph	$G_H = (V, E)$ with $E \subseteq \mathcal{P}(V) \setminus \emptyset$
Weighted graph	$G = (V, E, w)$ with $w : E \rightarrow \mathbb{R}$
Node labeled graph	$G = (V, E, x)$ with $x : V \rightarrow \mathbb{R}^n$
Heterogeneous graph	$G = (V, E, l)$ with $l : V \cup E \rightarrow L$ and labeling space L

With these formulations we are able to both classify all previously described graphs, but also formulate constraints on graphs for the upcoming definitions. While the structure–gene expression in combination with the associated CCFv3 ontology is a edge and node labeled, heterogeneous graph, the PPI graph from STRING, and all three connectivity graphs are edge-weighted, homogeneous ones. However, after our thresholding process all weights are reduced to the binary space $\{0, 1\}$.

Graph convolutional neural networks are a geometric generalization of convolutional neural networks (CNN) commonly used over images, in order to learn generalizable kernels incorporating and learning to highlight local features. A stack of CNNs trained on cat and dog images, was shown to learn general features, such as ear and snout shapes. However, CNN filters have fixed geometric relations, i.e. we may always name and point at the element “right of” our current element within an image. Such orders among neighbors do not exist within graphs, and are even counterproductive, taking the order-invariance of the graph nature. Further, the number of neighbors for an element may not be constant in a graph, while they traditionally are within 2D- and 3D-images. Within a potential formulation of GCNs we would like to keep certain properties of CNNs, namely the ability to exploit locality, correlation of filter sizes and the considered neighborhood, learnable weights and node invariance.

We distinguish several tasks for GNNs based on classification. Here classification may also be used for representation learning by increasing the dimensionality of the mapping image to the desired representation space dimension.

1. Graph classification - classifying a given, entire graph,
2. node classification - given a graph with incomplete node features, predict the labels of the missing ones, and
3. link prediction - given a graph with incomplete adjacency matrix, predict the the missing links.

In our formulated tasks, we are performing node classification in the gene expression example (see Section 3.1.1), and graph classification in the sub-tasks described in Sections 3.1.2 and 3.1.3.

For naming convention identifying each layer type, we follow the notation of PyTorch Geometric (Fey and Lenssen, 2019).

3.3.2.1 Graph convolutional neural layers (GCNConv)

The first formulation of GCNs was introduced by Kipf and Welling (2016) and assumes the following two properties of a graph.

1. A *homogeneous* graph $G = (V, E)$, with $|V| = N$ and E summarized by the adjacency matrix $A \in \mathbb{B}^{N \times N}$
2. Node features, defined by a node labeling $x : V \rightarrow \mathbb{R}^D$ with feature dimensionality D summarized in the feature matrix $X \in \mathbb{R}^{N \times D}$

We will use the protein-protein interaction (PPI) graph as underlying structure, while adding the representations reported in Section 3.3.1. The PPI dataset is represented by a graph $G = (V, E)$, where each protein is represented by a vertex $v \in V$, and each edge $e \in E \subseteq V \times V$ represents an interaction between two proteins.

A GCN layer is defined by an update rule. Let $H^{(l)} \in \mathbb{R}^{N \times D}$ the current node representation at layer l , then the update rule is defined by.

$$H^{(l+1)} := f(H^{(l)}, A) \text{ with } H^{(0)} := X \quad (2)$$

The crucial step is encoded in the function f and may consist of the following three sub-steps.

1. Applying learnable weight matrix
2. Propagate/aggregate features to/from neighbors using a message passing scheme, and
3. Application of an activation function

summarized by

$$f(H^{(l)}, A) := \sigma(AH^{(l)}\Theta^{(l)}) \quad (3)$$

with weight matrix $\Theta^{(l)}$ of layer l , and σ a non-linear activation function, e.g. RELU. Here features are propagated by simple multiplication of the adjacency matrix. We will formulate more sophisticated aggregation kernels later in this chapter.

For comparison, CNNs we preserve so called *translational invariance*, i.e. the same filter may pick up the same features from an object regardless of the position of the object within the image. We generalize this property for graphs with invariance of order among neighbors, called *permutational invariance*. Further this formulation allows for locality exploitation.

The second crucial contribution of Kipf and Welling (2016) is a "kernel trick" allowing for vital speedup of feature propagation. The authors reformulate the aggregation in the following way.

$$H^{(l+1)} = \hat{D}^{-1/2} \hat{A} \hat{D}^{-1/2} H^{(l)} \Theta \quad (4)$$

with $\hat{A} = A + I$ denoting the adjacency matrix with added self-loops for each vertex, D described by $\hat{D}_{ii} = \sum_{j=0}^N \hat{A}_{ij}$, a diagonal matrix displaying the degree of each node, and Θ denotes the learnable weight matrix for a given graph $G = (V, E)$. Note that \hat{A} is formulated such that nodes are also directly influenced by their own previous representation. Naturally, the number of graph convolutional layers stacked equals the radius of relevant nodes for each vertex within the graph.

The update rule for the respective nodes is given by a message passing scheme formalized by

$$\mathbf{h}^{(l)}_i = \Theta^T \sum_j^N \frac{1}{\sqrt{\hat{d}_j \hat{d}_i}} \mathbf{h}^{(l)}_j \quad (5)$$

where both \hat{d}_i, \hat{d}_j are dependent on the edge weights e_{ij} of the graph. With simple, single-valued edge weights such as $e_{ij} = 1 \forall (i, j) \in E$, all \hat{d}_i reduce to d_i , i.e., the degree of each vertex i . We denote this type of graph convolutional neural layers by GCNConv.

While in this initial formulation of a GCNConv the node-wise update step is defined by the sum over all neighboring node representations. This leads to the effect of *over-smoothing*, i.e. over-softening of strong signals, especially in highly connected, non-sparse graphs. Further this makes the stacking of multiple layers nearly impossible to achieve feasibility due to fading signals.

3.3.2.2 Graph attention networks (GATConv)

A second approach to graph and geometric deep learning are graph attention networks (GAT), introduced in Chorowski et al. (2015). Likewise to GCNs, GATs may be formulated with the following steps:

1. Application of a learnable weight matrix,
2. an attention mechanism,
3. a activation function to combine the attention values, and
4. the aggregation function.

The crucial difference to plain GCNs lies within the second step, i.e. the graph attention mechanism. We therefore compute the pair-wise importance of node j 's representation to node i 's for all pairs i, j .

We do so by concatenation of embeddings followed by the dot-product with a learnable weight vector with an eventual application of LeakyReLU. This process is summarized by

$$z_i^{(l)} := \Theta^{(l)} h_i^{(l)} \quad (6)$$

$$e_{ij}^{(l)} := \text{LeakyReLU} \left(a^{(l)T} \cdot (z_i^{(l)} \circ z_j^{(l)}) \right) \quad (7)$$

with learnable vector a and edge weight e_{ij} , and allows for inclusion of a shared attentional mechanism a . The resulting pair-wise coefficients measuring the mutual importance are summarized for node i among all possible neighbors j using a softmax function.

$$\alpha_{ij}^{(l)} = \frac{\exp(e_{ij}^{(l)})}{\sum_{k \in \mathcal{N}(i)} \exp(e_{ik}^{(l)})} \quad (8)$$

The aggregation is then formulated as a weighted sum with above coefficients.

$$h_i^{(l+1)} := \sigma \left(\sum_{j \in \mathcal{N}(i)} \alpha_{ij}^{(l)} z_j^{(l)} \right) \quad (9)$$

We are further able to generalize this approach by so called *multi-head attention* allowing for multiple attention vectors a , which are then summarized within the aggregation function by averaging over all heads. Similarly, to GCNConv we denote a single layer of GAT as GATConv.

3.3.2.3 Training deeper graph convolutional neural networks (GENConv)

The propagation formulation of GCNs may be adjusted to other message passing schemes, as it suffers from the issues described in Section 3.3.2.1. As the considered radius of influence is determined by the number of used GCNConv layers, but multiple layers disable the network to pick up any signal. Thus, we can rearrange the order of activation function σ , aggregation AGG, and linear neural layer MLP with this formulation as proposed by (Li et al., 2020):

$$\mathbf{x}'_i = \text{MLP} (\mathbf{x}_i + \text{AGG} (\{\sigma(\mathbf{x}_j + \mathbf{e}_{ji}) + \epsilon : j \in \mathcal{N}(i)\})) \quad (10)$$

where we only consider $\sigma \in \{\text{ReLU}, \text{LeakyReLU}\}$. We denote this layer type as GENConv. While the reordering is mainly important for numerical stability, this alteration also addresses the vanishing gradient problem for deeper convolutional networks (Li et al., 2020). Additionally, we can also generalize the aggregation function to allow different weighting functions such as learnable SoftMax or Power for the incoming signals for each vertex, substituting the averaging step in GCNConv. Hence, while GCNConv suffers from both vanishing gradients and signal fading for large scale and highly connected graphs, each propagation step in GENConv emphasizes signals with values close to 0 and 1, and thus emphasizes strong negative and positive signals. The same convolutional filter and weight matrix are applied to and learned for all nodes simultaneously. We further employ another mechanism to avoid redundancy and fading signals in stacked graph convolutional networks, using

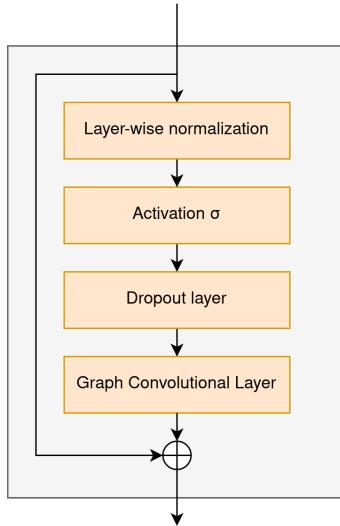


Figure 6: Textitext

residual connections and a normalization scheme (Li et al., 2019, 2020), introducing residual blocks for graph convolution. The residual blocks are reusable and may be stacked multiple times, allowing construction of even deeper graph convolutional neural networks. These residual blocks are composed of graph convolutional layer among other functions and hence allow for substitution of arbitrary graph layers as displayed in Figure 6.

3.3.2.4 Graph Kernels going beyond the Weisfeiler-Leman algorithm (KerGNN)

The fourth graph learning formulation, we present is KERGNN Feng et al. (2022). Recent studies (Xu et al., 2018; Morris et al., 2019) show that graph neural networks based on message passing are less expressive and powerful as the Weisfeiler-Leman (WL) algorithm (Leman and Weisfeiler, 1968) and the associated WL kernel definition (Shervashidze et al., 2009). The WL algorithm hereby describes an isomorphism test, providing a necessary but no sufficient test for non-isomorphism, which message passing may not supersede and outperform. Thus, Feng et al. (2022) provides a novel framework learning both hidden graphs combined with subgraph sampling to overcome this limitation, gaining additional expressive power. In comparison to message passing neural networks (MPNNs) which only consider *subtrees*, KerGNN the entire induced subgraph of the respective nodes neighborhood.

Let $G = (V, E)$ be some graph, $\phi_{[l]}$ be family of feature maps for each layer and $v \in V$ an arbitrary vertex. Further let $G_v = (V_v, E_v)$ be the induced neighborhood subgraph and ϕ_0 the feature map at layer 0 describing neighborhood mappings $\{\phi_0(u) : u \in V\}$. Then the prospective layer of the representation $\phi_{1,i}$ depended on the i -th graph filter $H_i^{(l)}$ with trainable adjacency matrix $A_i^{(l)}$ is defined by

$$\phi_{1,i}(v) := K\left(G_v, H_i^{(l)}\right) \quad (11)$$

with random walk kernel $K(\cdot, \cdot)$ (see Feng et al. (2022) for implementation details). Combined with a readout layer this may be leveraged for full graph representation learning. As we need to generate a low dimensional representations for the complete PPI graph with node features, this is naturally valuable. The readout layer is defined as a function Φ among all intermediate KerGNN layer embeddings ϕ_l given by

$$\Phi(G) := \text{concat} \left(\sum_{v \in V(G)} \phi_l(v) \mid l = 0, 1, \dots, L \right) \quad (12)$$

with L the number of stacked KerGNN layers. The authors show moreover, that this formulation is more powerful than the WL kernel and WL algorithm given a *sufficient* number of stacked KerGNN layers. This draws again connections to CNNs giving a permutation invariant relation of neighbors of node $v \in V$, which e.g. GCNs are not able to derive.

3.3.3 Dimensionality reduction techniques

The underlying task was briefly described in Section 3.1.2 with respect to gene expression. However, this task may also be formulated in a general. Here, we want to find a mapping $f_{\text{emb}} : \mathbb{R}^n \rightarrow \mathbb{R}^k$ with $k < n$ in order to transform data in high-dimensional space \mathbb{R}^n into lower dimensionality of space co-domain \mathbb{R}^k , preserving *meaningful properties* of the mapping's domain. The term *meaningful properties* is intentionally held vague as this may refer to both domain-specific but also space specific notions. This often refers to pair-wise linkage of entities.

In domain space, e.g. biomedicine, this may refer to external graphs, such as known biological or neurological interactions and biomedical correlations between embedded entities. Space specific features rather try to preserve spatial properties of the underlying space model, e.g. euclidean space. This may be expressed in terms of maintaining local proximity, e.g. in auto-encoders (Ng et al., 2011) or t-SNE (Van der Maaten and Hinton, 2008), or retaining cluster boundaries, e.g. in support vector machines (SVMs) (Noble, 2006). The co-domain \mathbb{R}^k of f_{emb} is termed *latent space* with latent dimensionality k . In order to better understand the yielded latent space, we choose $k < 4$ for a downstream, bijective mapping into color space for visualization. For our analysis we will only consider $k = 3$. Note that consider only unsupervised learning methods, as no ground truth is given for gene expression patterns and their relations.

We will briefly introduce all utilized dimensionality embedding methods and their respective assumptions and properties. PCA and t-SNE are not neural network-based, while UMAP and Parametric UMAP are.

3.3.3.1 Principal component analysis (PCA)

Principal component analysis (PCA) (Abdi and Williams, 2010; Wold et al., 1987) is among the most successful and simplest dimensionality reduction techniques. PCA therefore tries to capture the

datas underlying variance in space. Variance here both creates uncertainty and makes the target harder to explain and predict, but also gives a measure for importance of features. More specifically, features with no variance contain no information for the eventual embedding, similar to entropy in information theory. PCA yields a set of principal components, i.e. vectors forming a basis of the domain space, ranked by their respective variance in descending order.

Finding such principal components is done successively for each component. This may be formulated as a linear regression problem fitting a straight line trough the data. The prospective, next components are found likewise in embedding space, after removing all data correlated to the first component, i.e. the second components must be orthogonal to the first one (or generally all preceding ones). In order to embed our domain space into three dimensions as described above, we compute the first three principal components forming a basis of the domain space. Naturally, we then express every embedding, representing entities, as coordinates of that basis. Moreover, we linearly normalize each coefficient by the respective maximum and minimum values. This is feasible as we only consider euclidean spaces, preserving the respective bases.

3.3.3.2 t-SNE

t-Distributed Stochastic Neighbor Embedding (t-SNE) (Van der Maaten and Hinton, 2008) is a generalization of Stochastic Neighbor Embedding (SNE), and often used a direct competitor for PCA. However, as PCA only performs linear regression to build the linear basis of the respective domain space, its incapable to capture *non-linear* correlations, i.e. cannot separate data that may not be separated by a plane in respective dimensionality. This issue is addressed by t-SNE and used for visualization of complex manifolds and spaces such as image representation space returned by CNNs.

SNE measures similarity of data points by measuring their pair-wise conditional probability describing membership to the same cluster in the supervised and neighborhood in the unsupervised learning case, and likewise for t-SNE. After construction of distributions in domain, i.e. high-dimensional, space, t-SNE's algorithm constructs similar probability distributions for in the target co-domain, i.e. low-dimensional, space minimizing the Kullback-Leibler divergence (Hershey and Olsen, 2007) of two points with respect to their euclidean distance in the domain space. Kullback-Leibler divergence measures pairwise divergence of two given (uni-modal) distributions P, Q and is defined as relative entropy from Q to P by

$$D_{KL}(P||Q) := \int_{-\infty}^{\infty} p(x) \log \left(\frac{p(x)}{q(x)} \right) \quad (13)$$

for continuous spaces, where p and q denote the probability densities of P and Q , respectively.

3.3.3.3 UMAP

Uniform manifold approximation and projection (UMAP) (McInnes et al., 2018) is constructed based on and as a direct competitor of t-SNE. Likewise to t-SNE, UMAP tries to preserve the pair-wise distances, but changes the metric for distances. As described in Section 3.3.3.2, embeddings are correlated with distributions and hence a distance measure over probability distributions, i.e.

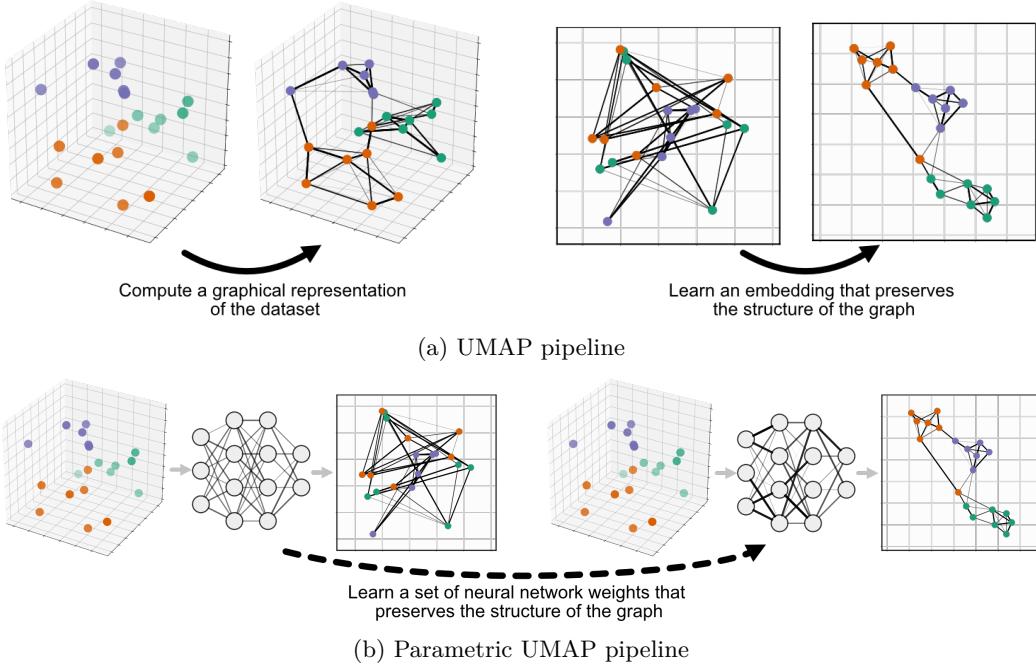


Figure 7: Visualization of UMAP’s and Parametric UMAP’s respective workflows. Images are taken from the PyPi `umap` library docs and the papers. The library is built and maintained by the UMAP and Parametric UMAP authors.

density functions for continuous variables as in our case, was introduced reusing the Kullback-Leibler divergence. UMAP interprets the data as samples of a manifold over Riemannian geometry based in algebraic topology. Riemannian manifolds are embedded as a generalization of Euclidean space, requiring and preserving Euclidean properties only locally. More specifically, Euclidean distance measures may be applied only locally. Hence, UMAP conserves Euclidean distances within the fuzzy simplicial complex, i.e. a “neighborhood subgraph”. The UMAP algorithm first computes a graphical representation of the dataset determined by an ablation of k-nearest neighbors, and learns the corresponding embeddings in a second step.

While this idea is very simple and naïve, it allows for both incorporation of neural networks for embeddings and analysis of the domain space’s manifold. This approach has shown great results for a variety of tasks, which we will not cite in detail here, and especially a significant improvement over PCA and t-SNE. A visualization of UMAP from the PyPi library `umap` is shown in Figure 7a. A great visualization of UMAP and its expressive power especially in comparison to t-SNE is given at the Google PAIR website with loads of examples. Moreover, see `umap-learn`’s comparison of UMAP with various other embedding methods with respect to runtime and embedding quality.

3.3.3.4 Parametric UMAP

The fourth presented dimensionality reduction method is Parametric UMAP (Sainburg et al.,

2021), that forms a direct successor of UMAP, not only by name. Developed in the same research group, UMAP learns a low-dimensional embedding of the underlying neighborhood graph in its second step, as described in Section 3.3.3.3. Within Parametric UMAP, this second step is substituted by neural network learning the correlation and relation of data and embeddings. This trick makes the method applicable to arbitrary deep learning applications, and enhances performance. The workflow of Parametric UMAP is shown in Figure 7b.

3.3.4 Hyperparameter tuning

As for the two classification tasks, namely gene expression prediction (see Section 3.1.1) and connectivity prediction (see Section 3.1.3), positive links are scattered and few, training and testing datasets are highly imbalanced. Exact number may be seen in the respective dataset description sections. The values

$$w_{\text{geneexp}} := \frac{|S| \cdot |G|}{\#\text{positives}} > 10, \text{ and} \quad (14)$$

$$w_{\text{connpred}} := \frac{|S| \cdot |S|}{\#\text{positives}} > 5 \quad (15)$$

describing imbalances within the gene expression and connectivity prediction dataset, hence need a suitable compensation in the respective loss functions and the evaluation metrics. Thus, we formulate a weighted version of the binary cross-entropy described by

$$l(x, y) := -w \cdot (y \cdot \log x + (1 - y) \cdot \log(1 - x)) \quad (16)$$

for given prediction x , ground truth y , and weights w as described in Equations 14 and 15, respectively. The losses are averaged among all predictions in the training set on which we apply the *Adam* optimizer (Kingma and Ba, 2015), guided by an early stopping scheme. For both prediction tasks we run a 5-fold cross validation for statistical robustness.

We determine the hyperparameters by a manual grid search among the search space, guided by the corresponding AUROC scores among the validation set. We further optimize the layer types, model depth and learning rate by a manual grid search, too.

For the task of dimensionality reduction, we do not have access to underlying ground truth caused by the explorative, and unsupervised nature. Corresponding loss functions are determined by each reduction method. Thus, hyperparameters of each embedding method is are determined likewise, with a manual grid search.

3.4 Evaluation and metrics

As mentioned, both classification datasets are highly imbalanced, and the dataset for dimensionality prediction is not bound to ground truth for evaluation. Therefor, we will treat these two problem

classes separately.

The classification tasks already follow the guidance of the customized and adjusted loss function. However, we need other metrics for evaluation of the observed predictions. We hence use the area under receiver operating characteristic curve (AUROC) on training and validation split data. The AUROC score is calculated by determining true positive rates at various false positive rate thresholds and use trapezoidal approximations to estimate the are under the curve. We use the widely used and open-source implementation of `scikit-learn` for that matter. We further choose the AUROC score as primary metric over measures like the area under precision recall curve (AUPRC) as AUPRC is sensitive to imbalanced datasets (Jeni et al., 2013b).

For evaluation of respective embeddings in the dimensionality reduction task, we use a custom metric based on similarity of embeddings for proximate and *related* sub-structures. We propose EMBSIM, a structure dependent similarity measure for embeddings.

Let S' be the set of all structures in the brain, and $f_{\text{emb}} : S \rightarrow \mathbb{R}^k, S \subseteq S'$ be the learned embedding function, where k is set to $k = 3$ if not stated otherwise. We are hereby only interested in the set of structures $S \subseteq S'$ that have a an associated embedding, i.e. gene expressions values are given for all $s \in S$ but not for all $s' \in S'$. Further, let s_1, s_2 be two structures, whose embedding similarity we are trying to determine. We compute this similarity by the pair-wise cosine similarity defined by

$$\text{sim}_{\cos}(x_1, x_2) := \frac{x_1 \cdot x_2}{\max(\|x_1\|_2 \cdot \|x_2\|_2, \epsilon)} \quad (17)$$

for embeddings x_1, x_2 . As we only operate within the space $[0, 1]$, we normalize $f_{\text{emb}}(S)$ by their maximum and minimum value along each dimension.

$$\text{norm}_S(s) = (\text{norm}'_{S,i}(s))_i \quad (18)$$

and

$$\text{norm}'_{S,i}(s) = \frac{f_{\text{emb},i}(s) - \min(f_{\text{emb},i}(S))}{\max(f_{\text{emb},i}(S)) - \min(f_{\text{emb},i}(S))}, s \in S \quad (19)$$

with $f_{\text{emb},i}$ the i -th dimension of f_{emb} . As this is only a linear transformation, this preserves the assumptions of all considered reduction methodologies.

Second, we weight these pairwise similarities by their relation, measured by the Resnik similarity (Resnik, 1995) $\text{sim}_{\text{Resnik}} : S \times S \rightarrow \mathbb{R}$, a semantic similarity measure, measuring the pairwise maximum information gain in the taxonomy of the closest common ancestor. We describe proximate and related sub-structures and their pairwise anatomical proximity, via the structure ontology of CCFv3. We further normalize these similarities by their maximum value among all structure pairs in the ontology. This measure gives us an assessment of importance of similarities for two structures. If two entities are close in the taxonomy want the metric to emphasize the correct similarity more than for distant and un-related structures.

Eventually, we combine these two values weighting the similarities obtained from the cosine similarity

by their corresponding Resnik similarity, normalized by the sum of weights. With

$$P_{\cos} := \left(\sum_{(t_1, t_2) \in S \times S} sim_{\cos}(norm_S(t_1), norm_S(t_2)) \right) \quad (20)$$

$$P_{\text{Resnik}} := \left(\sum_{(t_1, t_2) \in S \times S} sim_{\text{Resnik}}(norm_S(t_1), norm_S(t_2)) \right) \cdot \frac{1}{|S|^2} \quad (21)$$

we can eventually define EMBSIM by

$$\text{EMBSIM}(s_1, s_2) := \frac{sim_{\cos}(norm_S(s_1), norm_S(s_1)) \cdot sim_{\text{Resnik}}(norm_S(s_1), norm_S(s_2))}{P_{\text{Resnik}}} \quad (22)$$

and

$$\text{EMBVAR}(s_1, s_2) := \frac{\text{EMBSIM}(s_1, s_2)}{P_{\cos}} \quad (23)$$

EMBVAR is a direct extension of EMBSIM additionally penalizing overly-similar or even constant embeddings. We will do an extensive analysis of these measures within Section 4.2.

Second, we compare the resulting coloured brain images manually, and compare them via visual features. Notably this is prone to subjectiveness. Further, for comparison of various neural network based models fused to UMAP and Parametric UMAP, we will use the associated UMAP loss, which is invariant to model formulation but variant on structure selection.

What is EmbVar exactly?
Give good intuition here!
Note that EmbVar may reach values above 1

4 Results

4.1 Gene expression prediction

We originally started from the per section prediction in order to paste its performance and results to other "related" structures within in the mouse brain. One approach here is to score the models from one sub-structure to *related* structures via "transfer learning". Transfer learning was successfully applied especially in image classification and learning, and describes the re-purposing of property extraction methods, trained on one domain to the target domain, re-learning the final classification layers and steps. As features in images are fairly similar, mainly described by edges, vertices, and smaller shapes, the frequently utilized prelearned convolutional neural network (CNN) filters may be to other images. This saves both computing time, resources and energy, but also makes few-shot and zero-shot learning, i.e. learning and predicting based on very few examples in comparison to the amount of the model's parameters, feasible. Further this may yield additional insights into inter-domain relations and similarity. Filters learned on e.g. dog images, may be suitable for transfer learning predicting cat images, due to their phenotypical and anatomical resemblance, but less appropriate and practical for returning valuable features for e.g. optical character recognition from a document. By this performance gap for the two transfer learning tasks, we may assume that cats are more similar to dogs, while letters are rather dissimilar to dogs in their set of descriptive features.

We hence follow a likewise approach for mouse brains divisions. First, we will predict gene expression from molecular and phenotypical features described in Section 3.3.1 pursuing the frameworks described in Schulte-Sasse et al. (2021) and Wang et al. (2021)'s MOGONET for human tissues. Both use GCNCONV layers for their respective forecasts, hereby achieving tremendous performances. Second, we will try to see how generalizable these filters are for mouse brain tissues and their transcriptome from one structure to proximate ones.

For the first setup we anticipate the gene expression within a single structure, based on the non-structure dependent meta-information, given by the protein's representations. We thus run three different setups: (1) Molecular representations, (2) phenotypical features, and (3) both combined. We thus run five different neural network compositions described by their respective graph convolutional neural layer type. We will apply all layers described in Section 3.3.2 and a non-graph convolutional composition as baseline method described by "flat". Here flat includes linear layers similar to the learnable weight matrix of graph convolutional neural layers, but omitting the message passing over the graph. Sections are randomly sampled from *Frontal lobe*, *Parietal lobe*, *Occipital lobe* and *Temporal lobe*. The results are available in Table 3 and are based on 5 different sub-structures in the *transcriptomic smooth*, i.e. very similar in gene expression patters, *Frontal lobe*. Results within the other tissues perform similarly poor. Unfortunately, due to the specificity of the tissue and approach, we found no other works as baseline for this very task within mouse brain tissue.

In our experiments we generally observe a similar, but poor performance across all feature types, with a slight improvement of semantic phenotypical features over the molecular representations. Additionally, all tested graph convolutional layers and kernel types do not improve the down-stream performance of

Feature type	Flat	Graph Convolutional Layers			
		GCNConv	GATConv	GENConv	KERGNN
Molecular features	0.65	0.64	0.63	0.63	0.64
Phenotypical features	0.68	0.66	0.64	0.66	0.67
Molecular + phenotypical features	0.67	0.65	0.66	0.67	0.68

Table 3: AUROC scores for singular structures within the *Frontal lobe*; Forecasted with different graph convolutional neural layer types over three feature setups averaged among different substructures. Molecular features from DeepGOPlus (Kulmanov and Hoehndorf, 2019) and phenotypical features from DL2vec (Chen et al., 2020) over PhenomeNET (Hoehndorf et al., 2011).

this task. Contrary, GCNs over molecular features seem to worsen the predictive performance. This may be founded in the very node specific nature of our molecular representations, which may not propagate over the network. Eventually, we observe no behavior as reported in Schulte-Sasse et al. (2021) and Wang et al. (2021).

As described, we further test the graph convolutional neural filters in a transfer learning scenario. We thus learn the filters on different *related* sub-structures within the mouse brain and validate on the target structure. Likewise to the first sub-task, we analyze results for sub-structures in different compartments of the brain: *Frontal lobe*, *Parietal lobe*, *Occipital lobe* and *Temporal lobe*. For a given structure, we define *related* divisions as "siblings" and "cousins", i.e. structures connected by paths `hasParentn o hasChildn` with $n \leq 2$, within the structure ontology described by the Allen Mouse Brain Common Coordinate Framework (CCFv3). Results for this second step are not displayed in a separate table, but may be described here in detail. Due to the high similarity between neighboring structures, results in training and validation-phase were highly similar. For $n = 1$, i.e. transfer learning from sibling structures, high correlation of adjacent divisions leads to leak from training to validation datasets. This leads to arbitrary performance, determined by the networks ability to memorize the past samples, and the number of siblings in the ontology. This holds across all observed across all brain compartments. These results are fairly different, when concerning $n = 2$, i.e. cousin structures, where results were only determined by the number of neighbors. For smaller sets of relatives, especially seen in the temporal and frontal lobe, very similar behavior as for $n = 1$ was observed. However, for larger sets the results tended crucially towards outcomes shown in Table 3. This may be due to the diminishing transcriptomic similarity for larger distances within the structure ontology and the varying parcelation across brain compartments within this ontology.

We thus conclude that GCNs may not be able to exploit protein-protein interaction networks for gene expression prediction. This may be caused in the complicated structure and complex nature of gene expression. We elaborate further on this matter in Section 5. Due to negative outcomes within this task and the poor performances measured among all tested feature types, we move towards showing the influence of transcriptomic patterns for down-stream tasks.

4.2 Dimensionality reduction and its combination with different graphs structures

In this second section we want to show and demonstrate the results of our experiments on dimensionality reduction. This section is divided into two subsections. First, we show the validity of EMBSIM (see Section 3.4 and Equation 22), followed by the results of our application of graph convolutional neural networks.

4.2.1 On the validity of EmbSim and EmbVar

The formulation of EMBSIM contains various thoughts. First, we want to achieve so called sample invariance, i.e. metric that is invariant to the amount and selection of samples within the ontology. As our different datasets, especially in the task of connectivity prediction, only contain data on a rather small-scale subset of structures, we want to formulate a measure allowing for cross-subset comparison of embedding similarities. Second, we want to penalize overly-similar embeddings. Most intuitive measures, such as the sum of cosine similarities fail to address this issue. An embedding mapping all entities to the same embedding, e.g. $(0, 0, 0)^T$ is optimal, is optimal for this kind of measurement, as it maximizes the pairwise similarity. However, to cope with both issues and get a sufficient understanding of the underlying data, we will need both methods. Third, we naturally want our metric to correlate with our natural understanding of a good embedding, i.e. similar embeddings for structures with similar gene expression values.

First, we show the sample invariance of EMBSIM. We assume three different toy taxonomies and ontologies, describing the sample issue, all shown in Figure 8. Let a_i and b_j be entities of respective classes. Further, let $\text{sim}_{\cos}(a_i, a_j) = 1$ and $\text{sim}_{\cos}(b_i, b_j) = 1$ for all pairs i, j , describing maximum similarity within classes. However, we further assume $\text{sim}_{\cos}(a_i, b_j) = 0$ for all i, j . Then the smoothness functions yield the following results:

Tree type	EMBSIM	EMBVAR
Balanced 5×5 tree	0.89	0.94
Unbalanced 2×8 tree	0.91	1.00
Unbalanced 1×4 tree	0.91	1.00

As shown, both measures are invariant to sample size comparing the 2×8 and 1×4 trees, if and only if samples are drawn from the same distribution. However, EMBVAR heavily drops functioning when prompted with varying number of samples, caused by the added normalization term. This may be crucial for our purpose, as number of structures are significantly decreased in the connectivity prediction example. Additionally, we see better consistency in EMBSIM for these examples, correctly dissecting the clusters and computing persistent smoothness values. Thus, EMBSIM is sufficient and superior when comparing embeddings over the same set of sub-structures, but also when considering different subsets.

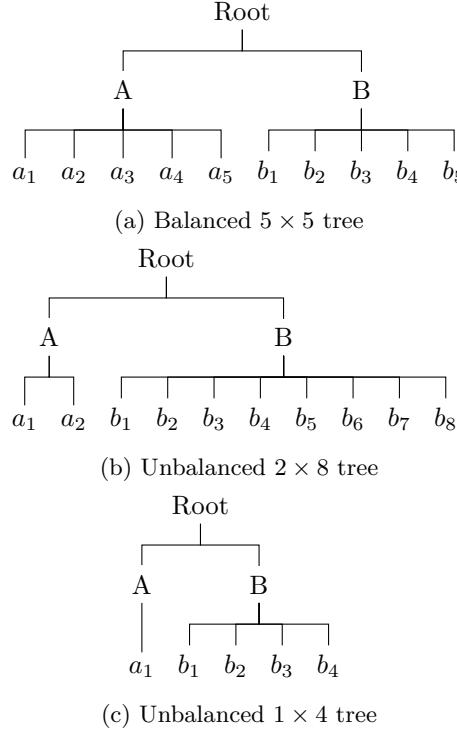


Figure 8: text

In the second issue of penalizing overly similar embeddings we less considered with the choice of sub-structures, but with the properties of embeddings. Therefore, we introduce three, naïve baseline functions that shall serve as comparison to overcome, and proof the third point. Functions are given by their respective probability functions $p(x)$ shown in Figure 9:

$$p_{\text{uni}}(x) := \begin{cases} 1 & \text{if } x \in [0, 1] \\ 0 & \text{otherwise} \end{cases} \quad (24)$$

$$p_{\text{dirac}}(x) := \begin{cases} \infty & \text{if } x = 0.3 \\ 0 & \text{otherwise} \end{cases} \quad \int_{-\infty}^{\infty} p_{\text{dirac}}(x) = 1 \quad (25)$$

$$p_{\text{gauss}}(x) := \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad \text{with } \mu := 0.3, \sigma = 0.05 \quad (26)$$

For all three functions, $\int_{-\infty}^{\infty} p(x) = 1$ holds by their respective definitions.

Further, all three are to be seen as distributions for sampling of the structure's respective embeddings, that naturally do not correlate with the corresponding division's properties. Applied smoothness metrics results are shown in Table 4. Let us consider p_{dirac} and p_{gauss} initially, constructing toy examples with too small image. Per definition, the vectors sampled from p_{dirac} are constant and thus have a mutual similarity of 1 over all possible pairs. However, as described in Section 3.4, we

Dimensionality reduction	Uniform p_{uni}	Constant p_{dirac}	Gaussian p_{gauss}	PCA
EMBSIM	0.79	—	0.94	0.96
EMBVAR	0.99	—	0.91	1.01

Table 4: Test

normalize all embeddings by their respective maximum and minimum values in among each coordinate with norm_S (see Equation 18). As this leads to a division by 0 error, EMBSIM and EMBVAR are consistently not computable for these cases. However, p_{gauss} is a more sturdy adversary for the smoothness metrics. While $p_{\text{gauss}}(x) > 0$ across the entire domain of x , the distribution may not be adjusted by norm_S and hence approximately reduced to p_{uni} . The results table Table 4 shows clearly, that EMBSIM struggles to penalize the overly similar embeddings of this very distribution. On the other hand, EMBVAR behaves nicely, showing a clear gap of smoothness in comparison to *reasonable* distributions like PCA. Hence, EMBVAR is clearly superior in penalizing overly smooth embeddings functions.

Third, we want the smoothness metrics to correlate with our natural understanding of good embeddings. As described in Valk et al. (2020) and Partel et al. (2020) gene expression patterns are very similar across the frontal lobe and the *Precentral gyrus*. Used distributions generate the brain colorings displayed in Figure 9, which do not coincide with our neurological background knowledge. We will provide the computed visualization of the *proper* embedding methods later in this chapter for comparison. However, EMBVAR weighs the smoothness of the image even higher than the generated representations from principal component analysis (PCA). This is caused by the normalization by the sum across all similarities. As random embeddings are very dissimilar, they are not penalized and even rewarded for their spread usage of the representation space. Fortunately, EMBSIM nicely shows a significant gap between the PCA generated and uniformly sampled vectors.

As we see from our toy examples, EMBSIM and EMBVAR have differing strengths measuring, but also struggle crucially with simple and naïve baseline methods. To summarize, EMBSIM differs precisely between non-similar and similar embeddings with respect to our similarity measure, but struggles to penalize overly akin and alike embeddings from semantically *better* mappings. This issue is resolved by EMBVAR. Thus we conclude that only both metrics combined may be sufficient to properly judge improvement over other embedding functions.

4.2.2 Enhancing neural embedding methods with GCNs

In this section we will examine in more detail the results of our dimensionality reduction experiments. Thus, we first investigate the overall similarity of gene expression values across the mouse brain, measured by EMBSIM and EMBVAR based on the raw expression values as baseline method to overcome. We combine this with a study on the most expressive genes and the corresponding normalization scheme (see Section 3.2.1). Afterwards, we analyze the influence of GCNs over PPI graphs on the performance and expressivity of the dimensionality reduction approaches.

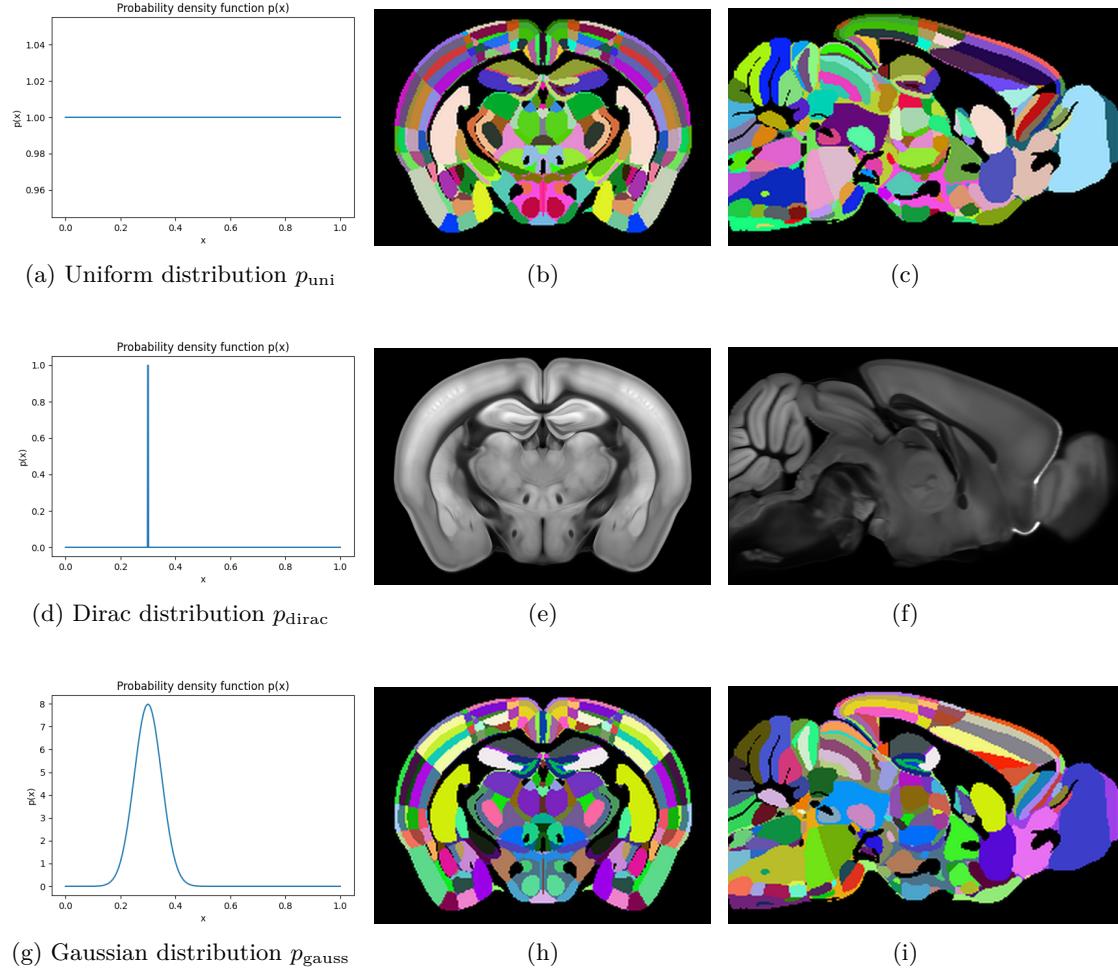


Figure 9: Baseline functions from Section 4.2.1 and the corresponding colored brain structures, coronal and sagittal cross-section. (a) - (c): Uniform distribution p_{uni} ; (d) - (f): Constant values with respect to Dirac distribution p_{dirac} with peak at $t = 0.3$, note that brain coloring for constant colors is not possible due to coordinate-wise normalization of embeddings, thus these images are placeholders; (g) - (i): Values sampled from Gaussian distribution p_{gauss} with $\mu = 0.3, \sigma = 0.05$

<i>N</i>	<i>norm</i>	Raw gene expr.		PCA		UMAP	
		EMBSIM	EMBVAR	EMBSIM	EMBVAR	EMBSIM	EMBVAR
20	none	0.91	0.99	0.94	1.00	0.96	0.99
	row	0.91	0.99	0.91	1.00	0.98	0.99
	column	0.94	1.00	0.95	1.00	0.95	0.99
	global	0.91	0.99	0.94	0.99	0.97	1.00
100	none	0.90	1.00	0.95	1.00	0.95	1.00
	row	0.90	1.00	0.92	1.00	0.94	1.00
	column	0.91	1.00	0.95	1.00	0.91	1.00
	global	0.90	1.00	0.94	1.00	0.95	1.01
1000	none	0.88	1.01	0.96	1.00	0.94	1.01
	row	0.88	1.01	0.94	1.00	0.96	1.00
	column	0.84	1.02	0.95	1.00	0.96	1.02
	global	0.88	1.00	0.96	1.00	0.94	1.01
16679	none	0.84	1.01	0.96	1.00	0.92	1.03
	row	0.84	1.01	0.94	1.01	0.96	1.00
	column	0.72	1.03	0.96	1.01	0.96	1.01
	global	0.84	1.01	0.97	0.99	0.94	1.01

Table 5: Textitex

Please note that the majority of these results are image-driven. As printed versions likely will not capture the entire color depth and nuances shown, please refer to the digital version of this script for more insights.

Our data preparation scheme allows for testing of different ways of representing omics data. We hence evaluated various ways to normalize gene expression values, as these normalizations capture varying biological information. Generally, there are the four possible normalization schemes possible, the three introduced and explained in Section 3.2.1 and no normalization, which makes the features infeasible for the neural network-based approaches, e.g. UMAP and Parametric UMAP. Further, we survey the behavior of dimensionality reduction methods, based on different subsets of genes. We therefor rank the genes based on their respective sum of expressions across all structures, with respect to each normalization scheme. We then select the $N \in \{20, 100, 1000, 16679\}$ most expressed genes and prune each sub-structure representation down to these genes. For a better overview we will only display the results PCA and UMAP here, but all reduction embeddings behave likewise. Results are depicted in Table 5. Experiments were repeated 5 times for the non-deterministic approach UMAP and averaged over all results.

There are several trends and properties clearly visible in this very table. Foremost, raw gene expression values differ in their corresponding similarity, measured by EMBSIM and EMBVAR, across the considered number of genes. The similarity is comparable to PCA and UMAP results for the subset

of 20 most expressed genes. However, the low EMBVAR value, diagnoses overly similar embeddings. The more genes we consider, the lower the like analogy gets, culminating in EMBSIMS and EMBVARS of 0.72 and 1.03 respectively. This indicates both the most information content, which follows naturally by the features derivation process, and spread of data. Moreover, we may clearly see a large discrepancy in kinship across the various normalization schemes. As described in Section 3.2.1, about 70% of all expression values are very close to zero for none, row-wise and global normalization, leading to increased sameness in representation.

Fortunately, both PCA and UMAP are able to both extract the important features, but also keep the *less relevant* values for spreading of embeddings along the representation space approaching a clearer dissection of clusters. Moreover, UMAP and PCA as representatives for traditional and neural network-based dimensionality reduction approaches both clearly outperform the baseline of similarity of raw gene expression values. This reasons and underlines our choice of normalization scheme and gene set choice, but also accentuates the validity of elected similarity metrics.

In the second step we approach the impact of GCNs on this embedding method. Initially, we show the computed performances across all introduced methods without GCNs, followed by a description and implementation details of the used model in combination with GCNs, and closed by their respective results and the insights we gain from this. Statistics of these trials are shown in Table 6. Corresponding brain images are shown in

For our experiments we evaluate the four methods PCA, t-SNE, UMAP and Parametric UMAP on the unsupervised task of dimensionality reduction based on the raw, normalized gene expression energy data over the full set of genes. We consider 843 structures and 16679 associated genes. As mappings to the corresponding proteins are not available for all genes and some genes are linked with the same Entrez ID, we yield a total subset of 10689 genes eventually. We prune the corresponding representations accordingly for GAT based methods, but also test for the full set of genes. Only results for the pruned set of genes are shown, as full set performances are *very* similar. We further extended a digit in these results tables, to underline the rather marginal gaps measured. We generally see a very solid performance across all applied methods. Moreover, all "baseline" methods, i.e. PCA, t-SNE and UMAP achieve akin EMBSIM and EMBVAR scores. A noticeable drop in both measures and the UMAP-loss is observed from UMAP to Parametric UMAP. For these experiments we tested numerous hyperparameter settings, as described in Section 3.4. Their corresponding images may be seen in Figure 10. We thus map the corresponding embeddings into color space and display a coronal and one sagittal cross-section across the colored CCFv3 reference atlas. For a better intuition about measured section boarders, please refer to Figure 11. Here PCA and t-SNE show a very diverse spectrum, causing the high performance with respect to EMBVAR. Moreover, the corresponding images are highly parcelated and boarders between neighboring structures are clearly visible. This lacking smoothness of embeddings may also be seen in the 3D-scatter plot, which we will not display in this publication due to lack of additional insights.

This issue is solved with the formulation of UMAP as described in more detail in Section 3.3.3,

Method GCN layer type	PCA	T-SNE	UMAP	Parametric UMAP				
				None	GCN	GAT	GEN	KERGNN
EMBSIM	0.956	0.956	0.959	0.952	0.920	0.927	0.968	0.964
EMBVAR	1.010	1.003	1.006	1.002	1.038	1.028	1.022	1.019
UMAP-loss	–	–	0.94	1.06	1.75	1.52	0.62	0.88

Table 6: Textitext

ensuring smoothness and closeness of similar embeddings. This is clearly visible in Figure 10e and 10f. Both coronal and sagittal cross-section greatly improve the smoothness of embeddings across structures, while clearly highlighting the overall regions. However, we evidently detect the issue of over-smoothing, which we could not fix with different sets of hyperparameters. This issue may also be seen in the values of EMBVAR in comparison to the right side of the table. At this point we will introduce the work of Partel et al. (2020), who introduced a novel in-situ sequencing (ISS) dataset and pipeline for mouse brains. The corresponding embeddings calculated over UMAP, too, are displayed in Figure 11. Here, gene expression data was not aggregated across divisions, but was left as voxels. Corresponding to our data and visualizations, embeddings nicely display the brains overall compartments and sub-structures with respect to gene expression values. Yet, UMAP leads to overly smoothed images with only little sensitivity to small derivations.

Unfortunately, we could not find a suitable implementation of UMAP allowing for custom, deeper encoding models (though theoretically possible), but we were successful looking for suitable implementations for Parametric UMAP. Links and further links are given in the Github repository.

Align to
middle

Thus, the following images and results are given with respect to deeper and thus more expressive neural network-based models than for UMAP. Moreover, we may re-use the UMAP loss, used as the optimization objective for UMAP and Parametric UMAP. It describes the proximity and boundness of simplices in the domain space, and how well they are preserved in lower dimensions. For the non-graph convolutional baseline encoder for Parametric UMAP we run a manual grid search for the best set of hyperparameters. Moreover, we instantiated control neural network, consisting of the same layers as GENConv, but removed the message passing step. These two cases coincided, thus leading to the graph-less baseline method and its data shown in the right part of Table 6.

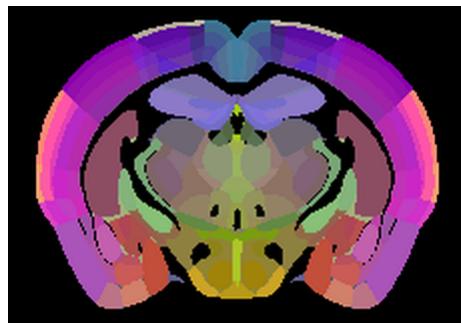
As base graph, we utilize the protein-protein interaction graph with associated gene expression value for each gene-protein pair as node feature. Thus every node only has a singular value as description. As encoding model for the combination with Parametric UMAP we applied the model shown in Figure 12. It consists of three stacked ResGraphConv Blocks allowing for deeper neural networks overcoming the oversmoothing of node features in GCNConv and GATConv, especially in highly connected graphs like our STRING-based PPI network. However, this circumvention does not work for GCNConv and GATConv as seen in the respective results. The performance drops crucially in comparison to the non-GCN Parametric UMAP model with a considerable jump in both EMBSIM and EMBVAR. The latter indicates overly similar embeddings, which may be caused by the respective kernel filters and message passing scheme. This also applies for GATConv, which applies the same message passing



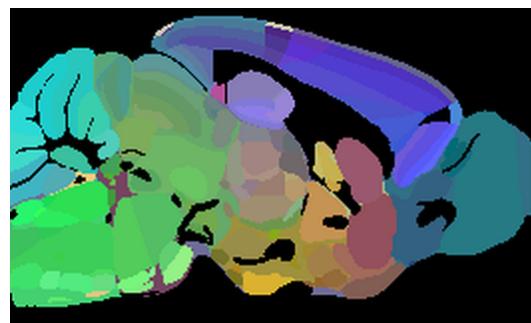
(a) PCA coronal



(b) PCA sagittal



(c) T-SNE coronal



(d) T-SNE sagittal



(e) UMAP coronal



(f) UMAP sagittal



(g) Parametric UMAP coronal



(h) Parametric UMAP sagittal

Figure 10: Textitext

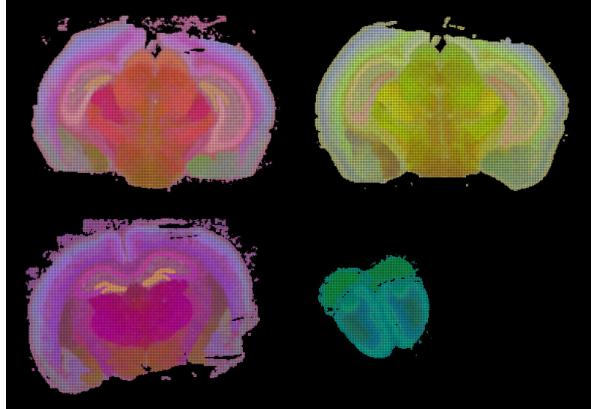


Figure 11: Textitext

approach as GCNConv. A large increase in EMBSIM and EMBVAR may be seen in juxtaposition to GENConv and KerGNN. Both methods try to overcome the oversmoothing issues with novel learnable weighting methods and a learnable softmax preserving strong signals for GENConv, and shadow graph construction for KerGNNs, cutting edges from the dense PPI network. An increase in both measures indicates much better embeddings, which may not be reproduced with non-GCN methods. This signifies both increased similarity and good spread across the representation space. The improvement of embeddings may also be observed in the gap in UMAP-loss, also indicating improved embeddings.

- results on the images?

Yet, we see a distinction in performance from GENConv to KerGNN in terms of UMAP-loss, which we are not able to explain but must be caused in the different formulations. We assume that some singular genes are highly predictive for the structures orientation, as indicated by the gene set experiments in Partel et al. (2020). Thus, GENConv might be able to underline, propagate and preserve these features better. The authors show, that a small subset of about 20 genes is able to properly represent each structure over UMAP. We show, that this is only partly true and overly simplifies the underlying issue in Figure 4. We conclude that GCNs may be able to naturally select and derive importance measures for these decisive genes and their corresponding interactions, which non-GCN deep neural networks and traditional approaches, i.e. PCA and t-SNE, are incapable of.

Include
GCN-based
images

4.3 On the linkage of connectivities and gene expression patterns

In this section we will describe the experiments and their outcomes for the connectivity prediction task. Here, we use normalized gene expression values as representations for each structure and predict the interactions between them. Further, the task is based on a two-entity prediction, whereas these entities are represented in the same representation space. This formulation causes a variety of challenges. Let us assume a naïve neural network (nNN) $f_{nNN} : \mathbb{R}^n \times \mathbb{R}^n \rightarrow [0, 1]$ with parameterization Φ where inputs are bluntly concatenated and put through a series of linear neural layers before being reduced to a dimensionality of 1 for the output function. Moreover, let $s_1, s_2 \in S$ be the gene expression-based representations of two structures in the mouse brain with S the set of structure embeddings. First,

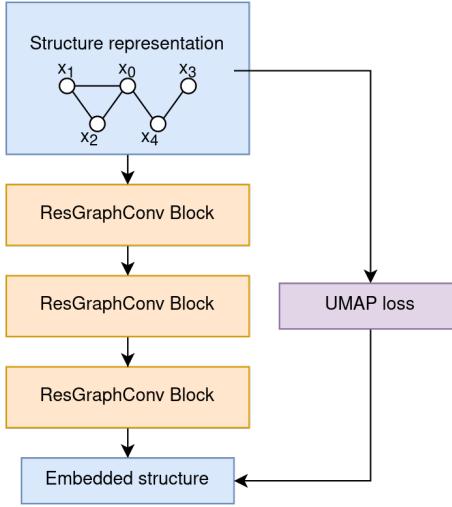


Figure 12: Textitext

neural network models impose and assume an order among the inputs, indicated by the Cartesian product characterizing f_{nNN} 's domain. This causes $f_{nNN}(s_1, s_2) \neq f_{nNN}(s_2, s_1)$ for every distinct pair $s_1, s_2 \in S$ with $s_1 \neq s_2$. This may be solved by either a special regularization term in the model's loss function or my a special model structure. Second, we wish our model to be descriptive and explainable in terms of interpretability of results. Naturally, neural networks and their intermediate representation spaces are seldom understandable for humans. This is frequently done by a mapping into color space and the corresponding visualization. However, even if we assume the second to last layer of f_{nNN} to be of dimensionality $k = 3$, i.e. a bijective mapping to RGB color space exists, the meaning stays unsolved, as a complex transformation of this space follows in form the neural output layer.

We aim to solve this issue by proposing a Siamese neural network(Bromley et al., 1993) (see Chicco (2021) for an overview) for this purpose. Siamese neural networks (SNNs) embed the two considered entities from the same representation domain into the same embedding space. Here, the SNN optimizer tries to maximize the similarity of the two embeddings if linked and minimize the kinship for non-linked ones. This solves the two issues mentioned above. On one hand, as both entities are put through the same embedding model and the similarity function is commutative, SNNs are invariant to the order of their arguments. On the other hand, we may set the dimensionality of the embedding space to $k = 3$ which is straightforward to visualize. Second, similar embeddings are embedded similarly imposing a natural structure among the embeddings. For the respective encoding model, we use a stacked architecture of ResGraphConv blocks (see Section 3.3.2.3 for more information) to incorporate the chosen graph convolutional neural layers equivalently as in Section 4.2.2. Again, we use the "flat", non-GCN model as a baseline approach for our method. Again, we use the same architecture but remove the corresponding message passing step. Eventually, we use Cosine similarity as similarity metric. The full model is shown in Figure 13.

Connectivity type	GCN layer type				
	None	GCN	GAT	GEN	KERGNN
Struc conn	0.80	0.73	0.76	0.82	0.83
Funn conn	0.76	0.72	0.73	0.79	0.81
Eff conn	0.77	0.75	0.75	0.79	0.78

(a) AUROC scores for embedding dim $k = 20$

Connectivity type	GCN layer type				
	None	GCN	GAT	GEN	KERGNN
Struc conn	0.70	0.68	0.67	0.72	0.73
Funn conn	0.69	0.54	0.63	0.71	0.73
Eff conn	0.70				

(b) AUROC scores for embedding dim $k = 3$

Table 7: Textitext

While the input dimensionality is fixed among for all entities, embedding space dimensionality remains a crucial hyperparameter and thus a central research question. Naturally, the *real* embedding space does not have to follow our desire for visualizations and better understanding. We thus analyze the impact of the output dimensionality on the predictive performance and descriptive capabilities of the SNN. We hence choose $k \in \{3, 20\}$ for a better understanding.

The results of our experiments are shown in Table 7a and 7b.

- Model requirements
 - should be order invariant
 - solve encoding problem
- different used graph convolutional filters
- How do we annotate graph

Brief Overview of Material

Findings (Results) of the Method of Study and Any Unplanned or Unexpected Situations that Occurred

Brief Descriptive Analysis Reliability and Validity of the Analysis

Explanation of the Hypothesis and Precise and Exact Data (Do Not Give Your Opinion)

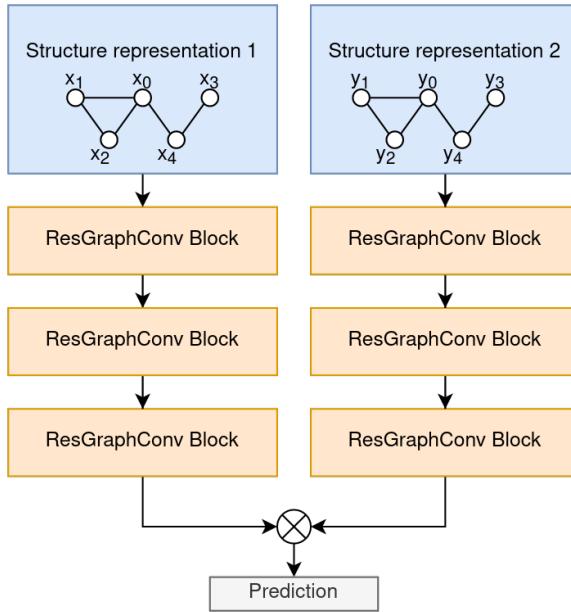


Figure 13: Textitext

5 Discussion

Brief Overview of Material

Full Discussion of Findings (Results) and Implications

Full Discussion of Research Analysis of Findings

Full Discussion of Hypothesis and of Findings

Post Analysis and Implications of Hypothesis and of Findings

5.1 Evaluating different feature types for gene expression prediction

- GCNs over protein-protein interaction graphs may not be applicable for gene expression prediction, but are suitable for predicting down-stream behavior
- other ways to incorporate structure data in there → no structure specific embeddings containing molecular features that are not gene expression value dependent available
 - structural ontology / closeness
 - developmental hierarchy of tissue

5.2 Gene expression patterns and their relation to GCNs

- influence of normalization technique → Structure, gene, global

- we considered including connectivity graphs into UMAP-loss, but no natural formulation for combination is possible. Here an artificial weighting of respective UMAP proximity graph and external connectivity graph is necessary. We conducted several experiments Collapses to either side, namely either overly weighting embedding proximity or connectivity matrix. We could not define a stable set of parameters for this to work.
- no integration with UMAP possible due to missing implementation of UMAP in PyTorch

5.3 Connectivity prediction

- we reduce to three dimensions, but there does not necessarily needs to be a 3D-representation
- naturally, much better results with higher embedding dimension → poor visualization possibilities
- only few structures considered
- eff conn and fun conn seem somewhat correlated, thus similar predictive performance of all approaches

Novelty:

- GCNs over gene expression was never applied here

Contribution of this work:

- Open source contribution, contributing a KerGNN implementation to PyTorch Geometric
- Contribution to AIDAconnect project and enhanced their pipeline
- Built plenty of pipelines on open-source data for the NeurDS group at CBS

6 Conclusion

Summary of Academic Study

Reference to Literature Review

Implications of Academic Study

Limitations of the Theory or Method of Research

Recommendations or Suggestions of Future Academic Study

- gene expression patterns within mouse brain and both possible hypothesis and tasks, and models over this
- gene knockout models and whether they can learn propagation of those?
- connection of FC and gene expression patterns and how to prove such interaction/correlation?
- possible gene knockout targets within mouse brain and possible structural influences

References

- H. Abdi and L. J. Williams. Principal component analysis. *Wiley interdisciplinary reviews: computational statistics*, 2(4):433–459, 2010.
- K. Achim, J.-B. Pettit, L. R. Saraiva, D. Gavriouchkina, T. Larsson, D. Arendt, and J. C. Marioni. High-throughput spatial mapping of single-cell rna-seq data to tissue of origin. *Nature biotechnology*, 33(5):503–509, 2015.
- J. D. Angrist and J.-S. Pischke. *Mostly harmless econometrics: An empiricist’s companion*. Princeton university press, 2009.
- A. Arnatkevičiūtė, B. D. Fulcher, R. Pocock, and A. Fornito. Hub connectivity, neuronal diversity, and gene expression in the *caenorhabditis elegans* connectome. *PLoS computational biology*, 14(2):e1005989, 2018.
- O. Aromolaran, T. Beder, M. Oswald, J. Oyelade, E. Adebiyi, and R. Koenig. Essential gene prediction in *drosophila melanogaster* using machine learning approaches based on sequence and functional features. *Computational and structural biotechnology journal*, 18:612–621, 2020.
- M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig, M. A. Harris, D. P. Hill, L. Issel-Tarver, A. Kasarskis, S. Lewis, J. C. Matese, J. E. Richardson, M. Ringwald, G. M. Rubin, and G. Sherlock. Gene ontology: tool for the unification of biology. *Nature Genetics*, 25(1):25–29, May 2000. doi: 10.1038/75556. URL <https://doi.org/10.1038/75556>.
- A. Q. Bauer, A. W. Kraft, G. A. Baxter, P. W. Wright, M. D. Reisman, A. R. Bice, J. J. Park, M. R. Bruchas, A. Z. Snyder, J.-M. Lee, et al. Effective connectivity measured using optogenetically evoked hemodynamic signals exhibits topography distinct from resting state functional connectivity in the mouse. *Cerebral Cortex*, 28(1):370–386, 2018.
- R. F. Betzel, J. D. Medaglia, A. E. Kahn, J. Soffer, D. R. Schonhaut, and D. S. Bassett. Structural, geometric and genetic factors predict interregional brain connectivity patterns probed by electrocorticography. *Nature biomedical engineering*, 3(11):902–916, 2019.
- M. Blum, H.-Y. Chang, S. Chuguransky, T. Grego, S. Kandasamy, A. Mitchell, G. Nuka, T. Paysan-Lafosse, M. Qureshi, S. Raj, L. Richardson, G. A. Salazar, L. Williams, P. Bork, A. Bridge, J. Gough, D. H. Haft, I. Letunic, A. Marchler-Bauer, H. Mi, D. A. Natale, M. Necci, C. A. Orengo, A. P. Pandurangan, C. Rivoire, C. J. A. Sigrist, I. Sillitoe, N. Thanki, P. D. Thomas, S. C. E. Tosatto, C. H. Wu, A. Bateman, and R. D. Finn. The InterPro protein families and domains database: 20 years on. *Nucleic Acids Research*, 49(D1):D344–D354, Nov. 2020. doi: 10.1093/nar/gkaa977. URL <https://doi.org/10.1093/nar/gkaa977>.
- J. W. Bohland, H. Bokil, S. D. Pathak, C.-K. Lee, L. Ng, C. Lau, C. Kuan, M. Hawrylycz, and P. P. Mitra. Clustering of spatial gene expression patterns in the mouse brain and comparison with classical neuroanatomy. *Methods*, 50(2):105–112, 2010.

- J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah. Signature verification using a "siamese" time delay neural network. *Advances in neural information processing systems*, 6, 1993.
- S. Carbon, E. Douglass, B. M. Good, D. R. Unni, N. L. Harris, C. J. Mungall, S. Basu, R. L. Chisholm, R. J. Dodson, E. Hartline, P. Fey, P. D. Thomas, L.-P. Albou, D. Ebert, M. J. Kesling, H. Mi, A. Muruganujan, X. Huang, T. Mushayahama, S. A. LaBonte, D. A. Siegele, G. Antonazzo, H. Attrill, N. H. Brown, P. Garapati, S. J. Marygold, V. Trovisco, G. dos Santos, K. Falls, C. Tabone, P. Zhou, J. L. Goodman, V. B. Strelets, J. Thurmond, P. Garmiri, R. Ishtiaq, M. Rodríguez-López, M. L. Acencio, M. Kuiper, A. Lægreid, C. Logie, R. C. Lovering, B. Kramarz, S. C. C. Saverimuttu, S. M. Pinheiro, H. Gunn, R. Su, K. E. Thurlow, M. Chibucos, M. Giglio, S. Nadendla, J. Munro, R. Jackson, M. J. Duesbury, N. Del-Toro, B. H. M. Meldal, K. Paneerselvam, L. Perfetto, P. Porras, S. Orchard, A. Shrivastava, H.-Y. Chang, R. D. Finn, A. L. Mitchell, N. D. Rawlings, L. Richardson, A. Sangrador-Vegas, J. A. Blake, K. R. Christie, M. E. Dolan, H. J. Drabkin, D. P. Hill, L. Ni, D. M. Sitnikov, M. A. Harris, S. G. Oliver, K. Rutherford, V. Wood, J. Hayles, J. Bähler, E. R. Bolton, J. L. D. Pons, M. R. Dwinell, G. T. Hayman, M. L. Kaldunski, A. E. Kwitek, S. J. F. Laulederkind, C. Plasterer, M. A. Tutaj, M. Vedi, S.-J. Wang, P. D'Eustachio, L. Matthews, J. P. Balhoff, S. A. Aleksander, M. J. Alexander, J. M. Cherry, S. R. Engel, F. Gondwe, K. Karra, S. R. Miyasato, R. S. Nash, M. Simison, M. S. Skrzypek, S. Weng, E. D. Wong, M. Feuermann, P. Gaudet, A. Morgat, E. Bakker, T. Z. Berardini, L. Reiser, S. Subramaniam, E. Huala, C. N. Arighi, A. Auchincloss, K. Axelsen, G. Argoud-Puy, A. Bateman, M.-C. Blatter, E. Boutet, E. Bowler, L. Breuza, A. Bridge, R. Britto, H. Bye-A-Jee, C. C. Casas, E. Coudert, P. Denny, A. Estreicher, M. L. Famiglietti, G. Georghiou, A. Gos, N. Gruaz-Gumowski, E. Hatton-Ellis, C. Hulo, A. Ignatchenko, F. Jungo, K. Laiho, P. L. Mercier, D. Lieberherr, A. Lock, Y. Lussi, A. MacDougall, M. Magrane, M. J. Martin, P. Masson, D. A. Natale, N. Hyka-Nouspikel, S. Orchard, I. Pedruzzi, L. Pourcel, S. Poux, S. Pundir, C. Rivoire, E. Speretta, S. Sundaram, N. Tyagi, K. Warner, R. Zaru, C. H. Wu, A. D. Diehl, J. N. Chan, C. Grove, R. Y. N. Lee, H.-M. Muller, D. Raciti, K. V. Auken, P. W. Sternberg, M. Berriman, M. Paulini, K. Howe, S. Gao, A. Wright, L. Stein, D. G. Howe, S. Toro, M. Westerfield, P. Jaiswal, L. Cooper, and J. Elser. The gene ontology resource: enriching a GOld mine. *Nucleic Acids Research*, 49(D1):D325–D334, Dec. 2020. doi: 10.1093/nar/gkaa1113. URL <https://doi.org/10.1093/nar/gkaa1113>.
- J. Chen, S. Suo, P. P. Tam, J.-D. J. Han, G. Peng, and N. Jing. Spatial transcriptomic analysis of cryosectioned tissue samples with geo-seq. *Nature protocols*, 12(3):566–580, 2017.
- J. Chen, A. Althagafi, and R. Hoechndorf. Predicting candidate genes from phenotypes, functions and anatomical site of expression. *Bioinformatics*, Oct. 2020. doi: 10.1093/bioinformatics/btaa879. URL <https://doi.org/10.1093/bioinformatics/btaa879>. advance access.
- D. Chicco. Siamese neural networks: An overview. *Artificial Neural Networks*, pages 73–94, 2021.
- M. D. Chikina, C. Huttenhower, C. T. Murphy, and O. G. Troyanskaya. Global prediction of tissue-specific gene expression and context-dependent gene networks in *caenorhabditis elegans*. *PLoS computational biology*, 5(6):e1000417, 2009.

- J. K. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio. Attention-based models for speech recognition. *Advances in neural information processing systems*, 28, 2015.
- T. L. Daigle, L. Madisen, T. A. Hage, M. T. Valley, U. Knoblich, R. S. Larsen, M. M. Takeno, L. Huang, H. Gu, R. Larsen, et al. A suite of transgenic driver and reporter mouse lines with enhanced brain-cell-type targeting and functionality. *Cell*, 174(2):465–480, 2018.
- P. Davis, M. Zarowiecki, V. Arnaboldi, A. Becerra, S. Cain, J. Chan, W. J. Chen, J. Cho, E. da Veiga Beltrame, S. Diamantakis, et al. Wormbase in 2022—data, processes, and tools for analyzing *caenorhabditis elegans*. *Genetics*, 220(4):iyac003, 2022.
- I. Diez and J. Sepulcre. Neurogenetic profiles delineate large-scale connectivity dynamics of the human brain. *Nature communications*, 9(1):1–10, 2018.
- A. Fakhry and S. Ji. High-resolution prediction of mouse brain connectivity using gene expression patterns. *Methods*, 73:71–78, 2015.
- A. Fakhry, T. Zeng, H. Peng, and S. Ji. Global analysis of gene expression and projection target correlations in the mouse brain. *Brain Informatics*, 2(2):107–117, 2015.
- A. Feng, C. You, S. Wang, and L. Tassiulas. Kergnns: Interpretable graph neural networks with graph kernels. *ArXiv Preprint: https://arxiv.org/abs/2201.00491*, 2022.
- M. Fey and J. E. Lenssen. Fast graph representation learning with pytorch geometric. *CoRR*, abs/1903.02428, 2019. URL <http://arxiv.org/abs/1903.02428>.
- A. Flores-Morales, H. Gullberg, L. Fernandez, N. Ståhlberg, N. H. Lee, B. Vennström, and G. Norstedt. Patterns of liver gene expression governed by $\text{tr}\beta$. *Molecular endocrinology*, 16(6):1257–1268, 2002.
- A. Fornito, A. Zalesky, and M. Breakspear. The connectomics of brain disorders. *Nature Reviews Neuroscience*, 16(3):159–172, 2015.
- K. Friston and C. Büchel. Attentional modulation of effective connectivity from v2 to v5/mt in humans. *Proceedings of the National Academy of Sciences*, 97(13):7591–7596, 2000.
- K. J. Friston. Functional and effective connectivity in neuroimaging: a synthesis. *Human brain mapping*, 2(1-2):56–78, 1994.
- K. J. Friston. Dysfunctional connectivity in schizophrenia. *World Psychiatry*, 1(2):66, 2002.
- K. J. Friston. Functional and effective connectivity: a review. *Brain connectivity*, 1(1):13–36, 2011.
- K. J. Friston, C. D. Frith, P. F. Liddle, and R. S. Frackowiak. Functional connectivity: the principal-component analysis of large (pet) data sets. *Journal of Cerebral Blood Flow & Metabolism*, 13(1):5–14, 1993.
- K. J. Friston, A. P. Holmes, K. J. Worsley, J.-P. Poline, C. D. Frith, and R. S. Frackowiak. Statistical parametric maps in functional imaging: a general linear approach. *Human brain mapping*, 2(4):189–210, 1994.

- B. D. Fulcher and A. Fornito. A transcriptional signature of hub connectivity in the mouse connectome. *Proceedings of the National Academy of Sciences*, 113(5):1435–1440, 2016.
- B. D. Fulcher, A. Arnatkeviciute, and A. Fornito. Overcoming false-positive gene-category enrichment in the analysis of spatially resolved transcriptomic brain atlas data. *Nature communications*, 12(1):1–13, 2021.
- J. Gillis and P. Pavlidis. “guilt by association” is the exception rather than the rule in gene networks. *PLoS Computational Biology*, 8(3):e1002444, Mar. 2012. doi: 10.1371/journal.pcbi.1002444. URL <https://doi.org/10.1371/journal.pcbi.1002444>.
- P. Goel, A. Kuceyeski, E. LoCastro, and A. Raj. Spatial patterns of genome-wide expression profiles reflect anatomic and fiber connectivity architecture of healthy human brain. *Human brain mapping*, 35(8):4204–4218, 2014.
- S. B. Hamida, S. Mendonça-Netto, T. M. Arefin, M. T. Nasseef, L.-J. Boulos, M. McNicholas, A. T. Ehrlich, E. Clarke, L. Moquin, A. Gratton, et al. Increased alcohol seeking in mice lacking gpr88 involves dysfunctional mesocorticolimbic networks. *Biological psychiatry*, 84(3):202–212, 2018.
- J. A. Harris, S. Mihalas, K. E. Hirokawa, J. D. Whitesell, H. Choi, A. Bernard, P. Bohn, S. Caldejon, L. Casal, A. Cho, et al. Hierarchical organization of cortical and thalamic connectivity. *Nature*, 575 (7781):195–202, 2019.
- M. Hawrylycz, R. A. Baldock, A. Burger, T. Hashikawa, G. A. Johnson, M. Martone, L. Ng, C. Lau, S. D. Larsen, J. Nissanov, L. Puelles, S. Ruffins, F. Verbeek, I. Zaslavsky, and J. Boline. Digital Atlasing and Standardization in the Mouse Brain. *PLoS Computational Biology*, 7(2):e1001065, Feb. 2011. ISSN 1553-7358. doi: 10.1371/journal.pcbi.1001065. URL <https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1001065>. Publisher: Public Library of Science.
- J. I. Herschkowitz, K. Simin, V. J. Weigman, I. Mikaelian, J. Usary, Z. Hu, K. E. Rasmussen, L. P. Jones, S. Assefnia, S. Chandrasekharan, et al. Identification of conserved gene expression features between murine mammary carcinoma models and human breast tumors. *Genome biology*, 8(5):1–17, 2007.
- J. R. Hershey and P. A. Olsen. Approximating the kullback leibler divergence between gaussian mixture models. In *2007 IEEE International Conference on Acoustics, Speech and Signal Processing- ICASSP'07*, volume 4, pages IV–317. IEEE, 2007.
- T. Hinnerichs and R. Hoehndorf. Dti-voodoo: machine learning over interaction networks and ontology-based background knowledge predicts drug–target interactions. *Bioinformatics*, 37(24):4835–4843, 2021.
- R. Hoehndorf, P. N. Schofield, and G. V. Gkoutos. PhenomeNET: a whole-phenome approach to disease gene discovery. *Nucleic Acids Research*, 39(18):e119–e119, July 2011. doi: 10.1093/nar/gkr538. URL <https://doi.org/10.1093/nar/gkr538>.

- L. A. Jeni, J. F. Cohn, and F. De La Torre. Facing imbalanced data–recommendations for the use of performance metrics. In *2013 Humaine Association Conference on Affective Computing and Intelligent Interaction*, pages 245–251, 2013a. doi: 10.1109/ACII.2013.47.
- L. A. Jeni, J. F. Cohn, and F. De La Torre. Facing imbalanced data–recommendations for the use of performance metrics. In *2013 Humaine association conference on affective computing and intelligent interaction*, pages 245–251. IEEE, 2013b.
- A. Kaufman, G. Dror, I. Meilijson, and E. Ruppin. Gene expression of *caenorhabditis elegans* neurons carries information on their synaptic connectivity. *PLoS computational biology*, 2(12):e167, 2006.
- R. Ke, M. Mignardi, A. Pacureanu, J. Svedlund, J. Botling, C. Wählby, and M. Nilsson. In situ sequencing for rna analysis in preserved tissue and cells. *Nature methods*, 10(9):857–860, 2013.
- S. T. Kelly and M. A. Black. graphsim: An R package for simulating gene expression data from graph structures of biological pathways. *Journal of Open Source Software*, 5(51):2161, July 2020. ISSN 2475-9066. doi: 10.21105/joss.02161. URL <https://joss.theoj.org/papers/10.21105/joss.02161>.
- D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2015.
- T. N. Kipf and M. Welling. Semi-supervised classification with graph convolutional networks. *CoRR*, abs/1609.02907, 2016. URL <http://arxiv.org/abs/1609.02907>.
- M. V. Kuleshov, M. R. Jones, A. D. Rouillard, N. F. Fernandez, Q. Duan, Z. Wang, S. Koplev, S. L. Jenkins, K. M. Jagodnik, A. Lachmann, et al. Enrichr: a comprehensive gene set enrichment analysis web server 2016 update. *Nucleic acids research*, 44(W1):W90–W97, 2016.
- M. Kulmanov and R. Hoehndorf. DeepGOPlus: improved protein function prediction from sequence. *Bioinformatics*, 36(2):422–429, 07 2019. ISSN 1367-4803. doi: 10.1093/bioinformatics/btz595. URL <https://doi.org/10.1093/bioinformatics/btz595>.
- S. H. Lecker, R. T. Jagoe, A. Gilbert, M. Gomes, V. Baracos, J. Bailey, S. R. Price, W. E. Mitch, and A. L. Goldberg. Multiple types of skeletal muscle atrophy involve a common program of changes in gene expression. *The FASEB Journal*, 18(1):39–51, 2004.
- R. Y. Lee and P. W. Sternberg. Building a cell and anatomy ontology of *caenorhabditis elegans*. *Comparative and Functional Genomics*, 4(1):121–126, 2003.
- E. S. Lein, M. J. Hawrylycz, N. Ao, M. Ayres, A. Bensinger, A. Bernard, A. F. Boe, M. S. Boguski, K. S. Brockway, E. J. Byrnes, L. Chen, L. Chen, T.-M. Chen, M. C. Chin, J. Chong, B. E. Crook, A. Czaplinska, C. N. Dang, S. Datta, N. R. Dee, A. L. Desaki, T. Desta, E. Diep, T. A. Dolbeare, M. J. Donelan, H.-W. Dong, J. G. Dougherty, B. J. Duncan, A. J. Ebbert, G. Eichele, L. K. Estin, C. Faber, B. A. Facer, R. Fields, S. R. Fischer, T. P. Fliss, C. Frenzley, S. N. Gates, K. J. Glattfelder, K. R. Halverson, M. R. Hart, J. G. Hohmann, M. P. Howell, D. P. Jeung, R. A. Johnson, P. T. Karr, R. Kawal, J. M. Kidney, R. H. Knapik, C. L. Kuan, J. H. Lake, A. R. Laramee, K. D. Larsen, C. Lau, T. A. Lemon, A. J. Liang, Y. Liu, L. T. Luong, J. Michaels, J. J. Morgan, R. J. Morgan,

- M. T. Mortrud, N. F. Mosqueda, L. L. Ng, R. Ng, G. J. Orta, C. C. Overly, T. H. Pak, S. E. Parry, S. D. Pathak, O. C. Pearson, R. B. Puchalski, Z. L. Riley, H. R. Rockett, S. A. Rowland, J. J. Royall, M. J. Ruiz, N. R. Sarno, K. Schaffnit, N. V. Shapovalova, T. Sivisay, C. R. Slaughterbeck, S. C. Smith, K. A. Smith, B. I. Smith, A. J. Sodt, N. N. Stewart, K.-R. Stumpf, S. M. Sunkin, M. Sutram, A. Tam, C. D. Teemer, C. Thaller, C. L. Thompson, L. R. Varnam, A. Visel, R. M. Whitlock, P. E. Wohnoutka, C. K. Wolkey, V. Y. Wong, M. Wood, M. B. Yaylaoglu, R. C. Young, B. L. Youngstrom, X. F. Yuan, B. Zhang, T. A. Zwingman, and A. R. Jones. Genome-wide atlas of gene expression in the adult mouse brain. *Nature*, 445(7124):168–176, Dec. 2006. doi: 10.1038/nature05453. URL <https://doi.org/10.1038/nature05453>.
- A. Leman and B. Weisfeiler. A reduction of a graph to a canonical form and an algebra arising during this reduction. *Nauchno-Technicheskaya Informatsiya*, 2(9):12–16, 1968.
- M. E. Lepperød, T. Stöber, T. Hafting, M. Fyhn, and K. P. Kording. Inferring causal connectivity from pairwise recordings and optogenetics. *bioRxiv*, page 463760, 2018.
- G. Li, M. Müller, A. Thabet, and B. Ghanem. Deepgcns: Can gcns go as deep as cnns? In *The IEEE International Conference on Computer Vision (ICCV)*, 2019.
- G. Li, C. Xiong, A. Thabet, and B. Ghanem. Deepergcn: All you need to train deeper gcns. *CoRR*, abs/2006.07739, 2020.
- J. Z. Li, B. G. Bunney, F. Meng, M. H. Hagenauer, D. M. Walsh, M. P. Vawter, S. J. Evans, P. V. Choudary, P. Cartagena, J. D. Barchas, et al. Circadian patterns of gene expression in the human brain and disruption in major depressive disorder. *Proceedings of the National Academy of Sciences*, 110(24):9950–9955, 2013.
- W.-C. Lo, C.-C. Lee, C.-Y. Lee, and P.-C. Lyu. Cpdb: a database of circular permutation in proteins. *Nucleic acids research*, 37(suppl_1):D328–D332, 2009.
- L. McInnes, J. Healy, and J. Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.
- T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. *CoRR*, abs/1310.4546, 2013. URL <http://arxiv.org/abs/1310.4546>.
- N. Milyaev, D. Osumi-Sutherland, S. Reeve, N. Burton, R. A. Baldock, and J. D. Armstrong. The virtual fly brain browser and query interface. *Bioinformatics*, 28(3):411–415, 2012.
- modENCODE Consortium, S. Roy, J. Ernst, P. V. Kharchenko, P. Kheradpour, N. Negre, M. L. Eaton, J. M. Landolin, C. A. Bristow, L. Ma, et al. Identification of functional elements and regulatory circuits by drosophila modencode. *Science*, 330(6012):1787–1797, 2010.
- J. R. Moffitt, J. Hao, G. Wang, K. H. Chen, H. P. Babcock, and X. Zhuang. High-throughput single-cell gene-expression profiling with multiplexed error-robust fluorescence in situ hybridization. *Proceedings of the National Academy of Sciences*, 113(39):11046–11051, 2016.

- C. Morris, M. Ritzert, M. Fey, W. L. Hamilton, J. E. Lenssen, G. Rattan, and M. Grohe. Weisfeiler and leman go neural: Higher-order graph neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33-01, pages 4602–4609, 2019.
- C. J. Mungall, C. Torniai, G. V. Gkoutos, S. E. Lewis, and M. A. Haendel. Uberon, an integrative multi-species anatomy ontology. *Genome biology*, 13(1):1–20, 2012.
- A. Ng et al. Sparse autoencoder. *CS294A Lecture notes*, 72(2011):1–19, 2011.
- L. Ng, S. Pathak, C. Kuan, C. Lau, H.-w. Dong, A. Sodt, C. Dang, B. Avants, P. Yushkevich, J. Gee, et al. Neuroinformatics for genome-wide 3-d gene expression mapping in the mouse brain. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 4(3):382–393, 2007.
- A. Nieto-Castanon. *Handbook of functional connectivity Magnetic Resonance Imaging methods in CONN*. Hilbert Press, 2020.
- W. S. Noble. What is a support vector machine? *Nature biotechnology*, 24(12):1565–1567, 2006.
- S. W. Oh, J. A. Harris, L. Ng, B. Winslow, N. Cain, S. Mihalas, Q. Wang, C. Lau, L. Kuan, A. M. Henry, et al. A mesoscale connectome of the mouse brain. *Nature*, 508(7495):207–214, 2014.
- S. Oliver. Guilt-by-association goes global. *Nature*, 403(6770):601–602, Feb. 2000. doi: 10.1038/35001165. URL <https://doi.org/10.1038/35001165>.
- N. Pallast, M. Diedenhofen, S. Blaschke, F. Wieters, D. Wiedermann, M. Hoehn, G. R. Fink, and M. Aswendt. Processing pipeline for atlas-based imaging data analysis of structural and functional mouse brain MRI (AIDAmri). *Frontiers in Neuroinformatics*, 13, June 2019. doi: 10.3389/fninf.2019.00042. URL <https://doi.org/10.3389/fninf.2019.00042>.
- L. Parkes, B. Fulcher, M. Yücel, and A. Fornito. Transcriptional signatures of connectomic subregions of the human striatum. *Genes, Brain and Behavior*, 16(7):647–663, 2017.
- G. Partel, M. M. Hilscher, G. Milli, L. Solorzano, A. H. Klemm, M. Nilsson, and C. Wählby. Automated identification of the mouse brain’s spatial compartments from *in situ* sequencing data. *BMC Biology*, 18(1), Oct. 2020. doi: 10.1186/s12915-020-00874-5. URL <https://doi.org/10.1186/s12915-020-00874-5>.
- A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d’Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. URL <http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf>.
- P. Pavlidis and W. S. Noble. Analysis of strain and regional variation in gene expression in mouse brain. *Genome Biology*, 2(10):research0042.1, Sept. 2001. ISSN 1474-760X. doi: 10.1186/gb-2001-2-10-research0042. URL <https://doi.org/10.1186/gb-2001-2-10-research0042>.

- J. Pearl. *Causality*. Cambridge university press, 2009.
- A. Ramasamy, D. Trabzuni, S. Guelfi, V. Varghese, C. Smith, R. Walker, T. De, L. Coin, R. De Silva, M. R. Cookson, et al. Genetic variability in the regulation of gene expression in ten regions of the human brain. *Nature neuroscience*, 17(10):1418–1428, 2014.
- C. H. Rankin. From gene to identified neuron to behaviour in *caenorhabditis elegans*. *Nature Reviews Genetics*, 3(8):622–630, 2002.
- S. Razick, G. Magklaras, and I. M. Donaldson. irefindex: a consolidated protein interaction database with provenance. *BMC bioinformatics*, 9(1):1–19, 2008.
- P. Resnik. Using information content to evaluate semantic similarity in a taxonomy. *arXiv preprint cmp-lg/9511007*, 1995.
- J. Richiardi, A. Altmann, A.-C. Milazzo, C. Chang, M. M. Chakravarty, T. Banaschewski, G. J. Barker, A. L. Bokde, U. Bromberg, C. Büchel, et al. Correlated gene expression supports synchronous activity in brain networks. *Science*, 348(6240):1241–1244, 2015.
- I. Roberti, M. Lovino, S. Di Cataldo, E. Ficarra, and G. Urgese. Exploiting gene expression profiles for the automated prediction of connectivity between brain regions. *International journal of molecular sciences*, 20(8):2035, 2019.
- P. Roland, C. Graufelds, J. Wählén, L. Ingelman, M. Andersson, A. Ledberg, J. Pedersen, S. Åkerman, A. Dabringhaus, and K. Zilles. Human brain atlas: for high-resolution functional and anatomical mapping. *Human Brain Mapping*, 1(3):173–184, 1994.
- M. Rubinov, R. J. Ypma, C. Watson, and E. T. Bullmore. Wiring cost and topological participation of the mouse brain connectome. *Proceedings of the National Academy of Sciences*, 112(32):10032–10037, 2015.
- T. Sainburg, L. McInnes, and T. Q. Gentner. Parametric umap embeddings for representation and semisupervised learning. *Neural Computation*, 33(11):2881–2907, 2021.
- R. Schulte-Sasse, S. Budach, D. Hnisz, and A. Marsico. Integration of multiomics data with graph convolutional networks to identify new cancer genes and their associated molecular mechanisms. *Nature Machine Intelligence*, 3(6):513–526, 2021.
- K. Sengupta, A. Gambin, S. Basu, and D. Plewczynski. Multinet: A diffusion-based approach to assign directionality in protein interactions using a consensus of eight protein interaction datasets. In *Proceedings of International Conference on Frontiers in Computing and Systems*, pages 13–20. Springer, 2023.
- N. Shervashidze, S. Vishwanathan, T. Petri, K. Mehlhorn, and K. Borgwardt. Efficient graphlet kernels for large graph comparison. In *Artificial intelligence and statistics*, pages 488–495. PMLR, 2009.
- C. L. Smith and J. T. Eppig. The mammalian phenotype ontology: enabling robust annotation and comparative analysis. *Wiley Interdisciplinary Reviews: Systems Biology and Medicine*, 1(3):390–399, Nov. 2009. doi: 10.1002/wsbm.44. URL <https://doi.org/10.1002/wsbm.44>.

- O. Sporns. *Networks of the Brain*. MIT press, 2016.
- A. Subramanian, H. Kuehn, J. Gould, P. Tamayo, and J. P. Mesirov. Gsea-p: a desktop application for gene set enrichment analysis. *Bioinformatics*, 23(23):3251–3253, 2007.
- H. Sun and O. Hobert. Temporal transitions in the post-mitotic nervous system of *caenorhabditis elegans*. *Nature*, 600(7887):93–99, 2021.
- D. Szklarczyk, A. Franceschini, S. Wyder, K. Forslund, D. Heller, J. Huerta-Cepas, M. Simonovic, A. Roth, A. Santos, K. P. Tsafou, M. Kuhn, Peer, L. J. Jensen, and C. von Mering. STRING v10: protein–protein interaction networks, integrated over the tree of life. *Nucleic Acids Research*, 43(D1): D447–D452, Oct. 2014. doi: 10.1093/nar/gku1003. URL <https://doi.org/10.1093/nar/gku1003>.
- N. Takata, N. Sato, Y. Komaki, H. Okano, and K. F. Tanaka. Flexible annotation atlas of the mouse brain: combining and dividing brain structures of the Allen Brain Atlas while maintaining anatomical hierarchy. *Scientific Reports*, 11(1):6234, Mar. 2021. ISSN 2045-2322. doi: 10.1038/s41598-021-85807-0. URL <https://www.nature.com/articles/s41598-021-85807-0>.
- Bandiera_abtest: a Cc_license_type: cc_by Cg_type: Nature Research Journals Number: 1 Primary_atype: Research Publisher: Nature Publishing Group Subject_term: Brain;Functional magnetic resonance imaging;Neuroscience Subject_term_id: brain;functional-magnetic-resonance-imaging;neuroscience.
- H. Y. Tan, A. G. Chen, B. Kolachana, J. A. Apud, V. S. Mattay, J. H. Callicott, Q. Chen, and D. R. Weinberger. Effective connectivity of akt1-mediated dopaminergic working memory networks and pharmacogenetics of anti-dopaminergic treatment. *Brain*, 135(5):1436–1445, 2012.
- K. Tomczak, P. Czerwińska, and M. Wiznerowicz. Review the cancer genome atlas (tcga): an immeasurable source of knowledge. *Contemporary Oncology/Współczesna Onkologia*, 2015(1):68–77, 2015.
- M. Trebacz, Z. Shams, M. Jamnik, P. Scherer, N. Simidjievski, H. A. Terre, and P. Liò. Using ontology embeddings for structural inductive bias in gene expression data analysis. *CoRR*, abs/2011.10998, 2020.
- N. A. Twine, K. Janitz, M. R. Wilkins, and M. Janitz. Whole transcriptome sequencing reveals gene expression and splicing differences in brain regions affected by alzheimer’s disease. *PloS one*, 6(1): e16266, 2011.
- S. L. Valk, T. Xu, D. S. Margulies, S. K. Masouleh, C. Paquola, A. Goulas, P. Kochunov, J. Smallwood, B. T. T. Yeo, B. C. Bernhardt, and S. B. Eickhoff. Shaping brain structure: Genetic and phylogenetic axes of macroscale organization of cortical thickness. *Science Advances*, 6(39):eabb3417, 2020. doi: 10.1126/sciadv.abb3417. URL <https://www.science.org/doi/abs/10.1126/sciadv.abb3417>.
- L. Van der Maaten and G. Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

- D. C. Van Essen, S. M. Smith, D. M. Barch, T. E. Behrens, E. Yacoub, K. Ugurbil, W.-M. H. Consortium, et al. The wu-minn human connectome project: an overview. *Neuroimage*, 80:62–79, 2013.
- V. Varadan, D. M. Miller III, and D. Anastassiou. Computational inference of the molecular logic for synaptic connectivity in *c. elegans*. *Bioinformatics*, 22(14):e497–e506, 2006.
- A. Vazquez, A. Flammini, A. Maritan, and A. Vespignani. Global protein function prediction from protein-protein interaction networks. *Nature Biotechnology*, 21(6):697–700, May 2003. doi: 10.1038/nbt825. URL <https://doi.org/10.1038/nbt825>.
- P. E. Vértes, T. Rittman, K. J. Whitaker, R. Romero-Garcia, F. Váša, M. G. Kitzbichler, K. Wagstyl, P. Fonagy, R. J. Dolan, P. B. Jones, et al. Gene transcription profiles associated with inter-modular hubs and connection distance in human functional magnetic resonance imaging networks. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 371(1705):20150362, 2016.
- Q. Wang, S.-L. Ding, Y. Li, J. Royall, D. Feng, P. Lesnar, N. Graddis, M. Naeemi, B. Facer, A. Ho, et al. The allen mouse brain common coordinate framework: a 3d reference atlas. *Cell*, 181(4):936–953, 2020.
- T. Wang, W. Shao, Z. Huang, H. Tang, J. Zhang, Z. Ding, and K. Huang. Mogonet integrates multi-omics data using graph convolutional networks allowing patient classification and biomarker identification. *Nature Communications*, 12(1):1–13, 2021.
- W. Wang, R. Han, M. Zhang, Y. Wang, T. Wang, Y. Wang, X. Shang, and J. Peng. A network-based method for brain disease gene prediction by integrating brain connectome and molecular network. *Briefings in Bioinformatics*, 23(1):bab459, 2022.
- J. D. Watson. The human genome project: past, present, and future. *Science*, 248(4951):44–49, 1990.
- C. W. Whitfield, A.-M. Cziko, and G. E. Robinson. Gene expression profiles in the brain predict behavior in individual honey bees. *Science*, 302(5643):296–299, 2003.
- S. Wold, K. Esbensen, and P. Geladi. Principal component analysis. *Chemometrics and intelligent laboratory systems*, 2(1-3):37–52, 1987.
- K. Xu, W. Hu, J. Leskovec, and S. Jegelka. How powerful are graph neural networks? *arXiv preprint arXiv:1810.00826*, 2018.
- Y. Yang, Q. Fang, and H.-B. Shen. Predicting gene regulatory interactions based on spatial gene expression data and deep learning. *PLoS computational biology*, 15(9):e1007324, 2019.
- M. A. Zapala, I. Hovatta, J. A. Ellison, L. Wodicka, J. A. Del Rio, R. Tennant, W. Tynan, R. S. Broide, R. Helton, B. S. Stoveken, et al. Adult mouse brain gene expression patterns bear an embryologic imprint. *Proceedings of the National Academy of Sciences*, 102(29):10357–10362, 2005.
- T. Zeng, R. Li, R. Mukkamala, J. Ye, and S. Ji. Deep convolutional neural networks for annotating gene expression patterns in the mouse brain. *BMC Bioinformatics*, 16(1):147, May 2015. ISSN 1471-2105. doi: 10.1186/s12859-015-0553-9. URL <https://doi.org/10.1186/s12859-015-0553-9>.

V. Zerbi, M. Pagani, M. Markicevic, M. Matteoli, D. Pozzi, M. Fagiolini, Y. Bozzi, A. Galbusera, M. L. Scattoni, G. Provenzano, A. Banerjee, F. Helmchen, M. A. Basson, J. Ellegood, J. P. Lerch, M. Rudin, A. Gozzi, and N. Wenderoth. Brain mapping across 16 autism mouse models reveals a spectrum of functional connectivity subtypes. *Molecular Psychiatry*, Aug. 2021. doi: 10.1038/s41380-021-01245-4. URL <https://doi.org/10.1038/s41380-021-01245-4>.

M. Zitnik and J. Leskovec. Predicting multicellular function through multi-layer tissue networks. *CoRR*, abs/1707.04638, 2017. URL <http://arxiv.org/abs/1707.04638>.