NORTHWESTERN UNIVERSITY

Geometric deep learning in neuroimaging and human reward behavior

A DISSERTATION

SUBMITTED TO THE GRADUATE SCHOOL
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

for the degree

DOCTOR OF PHILOSOPHY

Field of Electrical Engineering

By

Emanuel A. Azcona

EVANSTON, ILLINOIS

August 2022

Abstract

Geometric deep learning in neuroimaging and human reward behavior

Emanuel A. Azcona

In recent years, machine learning on graphs (or networks) has gone from a niche topic with only a few active researchers worldwide, to a heavily invested field with novel use cases for dealing with relationships and/or interactions within complex systems in the natural and social sciences. Traditionally, choosing the right type of model for leveraging the inductive biases of the task at-hand is a crucial step in machine learning scenarios (mostly supervised) because "there ain't no such thing as a free lunch," as the saying goes. Convolutional neural networks (CNNs) in particular, have been incredibly effective in numerous image processing problems and are becoming the de facto choice for considering data with grid-like topology. In a graph setting, where a grid-like structure is not always a guarantee, it is useful to leverage *relational* inductive biases within deep learning architectures in order to build systems that can learn, reason, and generalize from graph data. Graph-structured data is ubiquitous and all around us; often real-world entities are characterized by

their connection(s) to other things. Recent advances in research on graph representation learning, particularly geometric deep learning (GDL), has led to a plethora of techniques for deep graph embeddings, generalizations of CNNs to graph data, and a reframing of neural message-passing approaches to graphs inspired by belief propagation. By maintaining the notion of representation or feature learning, and learning by local gradient-descent type methods, advances in GDL have led to new state-of-the-art results in several domains, including social network analysis, 3D surface manifold modeling, mapping/way-finding, molecular modeling, question answering, and recommender systems.

The goal of this body of work was to provide a technical synthesis of GDL, through some methodological foundations and a demonstration of various benefits of GDL in structural neuroimaging and neuropsychological indicators, specifically human reward behavior. We begin with a discussion of GDL, specifically through GNN formulation, which has become amongst one of the fastest-growing paradigms for deep learning on graphs. Then we provide novel use cases in the analysis of human brain shape using 3D mesh surface manifolds to improve upon the state-of-the-art in machine learning for the classification of Alzheimer's disease and generating 3D brain models that are based on phenotypic priors. This thesis concludes with a new advancement in modeling human reward behavior as heterogeneous graphs (i.e., varying node/edge types), specifically using a portfolio of neurocognitive features to describe human preference towards a stimulus set, which are captured using a non-operant picture rating task across multiple distinct cohorts of human participants. Although comparatively nascent to other graph-based methods in the biomedical arena, the success of deep graph embeddings provided through GDL continues to showcase the advantages of graph representation learning in neuroimaging and computational cognitive science.

Acknowledgements

I always have trouble with this part. It's not because I don't know who to thank or acknowledge. It's because I hate goodbyes. I don't want to leave anyone out. Someone suggested I start with myself. So with that, I want to start off by saying: Emanuel I'm proud of you. After everything you have been through, everything that's happened, you kept moving along. You kept living. You kept laughing. Every time you smile, it makes me so happy. If that ever goes away again, I'm not going to miss it. I'm going to beg for it. A special someone reminded that if you're nervous, it probably means you care.

I'd first like to acknowledge both of my parents, Ramona Castillo and Rafael Azcona, whose support throughout my entire life has been instrumental to getting *us* to where *we* are today. I know I don't ever say this enough but I love you both. People always feel the need to ask if being an only child at home was lonely at times, but the truth is I grew up with my two best friends. Not everyone gets to say that. I can't thank you enough.

To my tití, Ceila Mendez, and the rest of the family, thank you for always being there for us. Some get just one, or none at all, but I was lucky enough to have another mother, a second family, and another place to call home just up the street.

To Shawneece Bailey, Tara Lao, and the rest of the Higher Education Opportunity Program (HEOP) family: thank you for giving this bodega kid from Queens a chance. Tara, I never told you this, but I played hooky from high school the day of my admissions interview with you. I promise it was for a good reason. You see, I only had 30 minutes between the end of the school day and getting to NYU on time. I had no choice but to run out 2 hours early from one of the emergency exits next to one of the construction workshop classrooms that day to make it to you on time. Thank you for always letting me sleep in your office, even after you both strongly advised against my habit of pulling constant all-nighters to get my work done. My mind says, "I told you so," but my body will probably regret it later. My family was the soil, but you two were the fertilizer that gave me the *opportunity* to grow.

I'd like to thank my research mentors throughout the years for always looking out for me and molding me into the researcher I am today. If and when you see my name out there it's okay if you shake your head and think to yourself, "I'm partially responsible for this." I would like to thank my advisor, Professor Aggelos Katsaggelos, for being a friend, a mentor, and something of a father figure away from home during my graduate school experience. Something just felt "right" when I visited Northwestern in March of 2017. I'm glad I wasn't wrong. I would also like to thank the other members of my committee, Hans Breiter, Ajay Kurani, and Todd Parrish for their contributions and guidance throughout the years.

Finally, I'd like to thank my colleagues and friends who have been a part of my work and my life in a variety of ways. Semih Barutçu, Samuel Lopez, and Arjun

Punjabi; some friends are brothers who find each other. Scout Wolfe, thank you for holding me together, reminding me to treat myself, and to be the "dancing queen." Jose Balbuena, Nathaniel Grammel, Juan Gabriel Serra Pérez, Florian Schiffers, Zaid Qureshi, Cindy Zhang, and *many*, *many* more: thank you for reminding me that family is more than just blood.

Dedication

This work is dedicated to my late grandmother, Modesta Antonia Vásquez, or as I knew her: Mamá. I miss you. I never did get to meet him, but he must have been pretty great if I am his namesake. Tell Papá, "job's finished."

Table of Contents

List of abbreviations

ABCD: Adolescent Brain Cognitive Development. 194

ABIDE: Autism Brain Imaging Data Exchange. 73

AD: Alzheimer's disease. 20, 23, 25, 26, 29, 30, 52, 58, 59, 61–65, 67, 68, 70–74, 80, 81, 83, 88, 90–92, 95–107, 120–122, 124–128, 130–138, 142–147, 150, 151

ADLs: activities of daily living. 58, 59, 125

ADNI: Alzheimer's Disease Neuroimaging Initiative. 67, 73, 74, 88, 91, 93, 97, 121, 122, 124, 138

AI: artificial intelligence. 26

AMHA: Automated Mental Health Assessment Study. 158, 159, 161, 162, 164, 175–178, 181, 184, 185, 187–194, 197, 198, 200

AUC: Area Under the Curve. 91, 125, 126, 130–132, 135, 150, 172

BCE: binary cross-entropy. 89, 118, 124

BN: batch normalization. 81, 82, 113, 117, 128, 129, 131, 132

CADDementia: Computer-Aided Diagnosis of Dementia. 67

CAM: class activation map. 23, 87, 88, 92, 93, 100, 101, 106, 107, 121, 132–135, 143, 145, 150, 151

CI: confidence interval. 174, 183–185, 189, 192, 199, 201

CNN: convolutional neural network. 3, 4, 24–26, 47, 49, 53, 57, 73, 87, 90, 91, 98, 100, 103, 107, 109, 110, 143, 148, 149, 218–221, 255–258

CoMA: convolutional mesh autoencoder. 73, 104, 106, 118, 119

CSF: cerebrospinal fluid. 73, 77, 91

CT: computed tomography. 64

CVAE: conditional variational autoencoder. 99, 101, 103, 106, 118, 119, 136, 138, 144, 147

DFT: discrete Fourier transform. 32, 33, 41, 43

DL: deep learning. 47, 68, 71, 80–82, 97–100, 102, 112, 114, 208

DSP: digital signal processing. 29–31, 33, 34, 36, 38, 39, 43, 44, 49, 53, 55, 209

EBS: Emotion and Behavior Study. 158–161, 175–178, 181, 184, 185, 187–190, 192–194, 196, 200

ELU: exponential linear unit. 113, 114, 117, 128, 129, 131, 132

FDG: fluorodeoxyglucose. 90, 91

FIR: finite impulse response. 31, 33

GAN: generative adversarial network. 105

GAP: global average pooling. 87, 114, 115, 117

GAT: graph attention network. 210, 212

GCN: graph convolutional network. 22, 23, 25, 29, 49–52, 56, 57, 71, 73, 75, 81–83, 88–92, 100, 101, 103, 106, 113, 128, 150, 210

GDL: geometric deep learning. 4, 5, 29, 56, 98, 105

GFT: graph Fourier transform. 40–43, 47, 48, 51

GNN: graph neural network. 4, 28, 51, 52, 56, 57, 202, 204, 208–210, 212, 213, 222–224

GPU: graphics processing unit. 144, 147

GSP: graph signal processing. 22, 29–31, 36, 38, 40, 43, 44, 49, 50, 75

HC: healthy control. 20, 62, 63, 65, 67, 68, 71–73, 80, 81, 88, 91, 92, 95, 97, 98, 100, 107, 118, 121, 122, 124, 126, 127, 131, 133, 134, 136–138, 143, 146, 147, 150

HGT: heterogeneous graph transformer. 24, 202, 204, 205, 210–214, 222, 223

IAPS: International Affective Picture System. 20, 24, 160, 165, 175, 178, 179, 181, 184–187, 196, 202–204, 215–222

IDFT: inverse discrete Fourier transform. 33

iff: if and only if. 111, 254

IGFT: inverse graph Fourier transform. 42, 47, 48, 76

IQR: interquartile range. 164, 174, 183, 199, 201

K-S: Kolmogorov-Smirnov. 175, 184, 186

KL: Kullback–Leibler. 105, 119, 120

LH: left hemisphere. 81, 93, 122, 125, 133–135, 137, 150

LSI: linear shift-invariant. 36, 39, 40, 254

MAE: mean absolute error. 120

MCI: mild cognitive impairment. 20, 59, 60, 64, 67, 74, 93, 98–100, 120–122, 124, 126, 127, 131, 132, 147

MGH-SQ: MGH Phenotype Genotype Project in Addiction and Mood Disorders symptom questionnaire. 160

MICCAI: Medical Image Computing and Computer-Assisted Intervention. 72

ML: machine learning. 26, 73, 74, 92, 202, 218, 255

MLP: multilayer perceptron. 23, 53, 82, 83, 89, 90, 109, 110, 112, 116–118, 128–130, 210, 219, 220, 222, 255–257

MMSE: mini-mental state examination. 63

MPNN: message passing neural network. 26, 29, 52–54, 56, 203

MRI: magnetic resonance imaging. 22, 25, 26, 57, 58, 61, 63–66, 68, 69, 73, 74, 77, 88, 90–92, 95–97, 100, 103, 121, 122, 146

MS: mesh sampling. 115, 116

MTL: medial temporal lobe. 62

NN: neural network. 26, 47, 53, 54, 79–81, 83, 95, 97–99, 105, 109–113, 147, 149, 208, 210, 220, 222, 258, 259

NYC: New York City. 22, 31, 32

OAG: Open Academic Graph. 24, 203, 204, 208

OASIS: Open-Access Series of Imaging Studies. 66

PET: positron emission tomography. 61, 64, 68, 73, 74, 90, 91, 105, 136

PHQ-9: Patient Health Questionnaire. 160, 216

QA: quality assessment. 94

ReLU: rectified linear unit. 81, 82, 88, 113, 114

ResBlock: residual learning block. 23, 80–82, 89, 112–116, 123, 124, 128, 129, 131, 149

RF: random forest. 90

RGB: red-green-blue. 30, 32, 83

RH: right hemisphere. 81, 93, 122, 125, 133–135, 137, 150

ROC: receiver operating characteristic. 91, 126, 131, 150

ROI: region of interest. 66, 74, 92, 101

RPT: relative preference theory. 20, 24, 153–156, 163–166, 169, 171, 173–175, 179, 180, 183–188, 190–194, 196–199, 206, 217, 220, 222

SD: standard deviation. 174, 185

SEM: standard error of the mean. 174, 184, 185

ShapeMI: Shape in Medical Imaging. 72

SOTA: state-of-the-art. 44, 54, 71, 90, 143, 148

SpiralConv: spiral convolution. 116

STAI: Spielberger State-Trait Anxiety Inventory. 160, 216

SVD: singular value decomposition. 55

SVM: support vector machine. 73, 208

TP: true positive. 23, 74, 92, 93, 96, 100, 101, 121, 133, 134

trimesh: triangular mesh. 20, 22, 23, 25, 29, 44–46, 57, 71, 73–75, 77–81, 88, 89, 91–104, 106, 107, 109–112, 114–120, 122, 123, 125, 129, 131–134, 136–138, 143–147, 202, 224

TV: total variation. 44

US: United States. 58, 59, 158, 194

VAE: variational autoencoder. 105, 106, 118

WM: white matter. 79, 81

List of Tables

List of Figures

Preface

The first study in this text focuses on analyzing brain morphology in the context of Alzheimer's disease (AD) and its early *in-vivo* classification based on human brain shape. Specifically, 2D mesh manifolds of individual brain structures are extracted from segmented 3D MRI volumes and used to train graph convolutional networks (GCNs) from [**3**] on the AD classification task. From a computational complexity and performance perspective, working with mesh manifolds has demonstrated to outperform traditional 3D convolutional neural networks (CNNs) in classification performance, along with significant improvements in computational complexity (thus training time). This GCN experimental study uses an efficient pooling strategy from [**3**] on graphs which, after a rearrangement of the vertices as a binary tree, becomes analogous to 1D pooling.

In the proceeding experiment, a triangular mesh (trimesh)-specific coarsening strategy is used from [**4**] to develop multi-scale hierarchical representations of water-tight trimeshes which capture both global and local context by dropping vertices that minimize quadric error [**5**]. Additionally, a novel convolution operator using spiral sequence kernels on trimeshes is introduced as an improvement on the prior spectral graph convolution approach with [**3**]. Instead of relying on *transductive* GCNs, this approach is based on a message passing formulation along the spiral sequence, thus making it a message passing neural network (MPNN). For both approaches, model performance is improved when compared to traditional Euclidean CNNs on the same task in terms of both classification accuracy and computational efficiency for model training. To provide further novelty to both approaches, a neural network visualization technique known as Grad-CAM, introduced with traditional 2D CNNs, is applied with both approaches to further support the learned classifiers. Specifically, we generate localized interpretable heatmaps on input meshes where the "importance" of regions on surface meshes is highlighted for particular predictions. In other words, having a trainable classifier is good and all, but an *interpretable* classifier provides much more utility in a clinical setting.

Structural measures of the brain captured using MRI can be used as biomarkers to stage the progression of AD progression, even before clinical symptoms manifest. In its early stages and progression, Alzheimer's disease can act as an "invisible" illness in a way. That is part of what makes this experience all the more necessary to detect early on. Science and medicine have had tremendous progress in the fields of combating noncommunicable diseases such as AD, heart disease, and diabetes through multiple advances in machine learning (ML) and artificial intelligence (AI). In a 2015 article [**6**], Insel *et al.* discuss the hidden global costs of noncommunicable diseases, which were noted by a team of scholars from the Harvard School of Public Health and the World Economic Forum to potentially pose a greater risk than contagious illnesses in the future. Their report predicted that the largest source of future costs in global health would be mental health issues; with a specificity of more than a third of the global economic burden of noncommunicable diseases by 2030.

Personality and approach-avoidance behavior are core concepts in research on mental health issues. In recent years, the pathophysiology of mental health problems in adults has been heavily researched, and continues to grow with increasing support. For example, specific forms of abnormal neural processing have been associated with depressive symptomatology. Much of this dysfunction centers on brain circuitry between the cortex and the limbic system. Specifically, patterns in reward and aversion circuits, which process emotional stimuli, are seen as important biological substrates for depression [**7, 8, 9**]. Approach-avoidance behaviors, measured by human interaction with reward stimuli, may be strong indicators in predicting a plethora of pathologies and behavioral biases in the context of reward.

In 2022, during a summer internship at Nike Inc., I had the good fortune of getting to experiment with human interaction data, in the context of sport activity recommender systems, by analyzing athlete* interaction data towards a portfolio of educational/interactive exercise mixed media. We abstracted athlete*-workout engagement data as interaction graphs by generalizing graphs to potentially have varying node/edge types. In the context of this thesis, we have only discussed *homogeneous* graphs so far. By abstracting nodes and edges to vary in *type*, we were able

*If you have a body, you are an athlete.

to apply recently developed GNN tools on *heterogeneous* graphs, towards athlete*-workout-category graphs to recommend new workouts to athletes*, by abstracting prior engagement.

In the final half of this thesis, we apply this same principle of *heterogeneity* in graphs to make predictions about humans based on approach-avoidance reward behavior towards a portfolio of validated emotional stimuli. The human reward behavior half of this work begins with an in-depth description of the approach-avoidance variable measures captured across three distinct cohorts of human participants. From this work, we found that the broad set of features extracted from a simple picture rating task provide a potential framework for summarizing human reward and aversion judgements. This set of summary metrics, derived from a simple rating task on a digital device, could characterize human preference at the big-data scale, potentially across the 83.72% of the world's population that currently owns a smartphone [**10**], or the 85% of Americans with a smartphone (at least 97% own a cellphone of some kind) [**11**]. The final segment of this work is aimed at making predictions about "human" nodes within a *heterogeneous* picture rating graph, with the hope in its continuation to potentially predict mental health issues based on the same reward behavioral variables.

CHAPTER 1

Introduction to Graph Signal Processing and Geometric Deep Learning

Abstract

Graph signal processing (GSP) is presented here from a traditional digital signal processing (DSP) perspective by drawing relevant correspondences that frame GSP techniques as generalizations of fundamental DSP concepts. Mainly, convolution on graph signals is defined to provide a foundation for graph convolutional networks (GCNs), which are used to design the geometric deep learning (GDL) Alzheimer's disease (AD) classification frameworks discussed in the proceeding chapters. In this work, triangular meshes (trimeshes) are used to represent the boundaries (surface) of neuroanatomical regions as mesh manifolds instead of 3D volumes. Triangular meshes (trimeshes) present an efficient alternative for characterizing 3D shape for object boundaries when compared to 3D volumes by using positional features in an object's native 3D space rather than voxel intensities that can differ with varying 3D structural imaging protocols across samples collected from different sites/devices. This chapter concludes with a foundation on message passing neural networks (MPNNs) to further generalize GCNs, which are used in different variations

(e.g., homogeneous, heterogeneous) in the subsequent chapters for improving the AD task and making predictions over complex relational graphs.

1.1. Shifts in Traditional Digital Signal Processing

Discrete signal processing [**12, 13, 14**] studies signals that are linearly-spaced into discrete sequences in d-dimensional Euclidean grids. Standard images are an example of 2D discrete signals, made up of a finite 2D Euclidean grid of pixels, containing a finite set of multiple distinct features that describe colors parameterized by a color space. In the simple case of grayscale images (Figure 1.1), each pixel's grayscale intensity is inferred by a scalar feature, using a grayscale palette inclusively ranging between 0 and 255 (black to white for each pixel). With red-green-blue (RGB) images (sometimes referred to as truecolor images), pixels are defined by their *color*, which is determined by the combination of the red (R), green (G), blue (B) additive primaries stored in each color plane at the pixel's location (total of 3 features per pixel) to produce any color using RGB primaries.

The 2D examples provided by Figure 1.1 and Figure 1.2 demonstrate that an underlying Euclidean-grid structure is an underlying assumption of the format for discretized data in DSP. At the highest level, graph signal processing (GSP) [**15, 16, 17, 18**] extends DSP to signal samples indexed by nodes of a graph with a generalizable topology, not limited to strict, grid-like lattice structures. Therefore like DSP, GSP is the study of: (1) signals and their representations, (2) systems that process signals, typically referred to as filters, (3) signal transforms (e.g. Fourier transform), (4) the sampling of signals, and several other specialized topics. The

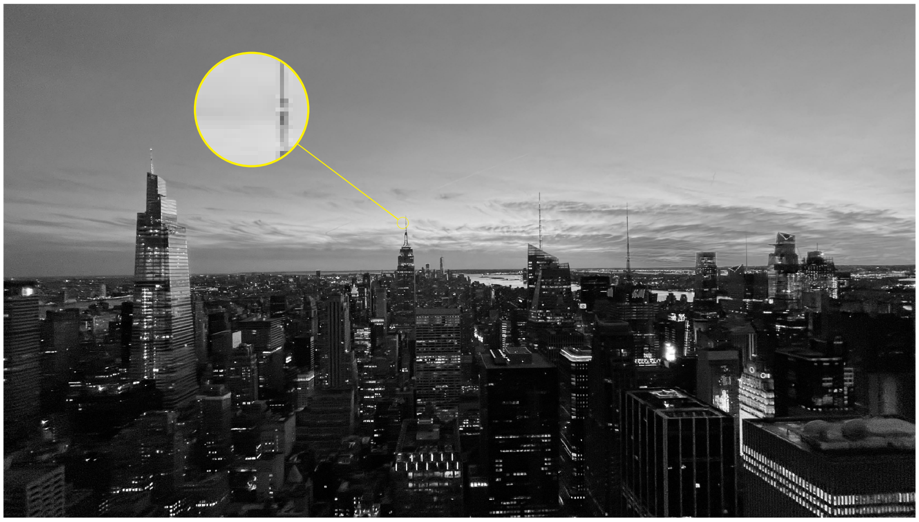


Figure 1.1. Grayscale image of downtown New York City (NYC) skyline from the top floor of the observation deck at the top of the Rockefeller Center taken in January of 2021. A 10x magnified zoom-in view of the antenna tower at the top of the Empire State building demonstrates the tiled lattice structure of 2D pixels, each described by their scalar intensities inferring shades of black on the grayscale color palette on the left.

foundation of the applications described in this work are based on generalized GSP abstractions of traditional DSP concepts and tools.

First, we consider N samples of a finite signal s_n for $n = 0, 1, \ldots, N-1$. For the purposes of this introduction, we restrict ourselves to signals of finite length and finite impulse response (FIR) filters. In DSP, the z-transform $S(z)$ of a 1D time signal, $s = \{s_n : k = 0, 1, \ldots, N-1\}$ organizes its samples into an ordered set, where the sample s_n at time n precedes s_{n+1} at time $n+1$, and succeeds s_{n-1} at time $n-1$. This ordered N-tuple representation is achieved by using a formal variable z^{-1}, referred

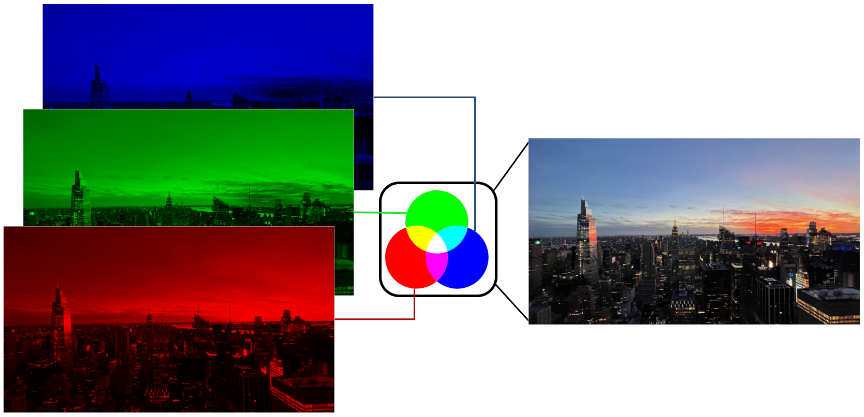


Figure 1.2. Individual RGB color plane primaries describe pixel color for corresponding truecolor version of NYC skyline image, as opposed to grayscale version in Figure 1.1. Each of the RGB planes is a 2D grid of scalar intensities varying within the inclusive range of $[0, 255]$.

to as a *shift* (or delay), so that the samples are formally represented as

$$S(z) = \sum_{n=0}^{N-1} s_n z^{-n}. \tag{1.1}$$

The z-transform is a useful tool that provides a formal polynomial representation (complex or real-valued) of a discrete signal (complex or real-valued) that is a useful representation for studying how signals are processed by filters.

The discrete Fourier transform (DFT) of an arbitrary signal s is represented using the Fourier coefficients of the signal $\hat{s} = \{\hat{s}_k : k = 0, 1, \ldots, N-1\}$ given by

$$\hat{s}_k = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} s_n e^{-j\frac{2\pi}{N}kn}. \tag{1.2}$$

The DFT provides a spectral representation of the signal s composed of the discrete frequencies $\Omega_k = \frac{2\pi k}{N}$, for $k = 0, 1, \ldots, N-1$, and the k spectral components

$$\left\{ x_k[n] = \frac{1}{\sqrt{N}} e^{-j\frac{2\pi}{N}kn} : n = 0, 1, \ldots, N-1 \right\}. \tag{1.3}$$

Signals can be recovered using their Fourier coefficients by way of the inverse discrete Fourier transform (IDFT), defined by

$$s_n = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} \hat{s}_k e^{j\frac{2\pi}{N}kn}, n = 0, 1, \ldots, N-1. \tag{1.4}$$

Aside from signals, systems that process signals (filters) are also studied in DSP. FIR filters can be represented by using the z-transform of their impulse response h_n such that

$$H(z) = \sum_{n=0}^{N-1} h_n z^{-n}. \tag{1.5}$$

The z-transform of an output, $Y(z)$, using the FIR filter, h, applied on s can be determined via multiplication in the z-domain.

$$Y(z) = H(z)S(z). \tag{1.6}$$

As addressed by Ortega *et al.* [**16**], given that (1) the signals and impulse response of the systems of interest are *finite* in length, and (2) that the product in Equation 1.6 above could result in $Y(z)$ being a polynomial in z^{-1} greater than $N-1$, boundary conditions must be considered. For simplicity, periodic extensions of the 1D signal s

are considered such that $s_n = s_{n \bmod N}$, i.e. the signal s_{n+N} is equal to s_n. With this additional extension, the 1D *shift* (or delay) filter

$$H_{\text{delay}}(z) = z^{-1}, \tag{1.7}$$

applied on the signal $x = \{x_n : n = 0, 1, \ldots, N-2, N-1\}$, would generate the output

$$y = H_{\text{delay}} \cdot x = \{y_n : n = N-1, 0, 1, \ldots, N-3, N-2\}\,.$$

By factoring out the summand from the exponent in Equation 1.5 we can define any filter h in DSP as a polynomial in z^{-1} (i.e. any series and/or parallel combinations of shifts defined in Equation 1.7 such that

$$H(z) = \sum_{n=0}^{N-1} h_n \left(H_{\text{delay}}\left(z\right)\right)^n = \sum_{n=0}^{N-1} h_n z^{-n}.$$

1.2. Shifts in Graph Signal Processing

To extend the concepts of signals and systems in DSP to *graphs*, i.e. samples whose nodes are indexed by nodes of an arbitrary network, we first begin by reinterpreting the finite signals from the previous section as vectors rather than ordered tuples or sequences; thus allowing us to rewrite an arbitrary *graph* signal $s = \{s_n : n = 0, 1, , \ldots, N-1\}$, real-valued or complex, as

$$\mathbf{s} = [s_0, s_1, \ldots, s_{N-1}]^{\intercal} \in \mathbb{C}^N. \tag{1.8}$$

By analogy, a filter h can be rewritten as a matrix $\mathbf{H}$, and Equation 1.6 can be rewritten as

$$\mathbf{s}_{\text{out}} = \mathbf{H} \cdot \mathbf{s}_{\text{in}}, \tag{1.9}$$

where filters are represented by 2D matrices and signals are represented as 1D vectors. The periodically-extended signal shift/delay described in Equation 1.7 of the previous section can be rewritten as the circulant matrix $\mathbf{A}_{\text{delay}}$ such that

$$\left[s_{N-1}, s_0, s_1, \ldots, s_{N-3}, s_{N-2}\right]^{\intercal} = \mathbf{A}_{\text{delay}} \cdot \left[s_0, s_1, \ldots, s_{N-2}, s_{N-1}\right]^{\intercal},$$

where

$$\mathbf{A}_{\text{delay}} = \begin{bmatrix} 0 & 0 & 0 & \ldots & 0 & 1 \\ 1 & 0 & 0 & \ldots & 0 & 0 \\ 0 & 1 & 0 & \ldots & 0 & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & \ldots & 1 & 0 & 0 \\ 0 & 0 & \ldots & 0 & 1 & 0 \end{bmatrix}. \tag{1.10}$$

By viewing the cyclic shift matrix $\mathbf{A}_{\text{delay}}$ as the adjacency matrix of a graph, we can then assume a cycle graph, $\mathcal{G}_c$, topology of the signal $\mathbf{s}$, as demonstrated by Figure 1.3. By labeling the rows and columns of $\mathbf{A}_{\text{delay}}$ from 0 to $N-1$, we can define the graph $\mathcal{G}_c = (\mathcal{V}_c, \mathcal{E}_c)$, with the set of vertices $\mathcal{V}_c = \{v_n : n = 0, 1, \ldots, N-1\}$, and the set of edges $\mathcal{E}_c$. Next, we set row n of $\mathbf{A}_{\text{delay}}$ to represent the set of inward-edges

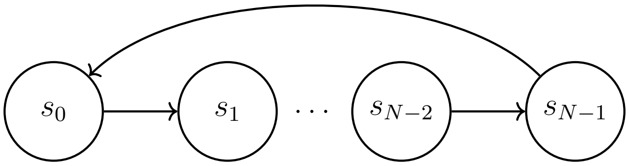


Figure 1.3. Periodically-extended discrete time sequence interpreted as a directed cycle graph, $\mathcal{G}_c$

of vertex n in $\mathcal{G}_c$ if there is a 1 at the entry in column m, $(\mathbf{A}_c)_{n,m} = 1$, then there is an edge connecting vertex m to n. In their review paper [**16**], Ortega *et al.* highlight the duality of $\mathbf{A}_{\text{delay}}$ in Equation 1.10, which represents the shift/delay z^{-1} in DSP, as well as the adjacency matrix of the directed cycle graph in Figure 1.3.

A *graph* interpretation of linear shift-invariant (LSI) systems in DSP can be extended to GSP [**15**]. Reconsidering the graph signal $\mathbf{s} \in \mathbb{C}^N$, where the signal's samples are indexed by the N nodes of the graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, for $\mathcal{V} = \{v_n : n = 1, 2, \ldots, N\}$, we denote the edge weights of the graph as w_{ij}, denoting an edge e_{ij} connecting vertex v_i to v_j, weighted by the scalar value w_{ij}, to define the following algebraic matrix representations associated to the graph $\mathcal{G}$:

Definition 1.2.1 (Graph adjacency matrix)**.** A graph's **adjacency matrix** is a square matrix, $\mathbf{A} \in \mathbb{R}^{N \times N}$, where

$$(\mathbf{A})_{ij} = \begin{cases} w_{ij}, & e_{ij} \in \mathcal{E} \\ 0, & e_{ij}\text{otherwise} \end{cases} \in \mathbb{R}^{N \times N}, \tag{1.11}$$

indicating whether a vertex v_i is adjacent to v_j, and weighted by w_{ij} if so. In the particular case where the graph is undirected, $w_{ij} = w_{ji}$, therefore making $\mathbf{A}$ symmetric.

Definition 1.2.2 (Graph degree matrix)**.** In the case of an undirected graph, the degree matrix, $\mathbf{D}$, is defined as the diagonal matrix

$$(\mathbf{D})_{ij} = \begin{cases} \sum_{j=1}^{N} (\mathbf{A})_{ij}, & i = j \\ 0, & i \neq j \end{cases} \in \mathbb{R}^{N \times N}, \tag{1.12}$$

denoting the weighted sum of edge weights connected to each vertex $v_i \in \mathcal{V}$, referred to as the degree of the corresponding vertex.

Definition 1.2.3 (Graph Laplacian matrix)**.** The **combinatorial graph Laplacian L**, of an undirected graph is defined using the graph's corresponding adjacency matrix and degree matrix such that

$$\mathbf{L} = \mathbf{D} - \mathbf{A}. \tag{1.13}$$

The graph Laplacian is a useful matrix representation of an undirected graph containing information about edge weights for adjacent vertices and the degree of each vertex. The **symmetric normalized graph Laplacian**, $\mathbf{L}^{\text{sym}}$, is defined as

$$\mathbf{L}^{\text{sym}} = \mathbf{D}^{-1/2}\mathbf{L}\mathbf{D}^{-1/2} = \mathbf{I}_N - \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}. \tag{1.14}$$

In this context, the adjacency matrix $\mathbf{A}$ can be adopted as the *graph shift* operator [**16, 15**] for any given graph. Other choices for a shift operator have been proposed, including the Laplacians of undirected graphs [**17**], and other variations of these matrices [**19, 20**]; each choice for a shift presenting different trade-offs. The adjacency matrix $\mathbf{A}$ reduces to the shift in traditional DSP and applies to directed and undirected graphs, while graph Laplacian spectra only apply to undirected graphs, so that $\mathbf{L}$ remains symmetric, positive semi-definite, and avoids numerous analytical difficulties that may arise by retaining other useful properties.

With 1D discrete signals, the basis $\{z^{-n}\}_{n=0}^{N-1}$ orders samples in the sequence by increasing order of the time index n (vertices in the cycle graph). Therefore, the z-transform of a signal $\mathbf{s}$ can be rewritten as

$$S(z) = \left[\left(z^{-1}\right)^{0}, z^{-1}, \ldots, z^{-(N-1)}\right] \left[s_0, s_1, \ldots, s_{N-1}\right]^{\intercal}.$$

With GSP, the ordering of samples is determined by the labeling of the vertices of the graph. This labeling or numbering fixes the adjacency matrix $\mathbf{A}$, therefore fixing the corresponding graph shift operator for that graph. The columns of the graph shift operator provide a basis for a representation of the graph signals.

Analogous to traditional DSP, the notion of linearity and shift-invariance for filters are also defined for arbitrary graphs. A filter represented by the matrix $\mathbf{H}$ is labeled as shift-invariant if it commutes with a graph shift such that

$$\mathbf{AH} = \mathbf{HA}. \tag{1.15}$$

Filters are also defined as **linear** systems if the filter's output for a linear combination of input signals equals the linear combination of outputs to each signal:

$$\mathbf{H}\left(\alpha\mathbf{s}_1+\beta\mathbf{s}_2\right)=\alpha\mathbf{H}\mathbf{s}_1+\beta\mathbf{H}\mathbf{s}_2. \tag{1.16}$$

Section A.1 of the Appendix, provides a theorem that shows all LSI graph filters are given by *polynomials* in a graph shift (i.e. $\mathbf{A}$)

$$\mathbf{H}=h\left(\mathbf{A}\right), \tag{1.17}$$

where $h(\cdot)$ is a polynomial function

$$h(x)=h_0+h_1x+h_2x^2+\cdots+h_Lx^L.$$

Analogous to traditional DSP, the coefficients h_i of the polynomial $h(\cdot)$ can be interpreted as graph filter *taps.* In the following section we analyze graph signals and their processing with LSI graph filters using frequency representations for graph signals and convolutions on graphs by way of spectral filtering.

1.3. Frequency Analysis and Spectral Filtering of Graph Signals

In DSP and other analyses involving linear systems, it is interesting to analyze signals that are invariant when processed by a linear filters, i.e.

$$h\cdot s_{\text{in}}=\alpha s_{\text{in}},$$

where α is a scalar. Such a signal s_{in} is referred to as an eigensignal of the filter h. With GSP, filters are defined as matrices, therefore the eigensignals of h are the eigenvectors of the corresponding graph filter $\mathbf{H}$. Interestingly enough, given that LSI filters are polynomials in the graph shift $\mathbf{A}$, only the eigenvectors and eigenvalues of $\mathbf{A}$ need be considered. Therefore we define the eigendecomposition of $\mathbf{A}$ as

$$\mathbf{A} = \mathbf{V}\boldsymbol{\Lambda}\mathbf{V}^{-1}, \tag{1.18}$$

where $\mathbf{V} = [\mathbf{v}_0, \mathbf{v}_1, \ldots, \mathbf{v}_{N-1}]$ is the matrix of N eigenvectors for $\mathbf{A}$, and $\boldsymbol{\Lambda} = \operatorname{diag}\left([\lambda_0, \lambda_1, \ldots, \lambda_{N-1}]\right)$ is the diagonal matrix of the distinct corresponding eigenvalues. Assuming that $\mathbf{A}$ has a complete set of eigenvectors, $\mathbf{V}$ becomes invertible. Therefore, since a graph filter $\mathbf{H}$ is defined by a polynomial in the graph shift based on Equation 1.17, and knowing the eigendecomposition in Equation 1.18, it is simple to verify that

$$\begin{aligned} \mathbf{H} &= h\left(\mathbf{A}\right) \\ &= h\left(\mathbf{V}\boldsymbol{\Lambda}\mathbf{V}^{-1}\right) \\ &= \sum_{m=0}^{M-1} h_m \left(\mathbf{V}\boldsymbol{\Lambda}\mathbf{V}^{-1}\right)^m \\ &= \mathbf{V}h\left(\boldsymbol{\Lambda}\right)\mathbf{V}^{-1}. \end{aligned} \tag{1.19}$$

Using this, we can introduce the graph Fourier transform (GFT) for signals on graphs. First, we begin by analyzing the eigendecomposition of the directed cycle graph's (time delay) adjacency as

$$\mathbf{A}_{\text{delay}} = \mathbf{DFT}_N^{-1} \begin{pmatrix} e^{-j\frac{2\pi\cdot 0}{N}} & & \\ & \ddots & \\ & & e^{-j\frac{2\pi\cdot(N-1)}{N}} \end{pmatrix} \mathbf{DFT}_N, \tag{1.20}$$

where $\mathbf{DFT}_N = \frac{1}{\sqrt{N}}\left[\omega_N^{kn}\right], \omega_N = \exp -j\frac{2\pi}{N}$, is the discrete Fourier transform (DFT) matrix. By correspondence, the inverse $\mathbf{DFT}_N^{-1} = \mathbf{DFT}_N^H$ matrix is the set of eigenvectors for $\mathbf{A}_{\text{delay}}$, and the corresponding eigenvalues are $e^{-j\frac{2\pi n}{N}}, n = 0, 1, \ldots, N-1$, the diagonal of the middle matrix in Equation 1.20. By analogy, the GFT follows with Equation 1.20.

Definition 1.3.1 (Graph Fourier transform (analysis decomposition))**.** Using the set of eigenvectors $\mathbf{V}$ from the eigendecomposition of $\mathbf{A}$ in Equation 1.18, the GFT of a graph signal $\mathbf{s}$ is defined as

$$\hat{\mathbf{s}} = \mathbf{F}\mathbf{s}, \tag{1.21}$$

for the GFT matrix

$$\mathbf{F} = \mathbf{V}^{-1}. \tag{1.22}$$

By construction, the eigenvectors $\mathbf{V}$ of the graph shift $\mathbf{A}$, are the graph spectral components, and the eigenvalues $\mathbf{\Lambda}$, is the diagonal matrix of λ_k entries that are the graph frequencies.

Definition 1.3.2 (Inverse graph Fourier transform (synthesis))**.** By design, the inverse graph Fourier transform (IGFT) is defined as

$$\mathbf{s} = \mathbf{F}^{-1}\hat{\mathbf{s}} = \mathbf{V}\hat{\mathbf{s}}, \tag{1.23}$$

for the IGFT matrix

$$\mathbf{F}^{-1} = \mathbf{V}. \tag{1.24}$$

The IGFT synthesizes the original signal $\mathbf{s}$ from its spectral components $\mathbf{V}$.

Using these definitions, we can now interpret how to process a graph signal with a filter $\mathbf{H}$.

Definition 1.3.3 (Spectral Graph Convolution (Spectral Filtering))**.** Following Equation 1.19 we can compute the output of $\mathbf{s}_{\text{in}}$ to a filter h

$$\begin{aligned}
\mathbf{s}_{\text{out}} &= \mathbf{H} \cdot \mathbf{s}_{\text{in}} \\
&= \mathbf{V} h\left(\mathbf{\Lambda}\right) \underbrace{\left(\mathbf{V}^{-1}\mathbf{s}_{\text{in}}\right)}_{\text{GFT}} \\
&= \mathbf{V} \underbrace{\operatorname{diag}\left[h\left(\lambda_0\right), \ldots, h\left(\lambda_{N-1}\right)\right] \hat{\mathbf{s}}_{\text{in}}}_{\text{Spectral filtering in graph Fourier space}} \\
&= \underbrace{\mathbf{V}\left[h\left(\lambda_0\right)\hat{\mathbf{s}}_{\text{in}_0}, \ldots, h\left(\lambda_{N-1}\right)\hat{\mathbf{s}}_{\text{in}_{N-1}}\right]^{\intercal}}_{\text{IGFT}}.
\end{aligned} \tag{1.25}$$

Therefore, graph convolution is implicitly defined by way of spectral filtering by taking the GFT of a signal ($\mathbf{V}^{-1}\mathbf{s}_{\text{in}}$), followed by an element-wise multiplication in the graph frequency space of the GFT signal $\hat{\mathbf{s}}_{\text{in}}$ by the frequency response of the filter $\left[h\left(\lambda_0\right),\ldots,h\left(\lambda_{N-1}\right)\right]^{\intercal}$.

This implicit definition of graph convolution by way of spectral filtering plays into the duality property of the traditional Fourier transform that applies to the GFT as well; where convolution in a certain space equates to multiplication in the corresponding Fourier space and vice versa. As in traditional DSP, the concept of low-, high-, and band-pass signals and filters are also considered in GSP using the notion of graph frequency from the eigenvalues of the spectral filter $h\left(\mathbf{\Lambda}\right)$ for the graph frequencies λ_k in $\mathbf{\Lambda} = \text{diag}\left[\lambda_0,\ldots,\lambda_{N-1}\right]$.

In the 1D DSP example, the concept of low-, high-, and band-pass signals/filters relate directly to the frequencies defined by the eigenvalues of the graph shift $\mathbf{A}_{\text{delay}}$ from Equation 1.10 and Equation 1.20

$$\Omega_k = \frac{2\pi k}{N}, \quad k = 0, 1, \ldots, N-1,$$

corresponding to the complex-valued spectral components defined by the eigenvectors in the 1D DFT matrix. This notion of frequency in 1D DSP is related to the degree of variation of the signal's spectral components in $\mathbf{V}$. The least varying spectral component in $\mathbf{V}$ corresponds to the frequency (eigenvalue) $\Omega_0 = 0$. The next frequency $\Omega_1 = \frac{2\pi}{N}$ represents the next higher variation spectral component, and so on. A correspondence between the ordered set of frequencies and the corresponding degrees of variation (or complexity) of the time spectral components exists, as highlighted by Ortega *et al.* [**16**].

Frequencies are defined by the eigenvalues of the graph shift $\mathbf{A}$ for an arbitrary graph $\mathcal{G}$ in GSP. Analogous to traditional DSP, graph frequencies λ_k can be ordered in relation to the complexity of the corresponding graph spectral components in $\mathbf{V}$. One method of measuring the complexity in each spectral component to determine an ordering, is to measure the total variation (TV) of the component through

$$\mathrm{TV}\left(\mathbf{v}_k\right) = \left\|\mathbf{v}_k - \mathbf{A}^{\mathrm{norm}}\mathbf{v}_k\right\|_1,$$

where $\|\cdot\|$ is the ℓ_1 norm, and $\mathbf{A}^{\mathrm{norm}} = \frac{1}{\lambda_{\max}}\mathbf{A}$. With this measure of complexity for each graph frequency's corresponding spectral component, the graph frequency λ_m is considered larger than the graph frequency λ_n if

$$\mathrm{TV}\left(\mathbf{v}_m\right) > \mathrm{TV}\left(\mathbf{v}_n\right).$$

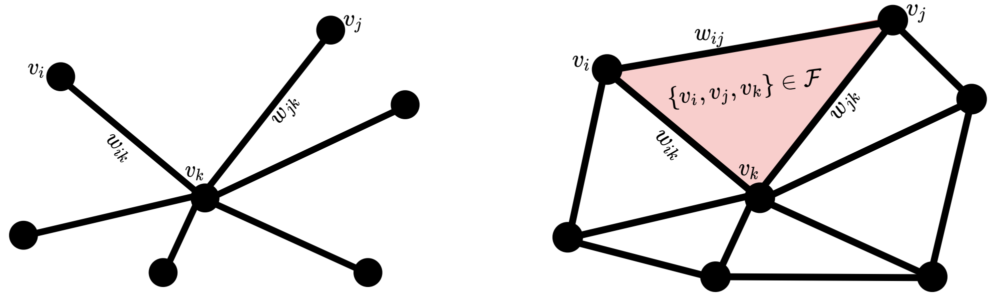
Using this type of ordering for graph frequencies, synonymous low-, high-, and band-pass signals and filters can be defined in GSP.

1.4. Interpretation of Mesh Manifolds as Graphs and Mesh Notation

State-of-the-art (SOTA) volumetric approaches to problems in 3D imaging such as 3D object classification and segmentation rely on operating with 3D image volume inputs [**21, 22, 23**]. In several medical, particularly clinical, practices, mesh representations are often required to study shape correspondences in biological systems such as organ morphology and computing area-based statistics. Triangular meshes (trimeshes) are often used to model the boundaries/surface of objects in 3D space, while tetrahedral meshes are often used to describe 3D structure. Approaches that rely on operating with 3D volumes to eventually produce surface meshes (e.g., 3D graphics), often rely on algorithms such as Marching Cubes [**24**] and some type of mesh smoothing technique [**25, 26**]; which are not differentiable, prevent an end-to-end optimization pipeline, and leave mesh outputs subject to artifacts that may be introduced during any step of the pipeline. Artifacts may also be introduced to output meshes with respect to the quality of the processed 3D volumes. In our analyses, we only focus on and refer to trimeshes describing 3D surfaces, and interchangeably refer to 3D surface trimeshes as just "meshes."

In computer graphics and vision applications, including medical image analysis, 2D manifolds can often be used to describe and model 3D shapes, particularly their surface/boundaries. To do this, first a discretization of the manifold consisting of N points (vertices) is assumed. Each point, v_i, is represented by its corresponding 3D coordinates as the ordered vector $\mathbf{p}_i = [x_i, y_i, z_i]$. The set of 3D coordinates for all N points $\{\mathbf{p}_i : i = 1, 2, \ldots, N\}$ of the mesh manifold is often referred to as a *point cloud*. Second, a graph is constructed upon these points, acting as vertices, meanwhile the edges of the graph are constructed using information about the local/global connectivity of vertices of the surface manifold. One example of determining edge weights w_{ij} between vertex v_i and v_j is to use the Gaussian-inspired distance

$$w_{ij} = \exp\left\{\frac{-\left\|\mathbf{p}_i - \mathbf{p}_j\right\|_2^2}{2\sigma^2}\right\},$$



(a) Simple arbitrary undirected graph (b) Triangular mesh (undirected graph)

Figure 1.4. Sample discretizations of 2D mesh manifolds.

where $\|\cdot\|_2$ represents the ℓ_2-norm and σ is arbitrarily selected. Two sample discretizations of 2D manifolds are shown in Figure 1.4, using the same edge weight function w_{ij}, with differing undirected graph structures.

Simple discretizations of 2D manifolds such as Figure 1.4a do not correctly capture the geometry of the underlying continuous manifold (no free lunch [**27**]). Trimeshes offer a geometrically consistent discretization that is possible with the additional structure of faces $\mathcal{F}$. A mesh $\mathcal{M}$ can be interpreted as an undirected graph described by $\mathcal{M}(\mathcal{V}, \mathcal{E}, \mathcal{F})$, the set of vertices $\mathcal{V}$, a corresponding set of edges $\mathcal{E}$ (each with an edge weight w_{ij}), and a corresponding set of faces, $\mathcal{F}$. Faces are represented using three-element tuples $\{v_i, v_j, v_k\} \in \mathcal{F}$, where the distinct elements of each tuple corresponds to the three vertices that make up each face using the corresponding set of edges $\{e_{ij}, e_{jk}, e_{ik}\} \in \mathcal{E}$. A trimesh is depicted in Figure 1.4b, where the face highlighted consists of a corresponding set of vertices $\{v_i, v_2, v_3\}$ and their corresponding edges. To encapsulate all of the shared features on the vertices of a single mesh, we use the vertex feature matrix, $\mathbf{X} \in \mathbb{R}^{N \times F}$. Like graph signals, the GFT, can be applied to vertex feature matrices, $\mathbf{X}$, such that

$$\hat{\mathbf{X}} = \mathbf{F}\mathbf{X},$$

in addition to the IGFT

$$\mathbf{X} = \mathbf{F}^{-1}\hat{\mathbf{X}}.$$

For reference, Table 1.1 summarizes all of the mesh notation necessary for the remainder of the text.

1.5. Graph Convolutional Networks and Geometric Deep Learning

Artificial neural networks (NNs) have had a massive impact and success over the past decade in many fields due to several technological advances in parallel computing and wider availability of larger-scale datasets; for which deep learning (DL) approaches like NNs tend to scale well with. However, early variants of NNs were mostly intended and implemented for data that exists on regular Euclidean grids. In many facets of real-world data, there is an underlying graph structure that is not Euclidean (i.e. molecular graphs, social networks, citation graphs, brain connectomes, GPS routes, etc.) as illustrated by the example in Figure 1.5. In recent years, several approaches have re-visited the problem of generalizing well-established NN models to work on arbitrarily structured graph data [**28, 29, 30, 31, 3, 32**].

A convolutional neural network (CNN) [**33**] is an efficient, parameterized DL architecture that is used to extract highly meaningful statistical patterns in large-scale and high-dimensional datasets. What makes CNNs so powerful in areas such

Table 1.1. Mesh signal processing notation and definitions.

Notation		**Definition**
$\mathcal{V}$	$\triangleq$	set of vertices
$\mathcal{E}$	$\triangleq$	set of edges
$\mathcal{F}$	$\triangleq$	set of faces
$\mathcal{M}(\mathcal{V}, \mathcal{E}, \mathcal{F})$	$\triangleq$	mesh defined by set of vertices $\mathcal{V}$, edges $\mathcal{E}$, and faces $\mathcal{F}$
$v_i \in \mathcal{V}$	$\triangleq$	arbitrary vertex
N	$\triangleq$	number of vertices
$\mathbf{x} \in \mathbb{R}^{F_{in}}$	$\triangleq$	vertex feature vector
$\mathbf{X} \in \mathbb{R}^{N \times F_{\mathrm{in}}}$	$\triangleq$	vertex feature matrix
$e_{ij} = e_{ji}$	$\triangleq$	arbitrary edge connecting vertex v_i to v_j
$w_{ij} = w_{ji}$	$\triangleq$	corresponding scalar edge weight for edge e_{ij}
$\mathcal{F}_{ijk} \in \mathcal{F}$	$\triangleq$	$\mathcal{F}_{ijk} = \{v_i, v_j, v_k\}$, arbitrary triangle face defined by its vertices
$\mathbf{A} \in \mathbb{R}^{N \times N}$	$\triangleq$	$(\mathbf{A})_{ij} = \begin{cases} w_{ij},\ e_{ij} \in \mathcal{E} \\ 0, \text{ otherwise} \end{cases}$, graph adjacency matrix
$\mathbf{D} \in \mathbb{R}^{N \times N}$	$\triangleq$	$(\mathbf{D})_{ii} = \sum_{N}^{j=1} (\mathbf{A})_{ij}$, vertex degree matrix
$\mathbf{L} \in \mathbb{R}^{N \times N}$	$\triangleq$	$\mathbf{L} = \mathbf{D} - \mathbf{A}$, combinatorial graph Laplacian matrix
$\mathbf{L}^{\mathrm{sym}} \in \mathbb{R}^{N \times N}$	$\triangleq$	$\mathbf{L}^{\mathrm{sym}} = \mathbf{D}^{-1/2}\mathbf{L}\mathbf{D}^{-1/2} = \mathbf{I}_N - \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$, symmetric normalized Laplacian
$\mathbf{A} = \mathbf{V}\boldsymbol{\Lambda}\mathbf{V}^{-1}$	$\triangleq$	eigendecomposition of arbitrary graph shift $\mathbf{A}$
$\mathbf{V} \in \mathbb{R}^{N \times N}$	$\triangleq$	complete set of eigenvectors (therefore invertible)
$\boldsymbol{\Lambda} \in \mathbb{R}^{N \times N}$	$\triangleq$	$\mathrm{diag}[\lambda_1, \ldots, \lambda_N]$, corresponding diagonal matrix of eigenvalues
$\mathbf{F} \in \mathbb{R}^{N \times N}$	$\triangleq$	$\mathbf{F} = \mathbf{V}^{-1}$, GFT matrix
$\mathbf{F}^{-1} \in \mathbb{R}^{N \times N}$	$\triangleq$	$\mathbf{F}^{-1} = \mathbf{V}$, IGFT matrix
$\hat{\mathbf{x}} \in \mathbb{R}^{N}$	$\triangleq$	$\hat{\mathbf{x}} = \mathbf{F}\mathbf{x}$, GFT of a graph signal $\mathbf{x}$
$\mathbf{H} \in \mathbb{R}^{N \times N}$	$\triangleq$	$\mathbf{H} = h(\mathbf{A})$, graph filter defined by arbitrary polynomial in $\mathbf{A}$

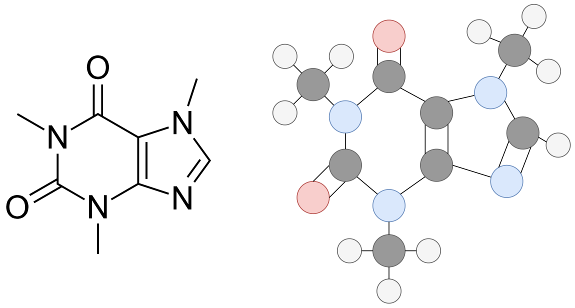
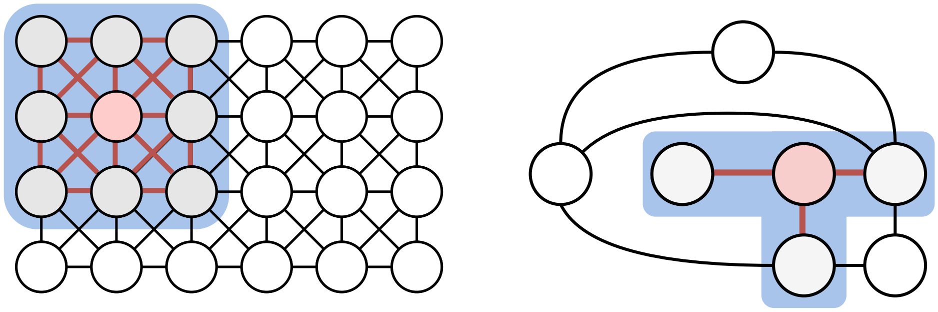


Figure 1.5. Molecular graph structure for caffeine.

as audio, image, and video processing is their ability to learn local stationary structures and compose them to form multi-scale hierarchical patterns by assuming an underlying Euclidean structure to data [**34**]. Precisely, CNNs learn to extract the local stationarity property of input signals by revealing local features that are shared across the data domain using localized convolutional filters or kernels, which are learned from the data. Translational equivariance is an extremely powerful inductive bias that makes CNNs a powerful tool for analyzing Euclidean signals.

1.5.1. Graph convolutional networks (GCNs)

Understanding graph convolutional networks (GCNs) can be simplified by drawing correspondences from traditional DSP to concepts in GSP and interpreting traditional Euclidean signals as graphs in the form of structured grids, particularly rectangular grids in the case of 2D images, as illustrated in Figure 1.6. With traditional 2D convolution, the translation of a pre-defined kernel $\mathbf{H}$ can be interpreted as a pixel *aggregation* scheme that computes the weighted sum of a neighborhood of pixels.



(a) 2D convolution on rectangular grid

(b) Graph convolution using $\mathbf{A}$.

Figure 1.6. GSP interpretation of traditional 2D convolution as weighted vertex/node aggregation on rectangular grid with respect to adjacent vertices

GCNs extend this further by not *assuming* an underlying rectangular grid topology. Instead, GCNs perform a linear mapping of $F_{\text{in}} \mapsto F_{\text{out}}$ features, for each individual vertex, by using a set of learnable parameters $\mathbf{W} \in \mathbb{R}^{\cdots \times F_{\text{in}} \times F_{\text{out}}}$, and a GSP filter $\mathbf{H}$ to describe the aggregation scheme for adjacent vertices (since GSP filters can be defined by polynomials in the graph shift, $\mathbf{H} = h\left(\mathbf{A}\right)$). For example, the output of a simple GCN that only uses the graph's adjacency matrix $\mathbf{A} \in \mathbb{R}^{N \times N}$ can be obtained such that

$$\mathbf{Y} = \mathbf{AXW} \in \mathbb{R}^{N \times F_{\text{out}}}, \tag{1.26}$$

for an input vertex feature matrix $\mathbf{X} \in \mathbb{R}^{N \times F_{\text{in}}}$, and the set of learnable parameters $\mathbf{W} \in \mathbb{R}^{F_{\text{in}} \times F_{\text{out}}}$. Notice that in this simple form: signal padding isn't necessary since the number of vertices N is preserved by the use of $\mathbf{A}$, a finite constraint on the topology of the graph space. Furthermore, the topology of the graph, defined by $\mathbf{A}$, does not change (even if it is used as the graph shift). What *does* end up changing

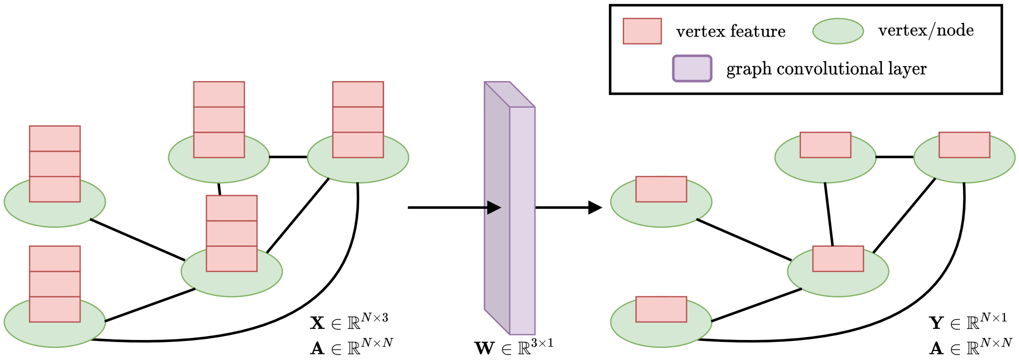

vertex feature
vertex/node
graph convolutional layer
$\mathbf{X} \in \mathbb{R}^{N\times 3}$
$\mathbf{A} \in \mathbb{R}^{N\times N}$
$\mathbf{W} \in \mathbb{R}^{3\times 1}$
$\mathbf{Y} \in \mathbb{R}^{N\times 1}$
$\mathbf{A} \in \mathbb{R}^{N\times N}$

Figure 1.7. Linear mapping of $F_{\text{in}} \mapsto F_{\text{out}}$ graph features per vertex on the same graph described by $\mathbf{A}$, for $F_{\text{in}} = 3$ and $F_{out} = 1$ in this example. The topology of the graph, therefore $\mathbf{A}$, remains the same after graph convolution is applied.

is the corresponding $F_{\text{in}} \mapsto F_{\text{out}}$ features for each vertex, as illustrated in Figure 1.7.

Up to this point, the only type of GNNs discussed have been strictly *spectral*. Graphs do not have a concrete concept or a mathematical definition of spatial translation. This leads to questions on the mathematical foundations that spectral graph filters are based on. With GCNs, using the GFT and its inverse helps us approximate convolutions on graphs (hence their namesake) using matrix multiplication operators that rely on the graph adjacency, $\mathbf{A}$ or Laplacian, $\mathbf{L}$. This can come with a number of caveats, particularly performance issues with very *large* graphs that have extremely large and sparse adjacency matrices (think about the millions of people in a social network and how many of them you would be "related" to). Furthermore, computing the eigenvectors/eigenvalues of large graphs can be computationally expensive, and its complexity grows with the size of the graph, limiting its scalability. Another shortfall is that such graph filters are not *localized* in general. This led researchers in 2017, the year I started my PhD and this body work, to introduce localized solutions like ChebNets [**3**], which are utilized for the classification of Alzheimer's disease (AD) in chapter 3.

1.5.2. Message passing neural networks (MPNNs)

As mentioned before, graphs do not have a concrete concept or a mathematical definition of spatial translation. Representing graphs can be challenging because they can be irregular, i.e., graphs can have a variable size of nodes, and each node in a graph has a different number of neighbors, rendering some operations such as convolutions not compatible with the graph structure. Spectral approaches to convolution on graphs assume a *fixed* graph topology (e.g., re-use of adjacency matrices which can dampen generalizability), thus "breaking" if the original graph structure is changed in some cases (i.e., inserting a single node changes $\mathbf{A}$).

As opposed to spectral graph convolution with GCNs, spatial-based approaches formulate graph convolution as aggregating feature information from neighbouring nodes on a graph. Together with different node sampling strategies, spatial graph convolution computations can be performed in a batch of nodes instead of the whole graph, which has the potential to improve scalability and efficiency of GNNs over very large graphs.

Drawing inspiration from ideas in neural message passing and believe propagation, message passing neural networks are a generalization of spatial convolution on graphs where the key is to learn a function to that generates a node's representation by aggregating its own features, its neighbours' features, and/or potential edge features relative to neighboring node pairs.

Using the convention defined in Table 1.1, with $\mathbf{x}_i^{(k-1)} \in \mathbb{R}^{F_{in}}$ denoting the feature vector of F_{in} features at vertex i and $\mathbf{e}_{i,j} \in \mathbb{R}^{E_{in}}$ denoting the (optional) E_{in} features on edge $e_{i,j}$ connecting vertex i to vertex j at layer $(k-1)$, the output of a MPNN layer is typically defined as

$$\mathbf{x}_i^{(k)} = \gamma^{(k)}\left(\mathbf{x}_i^{(k-1)}, \square_{j\in\mathcal{N}(i)}\phi^{(k)}\left(\mathbf{x}_i^{(k-1)}, \mathbf{x}_j^{(k-1)}, \mathbf{e}_{i,j}^{(k-1)}\right)\right), \tag{1.27}$$

where $\square$ represents a differentiable permutation-invariant operation (i.e. sum, mean or max), and $\gamma^{(k)}$ and $\phi^{(k)}$ denote differentiable kernel functions such as multilayer perceptrons (MLPs). CNNs defined for data types that exist in standard Euclidean grids have a clear one-to-one mapping. However for data types in irregular domains such as graphs, correspondences are defined using neighborhood connectivity for each vertex and weight matrices dependent on the kernel functions, γ and ϕ at each layer. Figure 1.8 provides a simple example of computation flowchart depiction of message passing in MPNNs for obtaining the feature map of a single vertex. This type of generalization is incredibly powerful because it allows us to experiment with and manipulate the learnable functions, $\gamma^{(k)}(\cdot)$ and $\phi^{(k)}(\cdot)$, with NNs or any differentiable function where back-propagation is possible.

Unlike traditional convolution in DSP, where weighted sums are taken over multiple sliding windows of overlap between the input signal and the impulse response of

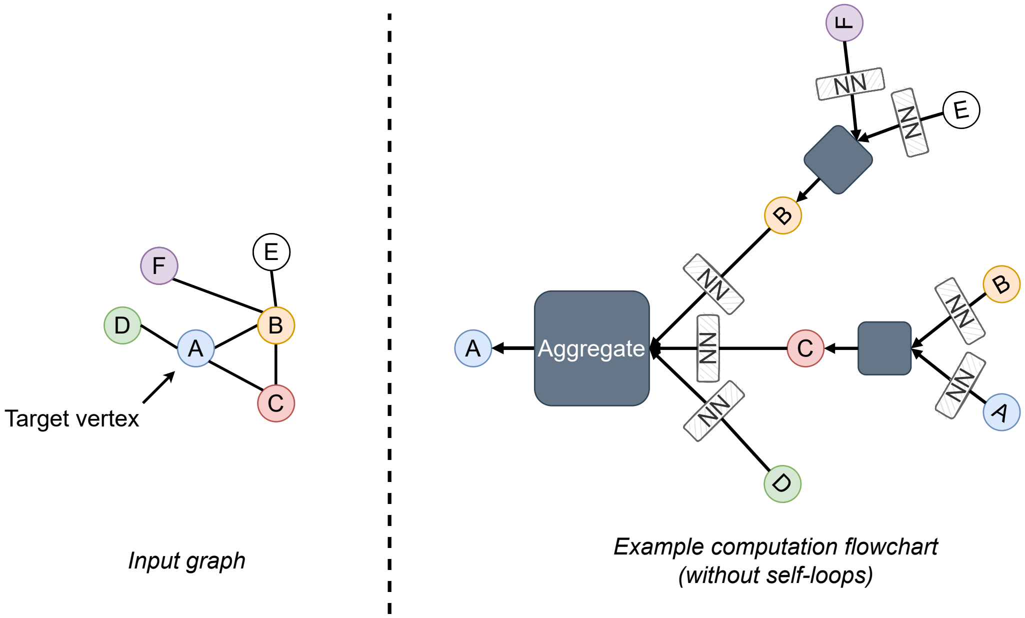


Figure 1.8. Illustrated example of message passing in MPNNs for updating the target node, A. In this example, the function $\gamma^{(k)}(\cdot)$ from Equation 1.27 is dropped after the arbitrary permutation-invariant aggregate, and the function $\phi^{(k)}(\cdot)$ is an arbitrary NN that can be a function of adjacent vertices and their respective edge features/weights.

the LSI system, MPNNs generalize this step to graphs by using permutation-invariant aggregation functions which are invariant to permutations of node orderings, such as the mean, sum and max operations. *Spatial* convolution on graphs works on local neighbourhoods of nodes and understands the properties of a node based on its K local neighbours and its own features. Unlike spectral convolution which may be computationally expensive over very large graphs, spatial convolutions are simple and have produced SOTA results in graph prediction tasks. For example, at Pinterest a spatial GNN known as PinSage [**35**] is paired with a highly efficient sub-graph sampling strategy based on random walks to structure convolutions on very large graphs. In particular, their paper [**35**] uses PinSage to build recommender systems on web-scale graphs, specifically 7.5 billion examples on a graph with 3 billion nodes representing pins and boards, and 18 billion edges across the Pinterest user-pin-board network.

1.5.3. Graph representation learning and geometric deep learning (GDL)

The overarching theme in graph representation learning is centered around around learning latent representations of nodes/vertices in a graph based on their individual features and how they relate to the localized and global topology of the graph, as depicted in Figure 1.9. In other words, the objective is to project nodes on a graph into a learned latent space, where geometric relationships in this space correspond to relationships in the original graph/network [**36**]. Formally, this idea can be analogized to the construct of *encoders* and *decoders* in DSP, in which a parameterized function is learned to encode individual nodes to low-dimensional vectors, or *embeddings*, as depicted in Figure 1.9.

Several methods already exist for embedding nodes on a graph. Particularly, matrix factorization-based methods use a data matrix (e.g., adjacency/Laplacian matrix) as input to learn embeddings for nodes on a graph based on techniques built upon singular value decomposition (SVD). Random walk-based methods generate sequences of nodes through random walks on a graph and then encode the sequences using a word2vec [**37**] model to learn node embeddings. Both of these "schools" of

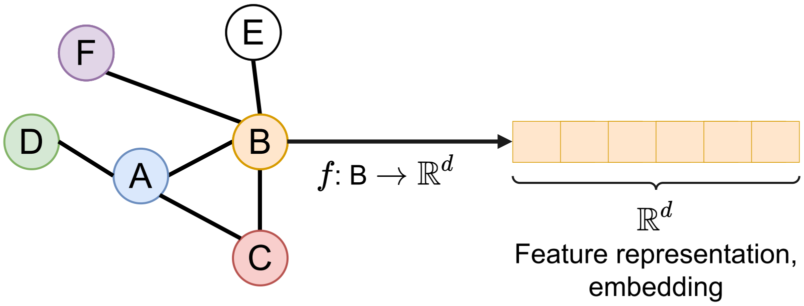


Figure 1.9. Illustration of the vertex embedding problem in graph representation learning where the goal is learn an encoder, f, which maps individual nodes on a graph to a low-dimensional embedding space, $\mathbb{R}^d$.

graph embedding techniques come with a number of caveats and limitations. Particularly, one major disadvantage to both *shallow* embedding paradigms is that most techniques in either do *not* leverage node features. In other words, the node embeddings obtained are strictly based on relational information/features of the graph. Another major, perhaps the most important, limitation to *shallow* embedding methods is that they are inherently *transductive* [**38**]. In other words, these methods can only generate node embeddings for nodes that are only present in the graph during training.

In recent years, neural network-based methods have exploded in popularity thanks to major advances in GNNs methodologies that exploit *relational* inductive biases with graph signals. To alleviate many of the limitations of shallow graph embedding methods, GNNs have become the most popular paradigm to define graph encoders and is often referred to as *geometric deep learning (GDL)*. Unlike spectral GCN approaches which require having a *fixed* graph topology with adjacencies, $\mathbf{A}$, MPNNs

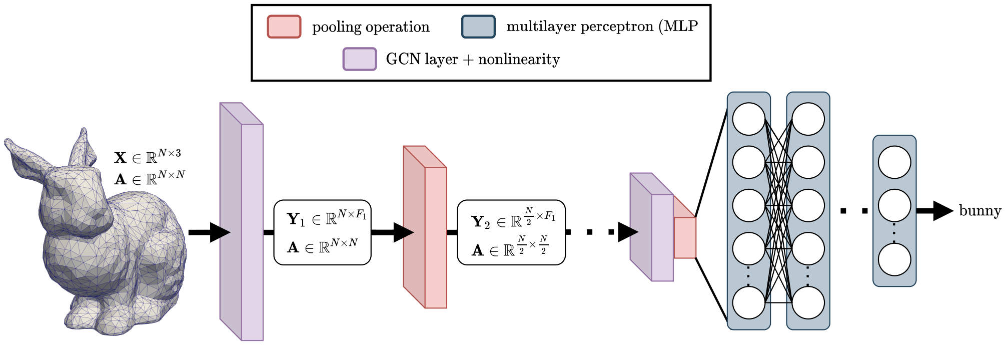

pooling operation
multilayer perceptron (MLP
GCN layer + nonlinearity
$\mathbf{X} \in \mathbb{R}^{N\times 3}$
$\mathbf{A} \in \mathbb{R}^{N\times N}$
$\mathbf{Y}_1 \in \mathbb{R}^{N\times F_1}$
$\mathbf{A} \in \mathbb{R}^{N\times N}$
$\mathbf{Y}_2 \in \mathbb{R}^{\frac{N}{2}\times F_1}$
$\mathbf{A} \in \mathbb{R}^{\frac{N}{2}\times \frac{N}{2}}$
bunny

Figure 1.10. Simple GCN architecture for object classification using surface trimesh representation of a 3D bunny.

make it possible to to design solutions to *inductive* applications, which involve generalizing to unseen nodes after training (e.g., the common cold-start problem in recommender systems [**39, 40**]).

Borrowing the conventional alternating sequence structure of convolutional layers and pooling operations from traditional Euclidean CNNs, GNNs can be constructed using synonymous graph neural networks layers and a variety of graph coarsening techniques [**3, 41, 42, 43, 44**], to reduce the the size of the graph by dropping edges and/or vertices.

The studies performed in this work apply GCNs to trimesh representations of surface boundaries for neuroanatomical regions extracted from segmented, T1-weighted MRIs. Figure 1.10 depicts a sample GCN structure which applies a trimesh coarsening strategy to reduce the number of vertices after each pooling operation by a factor of 2, thus reducing the computational complexity of each subsequent operation by a factor of 2 with respect to the number of vertices.

CHAPTER 2

Brain Morphology in Alzheimer's Disease

Abstract

Alzheimer's disease (AD) is the most common cause of dementia in older adults today. An estimated 5.7 million Americans were estimated to be living with AD since 2018 and today (2022) it is the fifth leading cause of death for adults 65+ years of age and sixth leading for all adults in the United States (US) alone. Patterns of change in brain shape captured by structural neuroimaging methods such as magnetic resonance imaging (MRI) have been identified to occur up to ten years before clinical symptoms that interfere with activities of daily living (ADLs) begin to appear. AD is not a normal part of human aging, although its likelihood can become more prevalent with age. In this chapter, a high level overview of AD neuropathology is presented to motivate a discussion on *in-vivo* discriminative characterizations of brain *shape* in the presence AD.

2.1. Neuropathology of Alzheimer's Disease

Alzheimer's disease (AD) is defined as the co-occurrence of neurofibrillary tangles and amyloid-β plaques in the human brain that end up causing irreversible, progressive brain atrophy. This is what causes the death of brain cells as a result of AD, which leads to severe memory loss and degradation of cognitive ability for independently performing activities of daily living (ADLs) [**45**]. In 2020, an estimated 5.8 million Americans aged 65 years or older had AD. In the United States (US) alone, AD is one of the top ten leading causes of death [**46**]; sixth leading for US adults and fifth leading among US adults aged 65 years or older [**47**]. Death rates from AD continue to increase, unlike heart disease and cancer death rates in the US which are declining [**48**]. With the baby boomer generation now turning 65, the elderly population (aged 65 and over) is expected to double by the year 2030. Although AD is not a part of normal aging, age is the greatest risk factor. Patterns of change in brain shape captured using structural imaging have been identified up to ten years before symptoms manifest, making its early detection possible via *shape*-based *in-vivo* biomarkers. First, we begin by understanding what the root causes of morphological changes correlated to AD neurodegeneration are.

At the neural-level, the human brain is composed of billions of connections between many neurons referred to as *synapses*. At synaptic junctions, neurotransmitters are released as a means of communication for driving cognitive or physical human functions. This is how the human brain communicates within itself to process how we see, feel, smell, hear, think, and remember. Synapse loss and dysfunction is a common feature across several neurodegenerative diseases including dementia, particularly of the AD type. The idea that AD pathology is a disorder of synaptic function is not new [**49**]. The late stages of AD have been shown to involve a significant loss of neurons and synapses [**50**]. Interestingly enough, the possibility of synaptic dysfunction occurring early in prodromal, or mild cognitive impairment

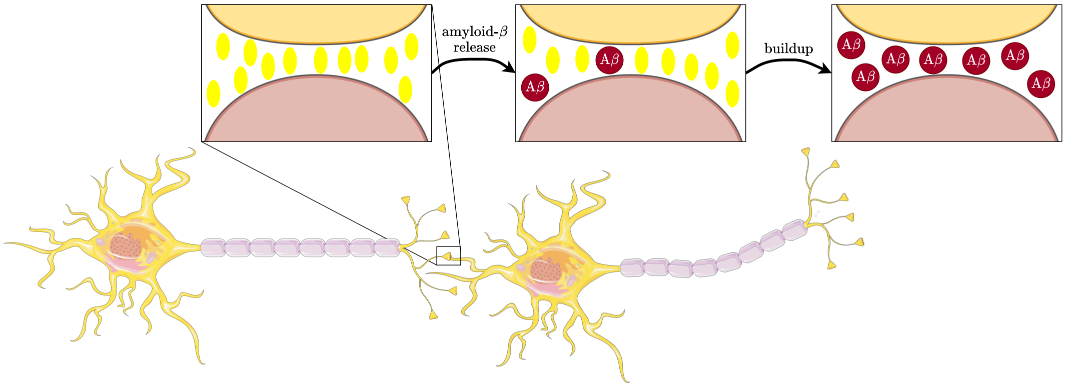


Figure 2.1. Example of synaptic connection between neurons and the buildup of amyloid-β, eventually leading to too much amyloid-β accumulating and forming plaques.

(MCI) stages of the disease, may be present before severe atrophy occurs in the later stages [**51**].

During neural communication, in addition to releasing neurotransmitters, i.e. glutamate, into the synapse, small peptides referred to as amyloid-β are also released. Typically, amyloid-β is cleared away and metabolized by microglia, often referred to as the "janitor cells" of the brain. When amyloid-β begins to accumulate, either by too much being released or not enough being cleared away, amyloid-β begins to pile up (depicted in Figure 2.1) and binds to itself forming aggregates referred to as amyloid-β *plaques*. At the tipping point of amyloid-β accumulation, synaptic dysfunction begins to occur due to inflammation and cellular damage. Microglia-mediated synaptic loss also begins to occur as a result of microglia becoming hyperactivated [**52**]. As the level of amyloid-β reaches a tipping point, there is a rapid spread of a crucial neural transport protein species referred to as τ, which becomes

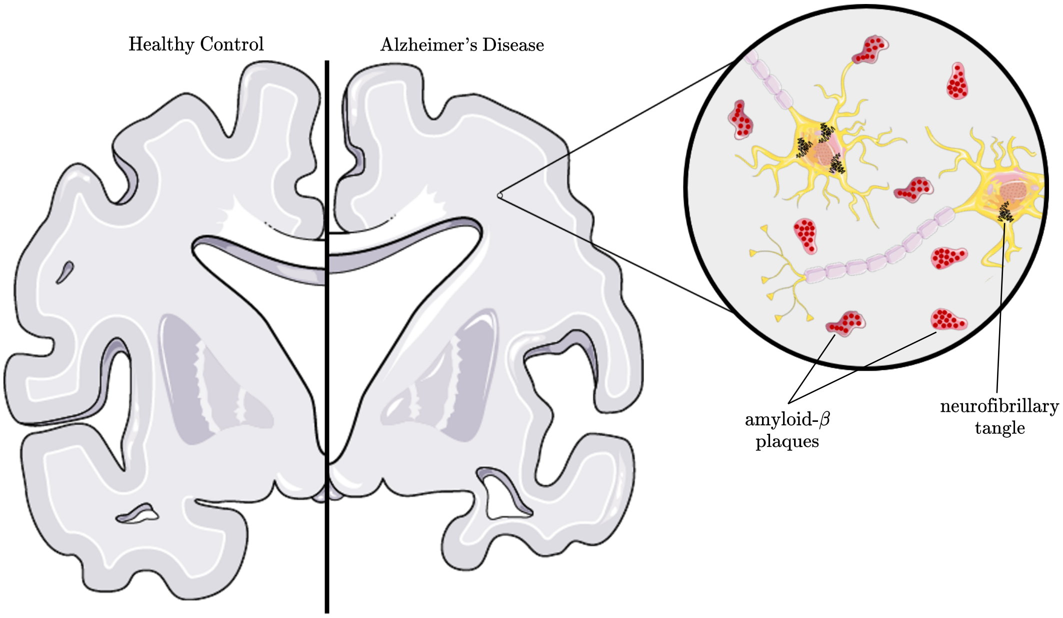


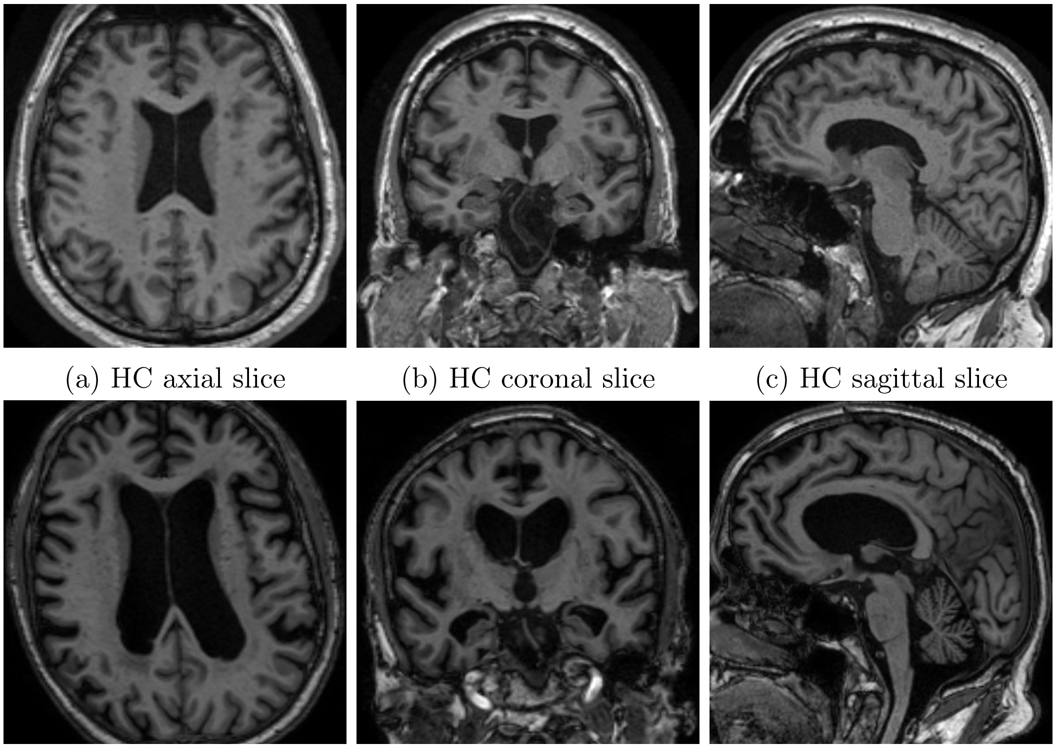
Figure 2.2. Amyloid-β plaques and neurofibrillary tangles with τ-proteins found in the brains of subjects with severe Alzheimer's disease.

hyperphosphorylated and twists itself into neurofibrillary tangles (depicted in Figure 2.2), which end up choking off neurons from within by blocking neural transport systems, thus harming the synaptic communication between neurons.

In AD, as neurons get injured and die throughout the disease's progression, connections between networks of neurons collapse and as a result: several brain regions progressively shrink. Currently, there is no validated *in-vivo* biomarker that can be used to directly measure synaptic integrity over time. Instead, synaptic dysfunction is inferred from the measurement of several different parameters in subjects living with AD, the most direct method being neuroanatomical (i.e. magnetic resonance imaging (MRI)) and functional (i.e. positron emission tomography (PET)). Structural measures of the brain captured using MRI can be used as biomarkers to stage the progression of AD progression, even before clinical symptoms manifest. Coupé *et al.* [**53**] recently modeled conservative lifespan trajectories of structural imaging biomarkers in AD that provide evidence in the early divergence of their AD models from the normal aging trajectory; starting at the hippocampus prior to 40 years of age, and expanding to the lateral ventricles and the amygdala around 40 years. Their results pointing to the hippocampus as the first brain region diverging from a normal aging model is in accordance with previous morphometric studies focused on the prodromal phase of the disease [**54, 55, 56, 57**]. The early divergence of the amygdala morphology in the AD model by Coupé *et al.* [**53**] is not surprising considering the role of emotion in memory. Degradations in emotional processing have been identified in subjects living with AD, expectedly with atrophy in the amygdala [**58**]. Until pioneering studies from New York University in 1989 [**59**], most early work did not examine the medial temporal lobe (MTL), the part of the brain with the highest density of AD histopathological markers (amyloid-β plaques and neurofibrillary tangles discussed earlier).

2.2. Structural Neuroimaging in Alzheimer's Disease

Given that AD is defined by its histopathology, it would be assumed that one way to track the spread of AD is revealing the plaques and tangles in the living brain using neuroimaging. However, clinical symptoms are not directly caused by the amyloid-β plaques and neurofibrillary τ tangles, but rather by the death of neurons and, in particular, the loss of their connections made through synapses [**60, 61**]. Structural differences in the brains of subjects with AD, when compared to healthy controls



(a) HC axial slice (b) HC coronal slice (c) HC sagittal slice

(d) AD axial slice (e) AD coronal slice (f) AD sagittal slice

Figure 2.3. Axial, coronal, and sagittal slices of T1-weighted MRI for randomly selected de-identified healthy control (HC, top row) subject and a randomly selected de-identified subject with Alzheimer's disease (AD, bottom row), respectively. Each column corresponds to a distinct type of slice/plane: axial, coronal, and sagittal respectively.

(HCs) are depicted in Figure 2.2, using arbitrary axial, coronal, and sagittal slices from real-world, de-identified T1-weighted MRIs.

Monitoring neuroanatomical MRI biomarkers in AD pathology is not the sole test used to determine diagnosis. No single test can determine AD diagnosis. Diagnoses are made by determining the presence of a combination of multiple symptoms and ruling out other causes of dementia. This type of evaluation would involve careful medical evaluation of mental status (i.e. mini-mental state examination (MMSE)), physical and neurological examinations, blood analyses, and brain imaging including MRI, computed tomography (CT), and positron emission tomography (PET). With CT, physicians can image 3D volumes of internal organs, bones, soft tissue, and blood vessels to rule out other causes of dementia. In PET, a small amount of radioactive material, often referred to as a radiotracer, is used to measure metabolic activity that is used to infer functional activity in the brain in a 3D image volume. Aside from neuroanatomical measures, functional measures such as PET can be used to image the buildup of amyloid-β plaques living in the brains of those with AD.

Although it is not the sole test, structural measures captured using MRI do provide some of the earliest neuroanatomical biomarkers in AD pathology. In a 2012 study [**62**], Tondelli *et al.* demonstrate that structural pathological changes in brain shape occur years before cognitive decline in AD, including earlier MCI stages. In this study, Tondelli *et al.* particularly identify structural MRI changes that are detectable up to ten years before clinical AD diagnosis. On top of being one of the leading causes of death globally, a cure is not yet known either. Given the degenerative nature of AD pathology and other pathologies linked to dementia, the ability for the brain to compensate for this type of progressive injury is significantly impaired over time. Given the growing body of evidence over the last few decades regarding the cognitive compensatory benefits of neuroplasticity, the early detection of structural changes linked to AD pathology would be highly beneficial for medical intervention with promising therapies [**63, 64**] that may enhance training-induced cognitive- and motor-learning to provide an additional compensatory layer of protection against irreversible brain atrophy. Since structural changes have been shown to occur subtly "behind the scenes" before clinical symptoms manifest, and they are used to infer neuronal death caused by amyloid-β buildup and τ-protein tangles, several studies have looked into anti-dementia drug therapies that have been reported to target neuronal injuries induced by glutamate and amyloid-β buildup [**65, 66, 67**]. Additionally anti-dementia drugs have also been reported to play a significant role in maintaining the structure and integrity (preventing further growth) of amyloid-β plaques and neurofibrillary tangles [**68, 69**]. Although these treatments are intended for maintenance and not recovery/repair, early intervention is beneficial in the maintenance of amyloid buildup and neurofibrillary tangles, as well as designing treatment plans that improve the likelihood of the compensatory benefits of neuroplasticity to lessen cognitive injury.

In this work, we extract the surface boundaries of multiple brain structures/regions in the form of watertight, triangular mesh manifolds from segmented T1-weighted MRIs, resulting in smoother, more realistic, and efficient representations of brain *shape* when compared to 3D volumes. Using these efficient mesh representations, we conduct multiple analyses studying automated methods in discriminating pathological AD brains apart from HCs solely based on brain shape, as well as automated generative methods that learn to capture morphological differences in AD brains when compared to HCs by learning from prior examples. In section 2.3, we discuss a recent methodology that primarily focuses on the discriminative characterizations of brain shape.

2.3. Discriminative Characterizations of Brain Shape

Multiples studies in neuroanatomical shape analyses have focused on single regions, i.e. the hippocampus [**70, 71, 72**], or the ventricles [**73, 74, 75, 76**], and in other studies of the cortex, measures such as thickness and gyrification are used [**77**]. In contrast, recent studies such as *BrainPrint* [**78**] have demonstrated the efficacy of monitoring shape information for multiple brain regions instead of a single region in isolation, as a holistic approach to brain morphometry, i.e. the study of brain structure and its change. With *BrainPrint*, Wachinger *et al.* capture compact and discriminative representations of brain morphology by solving the eigenvalue problem of the 2D and 3D Laplace-Beltrami operator on triangular (boundary) and tetrahedral (volumetric) meshes [**79**] extracted from region of interest (ROI) segmentations in T1-weighted MRI scans.

One of the interesting applications Wachinger *et al.* utilize *BrainPrint* [**78**] for is predicting age based on brain shape. On several datasets, they achieve their lowest prediction errors using shape descriptors from multiple brain regions rather than volumetric measurements of individual brain regions in isolation, indicating that shape information from multiple brain regions may contain additional information about subject similarity that is import for predicting age. On the Open-Access Series of Imaging Studies (OASIS) dataset [**80, 81, 82**], their prediction versus ground truth scatter plot shows a least squares regression line fit with Pearson's correlation coefficient, $r = 0.90$. An improvement in prediction accuracy for sex prediction is also indicated using shape information from *BrainPrint* versus volumetric measurements, most likely due to the detailed characterizations of brain morpohology offered by *BrainPrint*. Moreover, a consistent decrease in predictive performance is observed for predictions on the entire subject population of the Alzheimer's Disease Neuroimaging Initiative (ADNI) dataset [**83**] when compared to prediction on only HCs, particularly in age prediction. This decrease in accuracy for age prediction may indicate a change in the normal pattern of aging for subjects with AD or some form of MCI. As observed in Figure 2.3, the pronounced atrophy in subjects with AD may cause the brain morphology of younger subjects with AD to be similar to older subjects without the disease, thus making age prediction more difficult. The age predictions using *BrainPrint* [**78**] can be seen as an estimate of biological age, where the discrepancy in predicted biological age and chronological age can serve as a biomarker for neurodegenerative disease [**84**].

BrainPrint was also used to win the 2014 second prize in the challenge on Computer-Aided Diagnosis of Dementia (CADDementia), for the direct computer-aided shape-based diagnosis of Alzheimer's disease [**85**] with up to 80% accuracy in AD classification using the ADNI dataset [**83**]. That same year, *BrainPrint* was also presented to accurately classify unique subjects via brain shape [**86**] with over 99.8% accuracy across a population of 3,000 unique scans for 700 unique subjects, where each subject had approximately three to six longitudinal scans spread out with a minimum of six months apart. Over the past few decades, shape analysis techniques have slowly integrated into becoming mainstays in medical image analysis [**87**]; one reason being their value in providing efficient priors for volumetric or boundary segmentation. They also hold tremendous value in quantifying shape changes between subjects and populations, particularly in localizing anatomical variations between populations [**72**].

Since the morpohology of organs across a population is highly heterogeneous, quantifying and modeling these shape variations is challenging. In recent years, access to larger-scale imaging datasets have now made it possible to model underlying shape variations using novel approaches in machine learning, particularly deep learning (DL), which scale well with large and diverse datasets [**88**]. These approaches have the ability to learn complex, hierarchical feature representations that have proven to outperform hand-crafted features in a variety of medical imaging applications. For many DL approaches, one of the main reasons for their success is their use of convolutional layers, which take advantage of inductive biases in the form of translation equivariance properties in Euclidean signal; in the case of MRI and PET: discretized 3D volumes which contain voxel image intensities that infer structure and function/activity at locations in 3D space.

DL approaches that operate on 3D shape representations such as point clouds and meshes have only recently been explored. In a recent study, Gutiérrez-Becker *et al.* [**89**] outperformed *BrainPrint* for *in-vivo* HC/AD classification by using an end-to-end DL framework that learns on point cloud representations of multiple brain

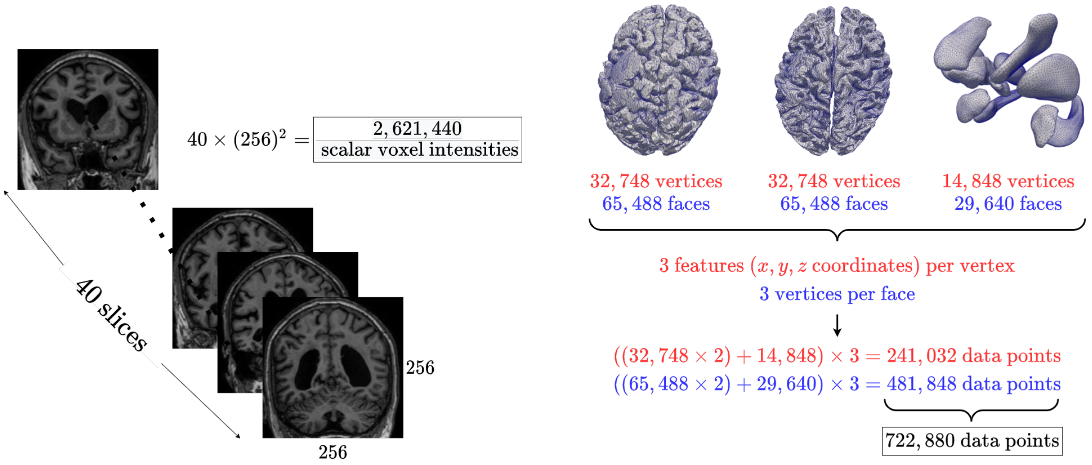


Figure 2.4. Simple walkthrough of dimensionality difference for 40 slices of an MRI scan versus surface mesh representation of cortex (gray and white matter) and subcortical brain region boundaries.

regions per sample. Other recent advances [**90, 91, 92, 93, 94, 95**], demonstrate that *graphs* derived from different types of brain-related connectivity, function, or structure (shape information) are more robust in accuracy and computation time, versus traditional neuroimaging methods. Surface (boundary) meshes of multiple distinct neuroanatomical regions are more robust shape descriptors of brain morphology, rather than direct image intensities. The inferences drawn from utilizing shape descriptors are able to remain robust with respect to intensity changes that may be caused by differing scanner hardware/protocols. Figure 2.4 outlines the significant difference in dimensionality offered by using a triangular surface mesh that quantifies the 3D shape boundaries of multiple neuroanatomical regions when compared to only 40 slices of a single MRI instance. The following chapter dives deeper into the geometric deep learning methodology that is applied on triangular meshes describing the 3D shape of multiple neuroanatomical regions to analyze brain morphology in Alzheimer's disease.

CHAPTER 3

Interpreting Brain Morphology in Association to Alzheimer's Disease Classification Using Spectral Graph Neural Networks

Abstract

In this study, the efficacy of graph-based machine learning on triangular mesh (trimesh) representations of the cortex and subcortical regions for the binary classification of subjects with Alzheimer's disease (AD) vs. healthy controls (HCs) is studied. Deep learning methods for classification tasks that utilize structural neuroimaging often require extensive learning parameters to optimize. Frequently, these approaches for automated *in-vivo* classification also lack visual interpretability for areas in the brain involved in making a prediction. This work: (a) analyzes brain shape using surface information of the cortex and subcortical regions using trimeshes, (b) proposes a residual graph learning architecture using a state-of-the-art graph convolutional network framework offering a significant reduction in learnable parameters versus volumetric approaches, and (c) offers visual interpretability of the network's reasoning via class-specific gradient information that localizes regions of interest in our inputs with respect to a specified outcome.

With our proposed method leveraging the use of cortical and subcortical surface information, we outperform other machine learning methods with a 96.35% testing

accuracy for the AD vs. HC problem. We confirm the validity of our model by observing its performance in a 25-trial Monte Carlo cross-validation. The generated visualization maps in our study show correspondences with current knowledge regarding the structural localization of pathological changes in the brain associated to AD. This chapter is a rewritten version of my published work at the international workshop on Shape in Medical Imaging (ShapeMI), held in conjunction with the 2020 Medical Image Computing and Computer-Assisted Intervention (MICCAI) international conference [**96**].

3.1. Introduction

Previous studies in neurodegenerative pathology, such as Alzheimer's disease (AD), have demonstrated correlations in cortical folding pattern [**97**] and different neurodegenerative pathologies. Specific patterns of atrophy in the cortex and subcortical regions have been linked to AD [**98, 99**]. For example, Ono *et al.* [**97**] discuss a potential to focus on high variability in association cortices like the intermediate sulcus of Jensen. As Pacheco *et al.* [**100**] also point out, widespread cortical thinning and a greater rate of atrophy is present in temporal lobe regions, primarily the left parahippocampal gyrus, for subjects with AD. Furthermore, Jong *et al.* [**101**] discuss irregularities like reduced putamen and thalamus volumes for subjects with AD. In studies focusing on AD, it is common to find bias towards more left-sided atrophy because of the verbal language tests given to assess memory function [**102**]. For example, if asymmetrical atrophy of the language network is more prominent, subjects may perform worse on verbal tests and be diagnosed with dementia earlier.

Machine learning (ML) methods have been a growing area of interest in the automated clinical diagnosis of AD. Prior studies [**103, 104, 105**] discuss the use of a support vector machine (SVM) in unimodal and multimodal imaging pipelines for the automated classification of AD using magnetic resonance imaging (MRI), positron emission tomography (PET), and cerebrospinal fluid (CSF). In recent ML [**106, 107**], the use of MRI and PET imaging in a multimodal convolutional neural network (CNN) for AD diagnosis is discussed. SVM-based approaches [**103, 104, 105**] have historically been hard to interpret, expensive to train, and often serve as the logical choice only when there is enough domain expertise to construct meaningful kernels. Multimodal volumetric CNNs like that of Punjabi *et al.* [**107**], often require a lot of memory and are frequently limited to smaller-batch operations or using lower resolution 3D volumes.

Motivated by 3D object detection via surfaces [**108**], cortical and subcortical irregularities correlated with AD, this study uses surface trimesh manifolds of the cortex and subcortical regions in AD vs. HC classification. Our technique leverages a reduction in computational complexity offered by using localized spectral filtering on graphs [**3**]. This approach has been used in prior work by Parisot *et al.* [**109**] to make similar predictions for AD and Autism using a graph convolutional network (GCN) on a multi-cohort subject population from ADNI and Autism Brain Imaging Data Exchange (ABIDE). Moreover, a recent study by Ranjan *et al.* [**4**] developed a convolutional mesh autoencoder (CoMA) framework using the same GCN basis [**3**] on trimeshes instead, to generate new trimeshes from a learned distribution and reconstruct input trimeshes with a 50% reduction in reconstruction error, all while using 75% fewer parameters than volumetric models.

The interpretability of results from ML models has remained an open issue for highlighting a potential region of interest (ROI) in images for certain predictive outcomes (i.e. true positive). In this study we demonstrate that it is possible to (1) extract meaningful surface trimeshes of the cortex and subcortical regions, (2) achieve accurate predictions for the clinical binary classification of AD using trimeshes, (3) extract class-discriminative localization maps for interpretable ROI maps, and (4) reduce the number of learnable parameters when compared to intensity-based volumetric approaches.

3.2. Methods

Data used in the preparation of this chapter was obtained from the Alzheimer's Disease Neuroimaging Initiative (ADNI) database (https://adni.loni.usc.edu). The ADNI was launched in 2003 as a public-private partnership, led by Principal Investigator Michael W. Weiner, MD. The primary goal of Alzheimer's Disease Neuroimaging Initiative (ADNI) has been to test whether serial magnetic resonance imaging (MRI), positron emission tomography (PET), other biological markers, and clinical and neuropsychological assessment can be combined to measure the progression of mild cognitive impairment (MCI) and early Alzheimer's disease (AD).

3.2.1. Localized spectral filtering on graphs using Chebyshev polynomials

Spectral graph convolution methods inherit ideas from a graph signal processing (GSP) perspective as described in chapter 1 and the GSP review by Wu *et al.* [**110**]. Like Defferrard *et al.* [**3**], our work focuses on using trimeshes, interpreted as undirected graphs and defined by a finite set of vertices, $\mathcal{V}$, with $N = |\mathcal{V}|$ vertices, and a corresponding set of edges, $\mathcal{E}$, with scalar edge weights, $w_{ij} = w_{ji}$ for $\{e_{ij}, e_{ji}\} \in \mathcal{E}$, which are stored in the i/j^{th} rows and j/i^{th} columns of the adjacency matrix, $\mathbf{A} \in \mathbb{R}^{N\times N}$. As defined in Table 1.1, a graph's vertex features are defined using the vertex feature matrix $\mathbf{X} \in \mathbb{R}^{N\times F}$ where each column, $\mathbf{x} \in \mathbb{R}^{N}$, represents the feature vector for a particular shared feature across all N vertices, $\{v_n : n = 1, \ldots, N\} \in \mathcal{V}$.

A great emphasis in GSP is placed on the symmetric normalized graph Laplacian, $\mathbf{L} = \mathbf{I}_N - \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$, where $\mathbf{I}_N$ is the identity matrix and $(\mathbf{D})_{ii} = \sum_j (\mathbf{A})_{ij}$ is the diagonal matrix of vertex degrees. In this work, we refer to the symmetric Laplacian $\mathbf{L}^{\text{sym}}$ as $\mathbf{L}$ and use it as the graph shift within GCN layers. Following Equation 1.18, $\mathbf{L}$ can also be factored via the eigendecomposition $\mathbf{L} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^{\intercal}$, where $\mathbf{V} \in \mathbb{R}^{N\times N}$ is the complete set of orthonormal eigenvectors for $\mathbf{L}$ and $\mathbf{\Lambda} = diag\left([\lambda_1, \ldots, \lambda_N]\right) \in \mathbb{R}^{N\times N}$ is the corresponding set of eigenvalues. Given the spectral filter, h_θ, defined in the graph Fourier space [**17**] as a polynomial of the Laplacian, $\mathbf{L}$, and $\mathbf{V}$'s orthonormality, we can filter $\mathbf{x}$ via multiplication such that

$$h_\theta *_{\mathcal{G}} \mathbf{x} = h_\theta(\mathbf{L})\mathbf{x} = h_\theta\left(\mathbf{V}\mathbf{\Lambda}\mathbf{V}^{\intercal}\right)\mathbf{x} = \mathbf{V}h_\theta(\mathbf{\Lambda})\mathbf{V}^{\intercal}\mathbf{x}, \tag{3.1}$$

where θ is the set of learnable parameters for the filter h_θ and $*_\mathcal{G}$ is the spectral convolution operator notation borrowed from the work of Defferrard *et al.* [**3**]. In this context, graph convolution is implicitly performed by using the duality property of the Fourier transform where the output is computed via multiplication by a spectral filter in the graph Fourier space, followed by the IGFT of the product.

This work uses Chebyshev polynomials of the first kind [**3, 111**] to approximate h_θ using the graph's spectrum such that

$$h_\theta(\tilde{\mathbf{L}}) = \sum_{k=0}^{K-1} \theta_k T_k(\tilde{\mathbf{L}}), \tag{3.2}$$

for the scaled Laplacian $\tilde{\mathbf{L}} = \frac{2\mathbf{L}}{\lambda_{max}} - \mathbf{I}_N$, where λ_{max} is the largest eigenvalue in $\mathbf{\Lambda}$, and K can be interpreted as the kernel size. The kernel size, K, which also corresponds to the Chebyshev polynomial's order, directly relates to the number of vertices aggregated within K-*hops* from each vertex to compute the output of a convolution for each corresponding input vertex. Chebyshev polynomials of the first kind are defined by the recurrence relation, $T_k(\tilde{\mathbf{L}}) = 2\tilde{\mathbf{L}}T_{k-1}(\tilde{\mathbf{L}}) - T_{k-2}(\tilde{\mathbf{L}})$ for $T_0(\tilde{\mathbf{L}}) = \mathbf{I}$ and $T_1(\tilde{\mathbf{L}}) = \tilde{\mathbf{L}}$ [**3**] and the ordinary generating function for T_n

$$\sum_{n=0}^{\infty} T_n\left(x\right) t^n = \frac{1 - tx}{1 = 2tx + t^2}.$$

3.2.2. Trimesh extraction of cortical ribbon and subcortical regions

Using FreeSurfer v6.0 [**112**], all MRIs were denoised followed by field inhomogeneity correction, and intensity and spatial normalization. Inner cortical surfaces (interface between gray and white matter) and outer cortical surfaces (CSF/gray matter interface) were extracted and automatically corrected for topological defects. Additionally, seven subcortical regions per hemisphere were segmented (amygdala, nucleus accumbens, caudate, hippocampus, pallidum, putamen, thalamus) and modeled into surface trimeshes using the SPHARM-PDM toolbox (https://www.nitrc.org/projects/spharm-pdm).

Surfaces were inflated, parameterized to a sphere, and registered to a corresponding spherical surface template using a rigid-body registration to preserve the cortical [**112**] and subcortical [**113**] anatomy. Surface templates were converted to trimeshes using their triangulation schemes. A scalar edge weight w_{ij} was assigned to edge $e_{ij} \in \mathcal{E}$ connecting vertices v_i and v_j, as a function of their geodesic distance ψ_{ij} defined as

$$w_{ij} = w_{ji} = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{\psi_{ij}}{\sigma}\right)^2}. \tag{3.3}$$

Surface templates were parcellated using a hierarchical bipartite partitioning of their corresponding trimesh. Starting with their initial trimesh representation of densely triangulated surfaces, spectral clustering was used to define two partitions. These two groups were then each separated yielding four child partitions, and this process was repeated until the average distance across neighbor partitions was below

Table 3.1. Vertex, edge, and face counts (per hemisphere) for trimesh surfaces of corresponding brain regions.

Regions		**Vertices**, N	**Edges**	**Faces**
subcortical	amygdala	512	3,060	1,020
	caudate	1,024	6,132	2,044
	hippocampus	2,048	12,276	4,092
	nucleus accumbens	256	1,524	508
	pallidum	512	3,060	1,020
	putamen	1,024	6,132	2,044
	thalamus	2,048	12,276	4,092
cortex	white matter surface	16,374	98,232	32,744
	pial surface	16,374	98,232	32,744

2.5 mm. For each partition, the central vertex was defined as the vertex whose centrality was highest and the distance across two partitions was defined as the geodesic distance (in mm) across the central vertices. Two partitions were neighbors if at least one vertex in each partition were connected. Finally, partitions were numbered so that partitions $2i$ and $2i+1$ at level L had the same parent partition i at level $L-1$. Therefore, for each level a graph was obtained such that the vertices of the graph were the central vertices of the partitions and the edges across neighboring vertices were weighted as in Equation 3.3. This serves as an improvement upon the work of Defferrard *et al.* [**3**] to ensure that no singleton is ever produced by pooling operations for the cortex and subcortical regions. At the finest level, trimeshes consisted of the vertex, edge, and face counts provided in Table 3.1.

Vertex features were defined as the Cartesian coordinates of the surface vertices in the subjects' native space, prior to spherical surface registration onto the spherical template. Similar studies, like that of Gutiérrez-Becker *et al.* [**89**], implement

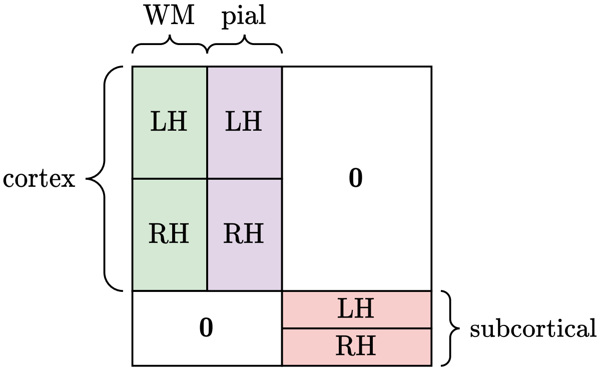


Figure 3.1. Illustration of block-diagonalization step for combining cortical and subcortical vertices into single vertex feature matrix for whole-brain graph. For each scan, the vertex feature matrices for WM and pial surfaces are concatenated along the feature dimension (coordinates) and hemispheres are concatenated along the vertex dimension (rows). A row-wise concatenation (with respect to each hemisphere) of the subcortical vertex feature matrices is also performed in alphabetical order of their names per hemisphere (listed alphabetically in Table 3.1).

"rotation network" modules as the first few layers of their NN architecture to aid in correcting and aligning their samples to a common template. Performing our template registration as an additional preprocessing step reduces the complexity of our architecture and eliminates the need of incorporating an "alignment" term to our cost function [**89**], by having a 1:1 correspondence of vertices across trimeshes registered to the same template.

Cortical ribbon surface vertices were assigned 6 features: the corresponding x, y, and z coordinates of both the white matter (WM) and pial surface vertices for each sample. This was decided because these trimeshes describing the surface boundaries of the cortical ribbon, use the same underlying mesh topology and therefore equivalent “faces” with different coordinates for the vertices of their respective triangles. Similar to the cortex, subcortical vertices had 3 features: their corresponding x, y, and z coordinates in the native space as well. To maintain the same number of features for all vertices per trimesh, the corresponding cortical and subcortical vertex feature matrices were block-diagonalized into a single vertex feature matrix per scan such that $\mathbf{X} \in \mathbb{R}^{47,616 \times 9}$, as depicted in Figure 3.1. This way, every vertex in the concatenated feature matrix contains the same number of features and the coordinate features *not* corresponding to a particular vertex are automatically zeroed, therefore disabling their influence on vertices that *do* correspond to those coordinate features. Sample surface boundary trimeshes from a randomly selected HC subject and another random subject with AD are illustrated in Figure 3.2.

3.2.3. Residual network architecture

Motivated by the success of residual NN architectures in DL frameworks for image recognition [**114**], the NN models used in this work are based on a residual learning architecture composed of “residual learning blocks” referred to as ResBlocks, depicted in Figure 3.3. These function by using an element-wise addition of the output from a previous block to the output of the current block. This methodology was demonstrated to allow for the training of deeper NN architectures, with the intuition that adding more layers allows for progressively learning more complex features within the architecture [**114, 115**].

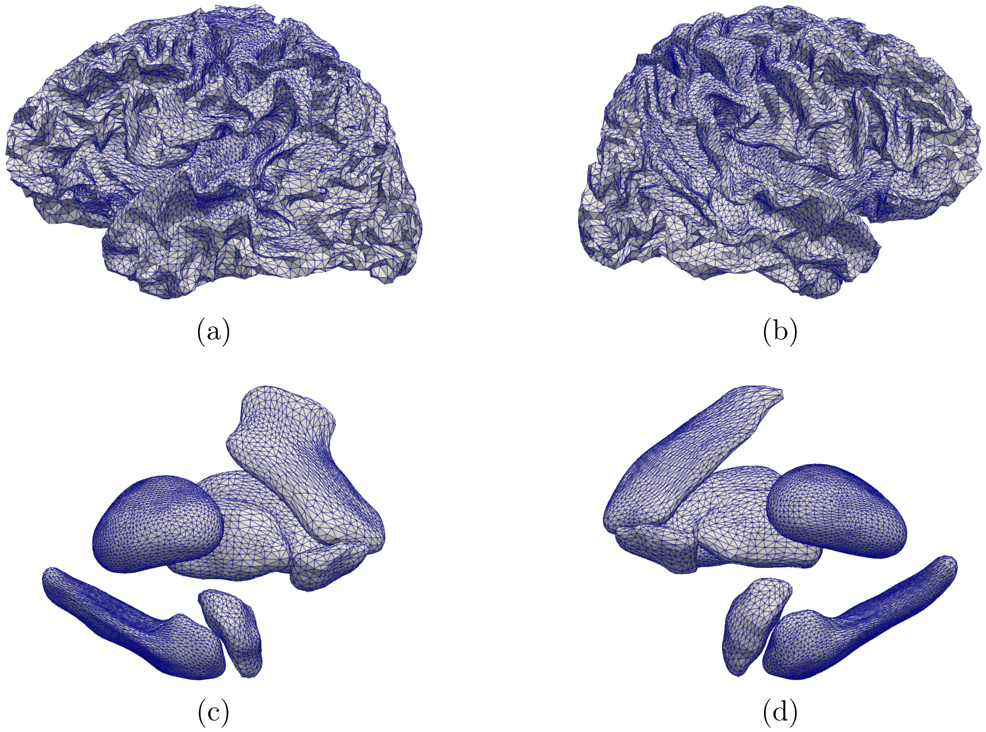


Figure 3.2. Lateral views of LH/RH (a/b) white matter surface trimeshes for randomly selected HC subject (blue) and medial views of LH/RH (c/d) trimeshes for subcortical regions of randomly selected subject with AD.

Within a ResBlock, a spectral graph convolutional layer is used as a linear mapping tool to map $F_{\text{in}} \mapsto F_{\text{out}}$ features per vertex. Analogous to traditional convolution with padding to preserve the input's dimensionality, spectral graph convolution preserves the number of vertices for graph inputs. A frequent issue in DL with training deep NNs is in the *internal covariate shift* in the distribution of inputs to layers with a model [**116**]. batch normalization (BN) is applied within each ResBlock prior to each ReLU activation and GCN layer to circumvent this issue and prevent our

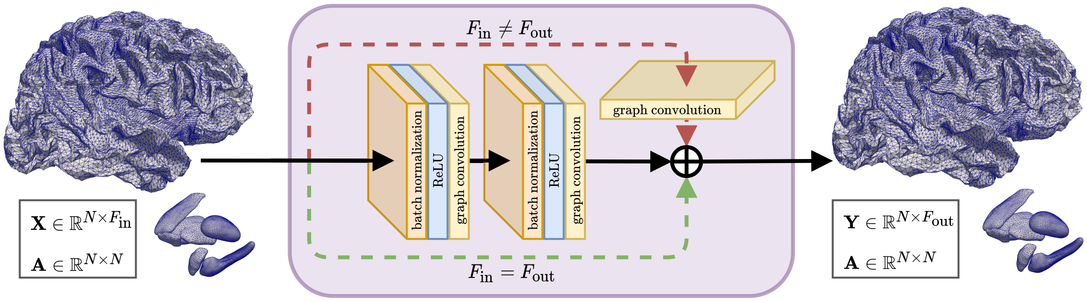

$F_{\text{in}} \neq F_{\text{out}}$
batch normalization
ReLU
graph convolution
batch normalization
ReLU
graph convolution
graph convolution
$F_{\text{in}} = F_{\text{out}}$
$\mathbf{X} \in \mathbb{R}^{N \times F_{\text{in}}}$
$\mathbf{A} \in \mathbb{R}^{N \times N}$
$\mathbf{Y} \in \mathbb{R}^{N \times F_{\text{out}}}$
$\mathbf{A} \in \mathbb{R}^{N \times N}$

Figure 3.3. Residual learning block (ResBlock) module used for GCN architecture in this study. Batch normalization (BN) (depicted in orange) is applied after spectral graph convolution (depicted in yellow). If the number of input features F_{in} does not match the number of filters F_{out} at a layer, the top (red) branch of the ResBlock uses spectral graph convolution as a linear mapping tool for mapping $F_{\text{in}} \mapsto F_{\text{out}}$ per vertex, to provide a dimensionality match with the addition of the main branch's output. Otherwise, the input of a ResBlock is added to the main branch output (green track). An element-wise ReLU activation function is used within the hidden layers of the ResBlock and as the final activation.

GCN model from "forever chasing a moving target," by standardizing inputs to layers within the network. This follows the convention used by other successful DL architectures in computer vision [**114, 115**].

Using ResBlocks, max-pooling operations as defined by Defferrard *et al.* [**3**] (discussed in following section), and a MLP, the complete residual GCN architecture used in this study is illustrated in Figure 3.4. An additional ResBlock, referred to as a "post-ResBlock," was introduced prior to the MLP as a final expansion of the receptive field for the coarsened vertex feature maps.

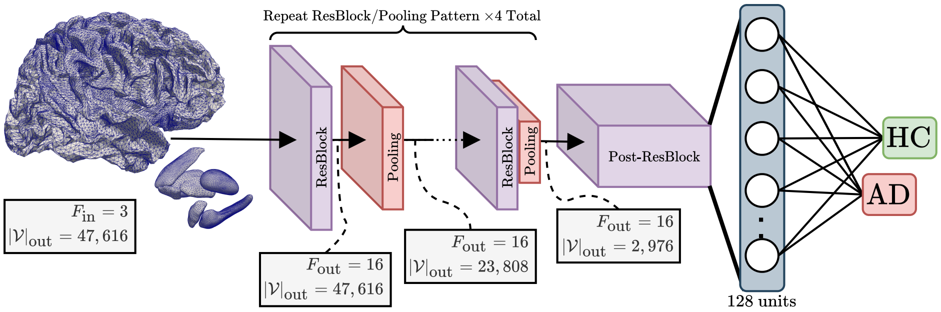

Repeat ResBlock/Pooling Pattern ×4 Total
$F_{\text{in}} = 3$
$|\mathcal{V}|_{\text{out}} = 47,616$
ResBlock
Pooling
$F_{\text{out}} = 16$
$|\mathcal{V}|_{\text{out}} = 47,616$
ResBlock
Pooling
$F_{\text{out}} = 16$
$|\mathcal{V}|_{\text{out}} = 23,808$
Post-ResBlock
$F_{\text{out}} = 16$
$|\mathcal{V}|_{\text{out}} = 2,976$
128 units
HC
AD

Figure 3.4. Residual GCN used for the binary classification of AD. In this study, max-pooling operations are used to downsample the vertex dimension by a factor of 2. A standard MLP layer is applied at the end of the network architecture to map the vertex feature maps to the corresponding label for classification.

3.2.4. Graph coarsening/pooling

Signal pooling, as discussed for traditional n-dimensional signals in Section B.2 of the chapter 6.5, is useful for reducing the size of an input signal (i.e. height and width of 2D RGB image). On graphs, the pooling (or coarsening) operation requires a labeling of meaningful neighborhoods in graphs in order to aggregate similar vertices with one another. Furthermore, performing this in sequential layers within a NN model is the equivalent to multi-scale clustering on graphs and it can be challenging to preserve local geometric structures. In fact, Bui *et al.* [**117**] prove that graph clustering is NP-hard and requires approximations.

Defferrard *et al.* [**3**] use the multi-level Graclus approach [**1**], built on Metis [**118**], to approximate graph pooling. This approach uses a greedy algorithm to compute successive coarsened versions of a given graph, while minimizing several popular spectral clustering objectives, from which they select the normalized cut [**119**].

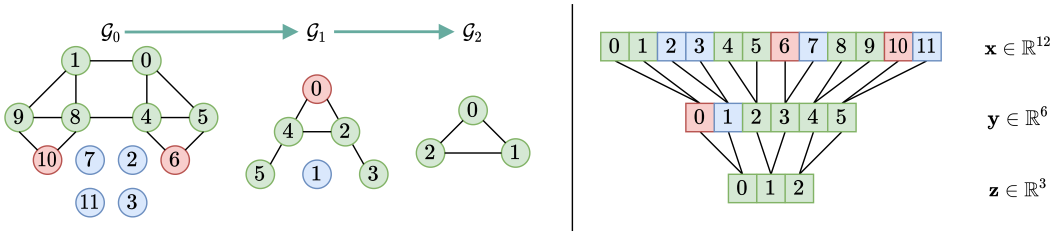


Figure 3.5. Visual walkthrough of graph pooling using Graclus [**1**] to coarsen a graph with 12 vertices down to 3 vertices. Every vertex at each level of coarsening is labeled with respect to the ordering from the binary tree arrangement obtained on the right-hand side. In this instance, the starting graph $\mathcal{G}_0$ is coarsened twice, by a factor of 2 each time, to obtain graph $\mathcal{G}_2$. Red vertices correspond to singletons at each level of coarsening, blue vertices correspond to zero-valued placeholder vertices that are ignored during pooling operations, and the green vertices are "marked" vertices that are each merged with another marked vertex to produce corresponding child vertices in the binary tree structure.

Starting with an arbitrary graph $\mathcal{G}_0$, the first step in graph coarsening is to establish the number of "levels" of potential coarsening. To do this, it is first important to know the number of vertices $|\mathcal{V}_0|$, in $\mathcal{G}_0$ and select a number of levels that is $\leq |\mathcal{V}_0| - 2$. The specifics of this will become clearer with further explanation, but for now it is noteworthy to keep in mind that if $|\mathcal{V}_0| = 3$ and 2 levels of coarsening are specified, this will eventually reduce the graph down to 1 vertex from 3 vertices, which is not a useful graph (floating singular vertex).

Once the number of coarsening levels is established, the attention is then turned on turning $\mathcal{G}_0 \mapsto \mathcal{G}_1$ first, as depicted by Figure 3.5. First, Graclus' [**1**] greedy rule consists of randomly selecting an "unmarked" vertex, $v_i \in \mathcal{V}_0$, and "marking" the neighbor, $v_j \in \mathcal{V}_0$, that maximizes the relationship

$$\max_{v_i, v_j} \left\{ w_{ij} \left(\frac{1}{d_i} + \frac{1}{d_j} \right) \right\}, \tag{3.4}$$

for the scalar edge weight $w_{ij} \in \mathcal{G}_0$. Once $v_j \in \mathcal{V}_0$ is determined, both v_i and v_j are marked as merged vertices and an arbitrary placeholder vertex $v_i \in \mathcal{G}_1$ is created and marked as their child vertex. This process is repeated for $\mathcal{G}_0$ until all vertices $v_i \in \mathcal{V}_0$ are marked and/or only singletons remain. For example, by following Figure 3.5 for context, it is observed that after 1 level of Graclus, the vertex pair $(v_0, v_1) \in \mathcal{V}_0$ are merged to form $v_0 \in \mathcal{V}_1$. Meanwhile, v_6 is a singleton that is carried over to $\mathcal{G}_1$ to become $v_3 \in \mathcal{V}_1$.

For an arbitrary graph $\mathcal{G}_0$, assume that Graclus is performed twice to produce two levels of coarsening ($\mathcal{G}_0 \rightarrow \mathcal{G}_1 \rightarrow \mathcal{G}_2$). After coarsening, the remaining vertices of the coarsened graphs after $\mathcal{G}_0$ are still not arranged in a meaningful way to perform a pooling operation of consistent sizing to go from $\mathcal{G}_0$ down to $\mathcal{G}_{1,2}$. To solve this, we first rearrange the vertices of the coarsest graph, $\mathcal{V}_2$, into a arbitrarily-ordered 1D vector. Based on this vector, the vertices at the preceding levels, $\mathcal{V}_{0,1}$, are then rearranged correspondingly, based on the parent-child relationships determined during coarsening. Following Figure 3.5, a placeholder vertex is inserted as the "co-parent" for every singleton carried over between coarsening levels. This is repeated until every vertex (except those in $\mathcal{V}_0$) has two parent vertices at the preceding level, creating a balanced binary tree, as depicted on the right in Figure 3.5. Afterwards, vertices at the coarsest level are arbitrarily permuted and the ordering is then propagated to the finer levels, producing a canonical ordering of the vertices in $\mathcal{V}_0$.

Using this balanced binary tree structure and treating the vertices at the finest level, $\mathcal{V}_0$, as a traditional 1D signal, 1D pooling can be applied (2D pooling described in section B.2 of the chapter 6.5). As a consequence of using a balanced binary tree structure to get a canonical ordering of vertices across different levels of graph coarsening, 1D-pooling operations can only be performed using power-of-2 pooling window sizes. Placeholder vertices, which are inserted to compensate for singleton-carryovers, are only considered for outlining individual pooling windows and their values are set to neutral values that are ignored during pooling. For example, after Graclus is performed on $\mathcal{G}_0$, if a 4-element max-pooling operation was performed on the updated $\mathcal{V}_0$ (now including placeholder vertices), the operation would reduce to

$$\begin{aligned}\mathbf{z} \in \mathbb{R}^3 &= \text{max-pool}_4\left([v_0, v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_9, v_{10}, v_{11}]\right)\\ &= \left[\max\left(v_0, v_1, v_2, v_3\right), \max\left(v_4, v_5, v_6, v_7\right), \max\left(v_8, v_9, v_{10}, v_{11}\right)\right]\\ &= \left[\max\left(v_0, v_1\right), \max\left(v_4, v_5, v_6\right), \max\left(v_8, v_9, v_{10}\right)\right],\end{aligned}$$

where the placeholder vertices, $v_{2,3,7,11} \in \mathcal{V}_0$, are ignored in the final steps of pooling.

Keep in mind, when selecting a 1D-pooling window of a power-of-2 size, the number of coarsening levels traversed in the binary tree is $\log_2$ of the size. In this particular case, because the expanded graph $\mathcal{G}_0$ (including placeholder vertices) is of length 12, and a length-4 pooling window is used, the output graph $\mathcal{G}_2$ is $\log_2(4) = 2$ levels down from $\mathcal{G}_0$ in the coarsening hierarchy.

3.2.5. Adaptation of Grad-CAM to trimeshes

The interpretability of CNN decision-making was addressed by Selvaraju *et al.* [**120**] via Grad-CAM to provide interpretable localized heatmaps that weigh the "importance" of areas in a 2D input image which are indicative of certain predictions after training a CNN. In their work, Selvaraju *et al.* [**120**] feed images to CNNs and gradients for each class (i.e. logits prior to softmax) are extracted at the final convolutional layer. Using these gradients, global average pooling (GAP) is applied across the feature maps of the final convolutional layer, for each class c, to extract the "neuron importance weights," $\alpha_c^{(f)} \in \mathbb{R}^{c \times f}$, whose formulation we re-adapted for meshes such that

$$\alpha_c^{(f)} = \frac{1}{N} \sum_i \frac{\partial y_c}{\partial A_i^{(f)}}, \tag{3.5}$$

where y_c corresponds to the class-score for class c, and $A_i^{(f)}$ refers to the value at vertex v_i for the vertex feature map $A^{(f)} \in \mathbb{R}^N$. The set of neuron importance weights, $\alpha_c^{(f)}$, is then projected back onto each feature map, $A^{(f)}$, to compute the class activation map (CAM), M_c, for each class, c, such that

$$M_c = \mathrm{ReLU}\left(\sum_f \alpha_c^{(f)} A^{(f)} \right) \in \mathbb{R}^N. \tag{3.6}$$

ReLU is applied to the linear sum of feature maps with respect to their neuron importance weights because we are only interested in the features that have a positive influence on the class interest [**120**].

As a consequence of multiple pooling operations within our GCN architecture, the extracted CAMs with respect to the vertices of the coarsest trimesh at the final convolutional layer are less focused to specific surface locations. Therefore, the CAMs of the coarsest trimesh are up-sampled back up to the same number of vertices at the finest level $\mathcal{G}_0$ using a trivial interpolation to provide a direct overlay on the original input trimesh.

3.3. Experiment

3.3.1. Dataset and preprocessing

T1-weighted MRIs from ADNI [**121, 83**] were selected with AD/HC diagnosis labels given up to 2 months after the corresponding scan. This was taken as a precaution to ensure that each diagnosis had clinical justification. The dataset in our study consisted of 1,191 different scans for 435 unique subjects. subsection 3.3.2 outlines our stratified data splitting strategy to ensure no data leakage occurs at the subject level across the training, validation, and testing sets [**122**].

Trimeshes for each MRI were extracted following the trimesh preprocessing steps described in subsection 3.2.2. The spatial standard deviation from Equation 3.3, σ, was set to 2 ad-hoc. The visual quality for each trimesh was assessed manually via a direct overlay over slices of the corresponding MRI. Laplacians for the cortex and each subcortical region were block-diagonalized to create one overall $\mathbf{L}$ representing a single trimesh with multiple connected components. Extracted feature matrices for each sample were min-max normalized per feature to the interval $[-1, 1]$ prior to feeding batches of data into the networks. The added zeros during block-diagonalization (as discussed in subsection 3.2.2) were ignored during each normalization step.

3.3.2. Network architecture and training

Extra care was taken in the shuffling of samples to avoid bias from subject overlap in our cross-validation [**122**]. A custom dataset splitting function was implemented such that the distribution of labels was preserved amongst each set while also ensuring to avoid subject overlap. 20% of the samples were selected at random for the testing set. Of the remaining 80%, 20% of those were withheld as the validation set, while the remaining belonged to the training set. A 25-trial Monte Carlo cross-validation was performed using this data split scheme.

The architecture in Figure 3.4 was implemented using 16 filters per GCN layer (not including the post-ResBlock), Chebyshev polynomials of order $K = 3$, and pooling windows of size $p = 2$. Four alternating ResBlock and pooling layers were cascaded as shown in Figure 3.4 prior to the post-ResBlock. The number of units at the post-ResBlock and MLP layer was 128. Our GCN was optimized by minimizing a standard binary cross-entropy (BCE) loss function

$$\mathcal{L} = -\frac{1}{M}\sum_{n=1}^{M} y_m \log(\hat{y_m}) + (1 - y_m)\log\left(1 - \hat{y_m}\right), \tag{3.7}$$

where $\hat{y_m}$ is the predicted class for the m^{th} sample of M total samples and y_m is the ground truth label for the corresponding sample index, m.

Networks were trained using batches of 32 samples per step for 100 epochs in each Monte Carlo trial. The Adam [**123**] optimizer was used with a learning rate of 5×10^{-4} and a learning rate decay of 0.999. Experiments were implemented in Python 3.6 using Tensorflow 1.13.4 using an NVIDIA GeForce GTX TITAN Z GPU in a Dell Precision Tower 7910 with Linux Mint 19.2.

3.4. Results and Discussion

3.4.1. AD vs. HC classification

Our baseline cross-validation includes the same MLP classifier, ridge, classifier, and a 100-estimator random forest (RF) classifier set up by Parisot *et al.* [**109**], where a similar graph approach is also used in the classification of AD based on subject population graphs. The MLP designed was analogous to the GCN design by Parisot *et al.* [**109**] such that the number of hidden layers and parameters was the same as our GCN. The boxplot in Figure 3.6 demonstrates our GCN outperforming other SOTA classifiers not limited to graph methods on our dataset split.

The results in Table 3.2 highlight comparable metrics of our model versus other studies that operate on 3D voxels from MRI volumes, including the work of Punjabi *et al.* [**107**]. In their work, Punjabi *et al.* train a multi-modal CNN using both volumetric MRI and FDG-PET imaging for the same task, which we outperform while

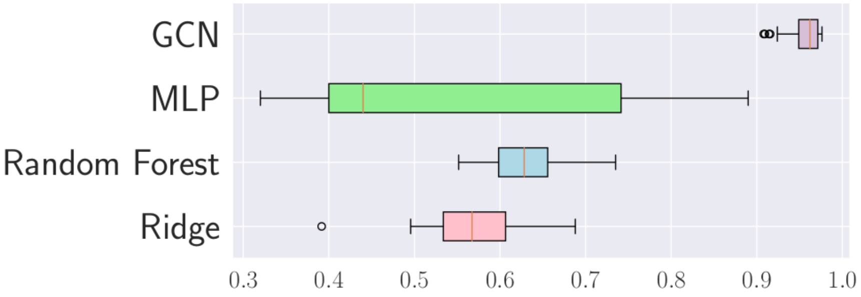


Figure 3.6. Monte Carlo cross-validation accuracy results for GCN and baseline models from [**109**] applied to brain trimeshes.

Table 3.2. Model comparison to classifiers in studies not limited to surface methods. Accuracy, sensitivity, specificity, and the Area Under the Curve (AUC) of the receiver operating characteristic (ROC) curve statistics reported across different studies using subsets of the same ADNI dataset.

Study	**Modality**	**AD/HC**	**Acc.**	**Sens.**	**Spec.**	**ROC-AUC**
[**107**]	MRI	–/– (723)	73.76	–	–	–
[**107**]	MRI+PET$_{\text{amyloid}}$	–/– (723)	92.34	–	–	–
[**104**]	MRI+PET$_{\text{FDG}}$	51/52	94.37	94.71	94.04	**97.24**
[**124**]	MRI+CSF	96/111	91.80	88.50	94.60	95.80
[**125**]	MRI	228/188	84.13	82.45	85.63	90.00
[**126**]	MRI	92/94	93.01	89.13	**96.80**	93.51
[**127**]	MRI	70/70	**97.60**	–	–	–
[**128**]	MRI	200/232	94.74	**95.24**	94.26	–
Ours	MRI	167/265	96.35	92.37	96.74	96.84

training and evaluating on a smaller subset of their subject population. Furthermore, volumetric models like those by Punjabi *et al.* rely on 3D CNNs with far more learned parameters (×2 for a modality fusion model), when compared to sparser approaches likes GCNs. Similar to Ranjan *et al.* [**4**], we also achieve comparable results with far less learnable parameters by working with sparse surface representations like trimeshes and focusing on brain *shape* features instead of raw voxel intensities from MRIs and applying voxel-based classifiers on dense volumes.

3.4.2. Class activation map (CAM) visualization

By employing Grad-CAM on our best GCN, an average CAM was generated for true positive (TP) predictions (Figure 3.7). We project our CAM onto the cortical template [**129**] provided by FreeSurfer [**112**] and the homemade subcortical region templates detailed in [**113**]. The color scale highlights areas from least-to-most influential in TP predictions. The patterns in the CAM match previously described distributions of cortical and subcortical atrophy [**98, 130**]. One reason we may observe a mismatch between the CAM and expected atrophy in the inferior parietal lobule could be the degree of variability in highly folded association cortex, e.g., the intermediate sulcus of Jensen is found only in some individuals [**97, 131**]. The slightly more left-lateralized pattern in the CAM aligns with previous reports that propose greater pathologic burden and neurodegeneration of the language network which leads to worsening on verbal-based neuropsychological measures of memory resulting in a diagnosis for AD [**102**].

3.5. Conclusion and Future Work

In this work, we demonstrated the effectiveness of using cortical and subcortical surface meshes in the context of binary AD clinical diagnosis and ROI visualization in TP predictions. Furthermore, we compared the cross-validation results of our model for the same AD vs. HC problem using other ML models on our data. Additionally,

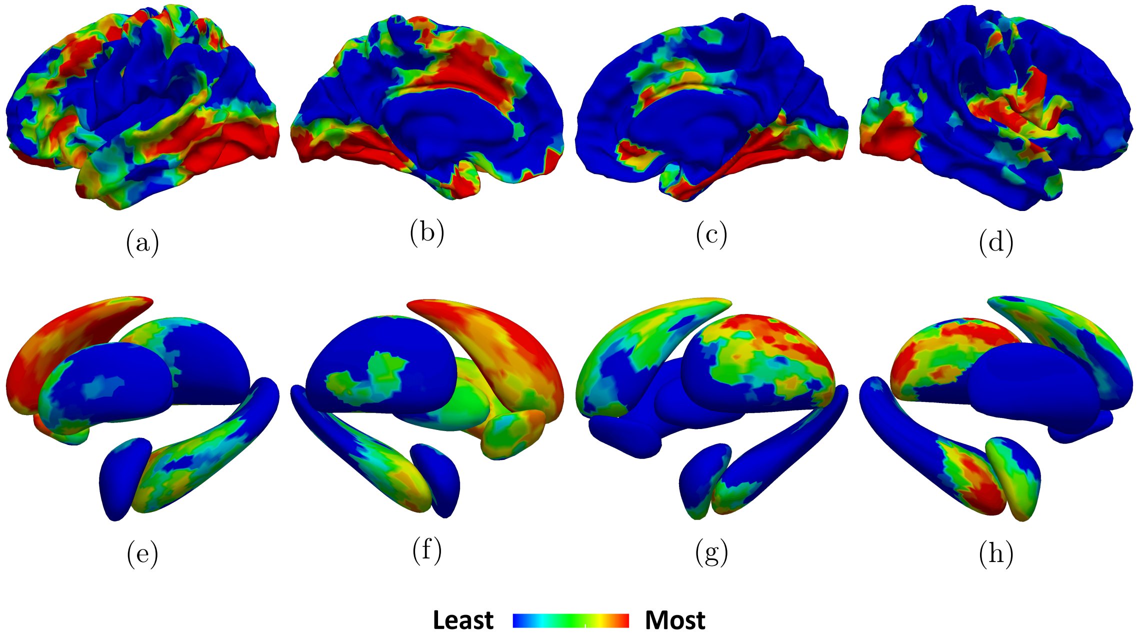


Figure 3.7. Average TP CAMs on the cortical template from [**129, 112**] (top) and subcortical regions from [**113**] (bottom) including: (a-b, e-f) lateral-medial views of the LH respectively, (c-d, g-h) medial-lateral views of the RH respectively.

our final results were comparable to the results of other studies that use traditional neuroimaging modalities as inputs. When compared to the performance of the multi-modal approach used by Punjabi *et al.* [**107**], our model outperforms their approach, thus potentially indicating the reliability of leveraging shape information represented as trimeshes to perform the same binary classification task.

Natural extensions of this work could be to (1) expand our classification problem to include a third class from ADNI, mild cognitive impairment (MCI), (2) increase the population in our study to include those in ADNI3 [**121**], (3) work on longitudinal predictions, and (4) compare our model's performance in using only the cortex, subcortical regions, or both. Additionally, having a 3D-volume-to-mesh dataset offers the potential for developing generative networks [**132**], for performing the graph extraction preprocessing step described in subsection 3.2.2. This will provide more autonomy and limit the need for the manual quality assessment (QA) of trimeshes as a part of our pipeline.

CHAPTER 4

Analyzing Brain Morphology in Alzheimer's Disease Using Discriminative and Generative Spiral Networks

Abstract

Several patterns of atrophy have been identified and strongly related to Alzheimer's disease (AD) pathology and its progression. Morphological changes in brain *shape* have been identified up to ten years before clinical diagnoses of AD, making its early detection more relevant. We propose novel geometric deep learning frameworks for the analysis of brain shape in the context of neurodegeneration caused by AD. Our deep neural networks learn low-dimensional shape descriptors of multiple neuroanatomical regions, instead of using handcrafted features for each region. A discriminative network using spiral convolution on 3D trimeshes is constructed for the *in-vivo* binary classification of AD vs. healthy controls (HCs) using a fast and efficient "spiral" convolution operator on 3D trimesh surfaces of human brain subcortical regions extracted from T1-weighted magnetic resonance imaging (MRI). Our network architecture consists of modular learning blocks using residual connections to improve overall classifier performance.

In this study: (1) a discriminative network is used to analyze the efficacy of disease classification using input data from multiple brain regions and compared

to using a single hemisphere or a single bilateral brain region. It also outperforms prior work using spectral graph convolution on the same the same tasks, as well as alternative methods that operate on intermediate point cloud representations of 3D shapes. (2) Additionally, visual interpretations for regions on the surface of brain regions that are associated to true positive AD predictions are generated and fall in accordance with the current reports on the structural localization of pathological changes associated to AD. (3) A conditional generative network is also implemented to analyze the effects of phenotypic priors given to the model (i.e. AD diagnosis) in generating subcortical regions. The generated surface trimeshes by our model indicate learned morphological differences in the presence of AD that agrees with the current literature on patterns of atrophy associated to AD pathology. In particular, our inference results demonstrate an overall reduction in subcortical trimesh volume and surface area in the presence of AD, especially in the hippocampus. The low-dimensional shape descriptors obtained by our generative model are also evaluated in our discriminative baseline comparisons versus our discriminative network and the alternative shape-based approaches.

4.1. Introduction

Advances in magnetic resonance imaging (MRI) have enabled a plethora of non-invasive shape analysis tools and techniques for modeling the human anatomy in high detail, specifically neuroanatomical shape modeling [**87**]. Methodological insights in human brain shape analyses have demonstrated powerful utility for their visualization capabilities and valued characterizations of neuropathology and neurodevelopment. Shape-based descriptors have proven to be effective for a variety of tasks such as: segmentation, observing and identifying shape asymmetries, and surface analyses using triangular meshes (trimeshes) [**133**]. Morphological patterns of change in brain regions have often been predictive of different neurodevelopmental and neurodegenerative diseases, such as: schizophrenia, epilepsy [**134**], Lewy bodies, and Alzheimer's disease (AD) [**135**]. Neuroanatomical changes in structural MRI have been identified up to ten years before clinical diagnoses in AD [**136**]. Wachinger *et al.* [**78**] employ BrainPrint to yield extensive characterizations of brain anatomy using region-specific shape descriptors with samples from the Alzheimer's Disease Neuroimaging Initiative (ADNI) dataset [**83**] to identify unique individuals (3,000 subjects) with a 99.8% accuracy. Gutiérrez-Becker *et al.* [**89**] demonstrate a strong performance (0.80/0.79/0.78 for precision/recall/F1-score respectively) using BrainPrint to classify scans belonging to AD subjects apart from healthy controls (HCs), in which they later outperform in a baseline comparison using their own shape descriptors (0.83/0.84/0.82 precision/recall/F1-score respectively) learned on point cloud representations of neuroanatomical shapes.

Working with geometric shape descriptors offers a more robust representation of brain morphology, rather than direct image intensities. The inferences drawn from utilizing shape descriptors are able to remain robust with respect to intensity changes that may be caused by differing scanner hardware/protocols. A recent development in deep learning (DL), PointNet [**137**], introduces artificial neural networks (NNs) designed for operating on 3D point clouds in tasks such as object identification. Gutiérrez-Becker *et al.* [**89**] utilize the point cloud operations from PointNet [**137**] to construct deep NNs which are trained for AD vs. HC classification on unordered 3D point cloud representations of subcortical brain regions. Their framework is also evaluated on the mild cognitive impairment (MCI) vs. HC classification task, which yields a significant drop in classifier performance due to the high variability within the MCI class, since the detection of MCI is more symptomatic and it is sub-divided into different stages (typically early MCI and late MCI).

Generalizations of successful convolutional neural networks (CNNs) to non-Euclidean data types, such as point clouds and trimeshes, fall under the wide umbrella of geometric deep learning (GDL) [**138**]. Similar to 3D voxels [**139**], point clouds [**140**] are intermediate representations of 3D shapes, unlike direct surface representations, i.e. a triangular mesh (trimesh). Despite their high success, voxel-based DL approaches typically suffer from high computational complexity, and point cloud approaches suffer from an absence of smoothness in data representation [**141**]. Polygon meshes are direct and effective surface representations of object *shape*, when compared to voxels. Geometric learning on meshes has only recently been explored [**142, 143, 4, 144, 145**] for shape completion, non-linear facial morphable model generation and classification, 3D surface segmentation, and reconstruction from 2D/3D images. A novel representation learning and generative DL framework using spiral convolution on fixed topology meshes, was established with Neural3DMM [**141**] and later improved upon with SpiralNet++ [**146**].

Given the relevance and valued characterizations of brain shape in neuropathology and neurodevelopment, as well as the added value of successful DL methods for shape-driven tasks using 3D point clouds [**137, 147**], we improve upon the work by Gutiérrez-Becker *et al.* [**89**], which operates on unordered point clouds of 3D brain regions. We extend their discriminative networks by working with spiral convolution operators on trimeshes instead. Similar to Gutiérrez-Becker *et al.* [**89**], we use a *conditional* generative network framework to introduce non-imaging data, particularly AD diagnosis, and analyze the learned morphological patterns of generated trimeshes with respect to diagnostic priors.

Our framework is based upon the spiral convolution operators defined in SpiralNet++ [**146**] and the residual NN framework for image recognition established by He *et al.* [**114**]. We quantitatively evaluate the performance of our model in AD/MCI binary classification with an ablation study using different subcortical region inputs (all regions, per-hemisphere, and bilaterally per-region) to analyze the efficacy of incorporating input data from multiple brain regions. Furthermore, we perform a baseline comparison of our spiral framework's performance with our prior work [**96**] using spectral graph convolution [**32**], and the point cloud approach by Gutiérrez-Becker *et al.* [**89**] on the same AD/MCI classification tasks. Our generative model is based upon a conditional variational autoencoder (CVAE) [**148**] framework, which is used to extract low-dimensional brain shape descriptors that are then used for the same AD/MCI binary classification tasks when compared to HCs. We also experiment with the effects of conditioning our generative model on AD diagnosis during training and trimesh generation (synthesis).

An interpretation of classifier *reasoning* is often a desired quality of DL frameworks that is often neglected but highly needed, especially in medical image analyses. This paper is an extension of our preliminary work [**96**] where spectral graph convolutional networks (GCNs) [**32**] were used for binary AD classification and we adapted Grad-CAM [**120**] on trimeshes to provide visually interpretable heatmaps that localize areas on trimeshes which drive true positive (TP) AD predictions. Given Grad-CAM's modularity to work with any CNN model, we apply a trimesh adaptation of Grad-CAM [**96**] to the discriminative network in this study.

In summary our contributions are as follows:

(1) **A joint framework for improved *in-vivo* pathological classification using multiple subcortical regions in a single scan**. A holistic view of brain subcortical anatomy is provided to demonstrate stronger discriminative performance with multiple brain regions. For AD in particular [**149, 150**], correspondences across multiple regions are often more robust than studying one organ in isolation, especially in neuroimaging where segmenting multiple subcortical regions is possible from a single MRI volume. AD has also been identified to start in localized brain regions (good for early detection) and has been shown to progressively spread to multiple brain regions (good for robust detection).

(2) **Discriminative SpiralNets for improved AD classification on trimeshes versus prior work using spectral method**. We demonstrate an improvement in accuracy, precision, recall, and F1-score upon our prior work [**96**] by using spiral convolution on brain surface trimeshes for AD classification. Our discriminative SpiralNet also outperforms alternative shape-based descriptor approaches which operate on intermediate shape representations such as point clouds.

(3) **Mesh Grad-CAM adaptation to provide visual reasoning in localized region of interest (ROI) on trimesh manifolds that drive TP predictions in AD classification**. Our prior adaptation of Grad-CAM [**96**] was successful in localizing ROIs on trimeshes for TP predictions from our GCNs. Although a different convolution operator is used in this proposed framework, learned feature maps are still attainable from convolutional layers for generating class activation maps (CAMs) onto input trimesh surfaces. These CAMs are a visual interpretation of regions along the surface of subcortical regions whose shape is indicative of TP AD predictions by our spiral networks. Our obtained CAMs draw direct correspondences with brain shape deformations tightly correlated with AD pathology.

(4) **Conditional generative spiral networks for low-dimensional representation learning on brain trimesh manifolds with diagnostic priors**. Our generative CVAEs are able to learn low-dimensional discriminative representations of trimeshes, which are then evaluated against our

proposed discriminative network and prior baseline methods. The morphological effects of conditioning on AD are also analyzed and supported by multiple reports on the neuroanatomical changes in AD progression.

4.2. Related Work

4.2.1. PointNet on 3D neuroanatomical surfaces

As an improvement upon *BrainPrint* [**85, 86**], Gutiérrez-Becker *et al.* [**89**] introduced a DL approach for the shape analysis of neuroanatomical regions in AD using point clouds distributed on the surface of each region. Point clouds are a lightweight representation of 3D surfaces which avoids topological constraints of surfaces and can be trivial to roughly obtain given a segmented surface. Although computationally robust, their method still operates on and outputs *intermediate representations* of brain shape.

Methods that generate intermediate representations of 3D surfaces (i.e. pixels) are left insensitive to the physical constraints of the boundaries for a 3D object. The output quality of postprocessing steps taken to generate 3D surfaces, like trimeshes, therefore become dependent on the output quality of the intermediate representations [**151**]. In this work, we improve upon the framework established by Gutiérrez-Becker *et al.* [**89**] by working with spiral convolution operators that operate directly on 3D morphable trimeshes [**141**] that are registered to a common template topology. We also improve upon their framework by way of residual connections [**114**] within our classifier, and demonstrate an improvement in classification performance using residual connections within alternative approaches in our baseline classifier comparison.

Additionally, Gutiérrez-Becker *et al.* [**89**] demonstrate a powerful framework for fixed-size point cloud reconstruction and generation using a *PointNet* [**137, 147**] CVAE architecture. Although point cloud methods can be compact and robust, they can still lack an underlying smooth structure whose approximation is dependent on the output quality of the close, whereas surface trimeshes are more realistic, less sensitive to noise, and are capable of preserving high-quality 3D geometry generation. In this work we construct CVAEs using fixed-size trimeshes which are registered to a common template during preprocessing.

4.2.2. Spectral graph convolution (ChebyNets)

Morphable meshes [**141**], specifically trimeshes [**152**], are direct surface representations of 3D volume boundaries that can be used for 3D visualization, describing 3D texture, and contextualizing *shape*. By construction, trimeshes are undirected graphs. Modeling convolution on 3D trimeshes can be more memory efficient and allow for the processing of higher resolution 3D structures when compared to volumetric approaches using 3D CNNs. Our prior work [**96**] demonstrates an improvement in AD classification with spectral GCNs referred to as ChebyNets [**3**], using a dataset of 3D trimeshes from a subset of T1-weighted MRIss in the subject population used by Punjabi *et al.* [**107**], a volumetric approach. ChebyNets are also implemented and utilized by Ranjan *et al.* for a generative framework using a convolutional mesh autoencoder (CoMA) for generating 3D human faces.

Spectral filtering on graphs [**3, 32**] can come with a number of caveats. Spectral filters are inherently *isotropic* by design since they particularly rely on the graph Laplacian $\mathbf{L}$ or adjacency matrix $\mathbf{A}$ as the graph shift operator, each performing weighted averages (aggregations) of adjacent vertices with respect to each vertex, regardless of order/locality. Gong *et al.* [**146**] emphasize that the isotropic nature of spectral filters for undirected graphs is a side effect of needing to design a permutation-invariant operator with a small number of parameters, under the *absence* of canonical ordering.

While a "necessary evil" for certain graph learning applications [**141**], spectral graph filters are still basis-dependent and can be rather weak on trimeshes since they are locally rotational-invariant. On the other hand, spiral convolutional filters take advantage of the fact that trimeshes are locally Euclidean and a canonical ordering of neighbors for each vertex can be established, such as a spiral ordering starting an arbitrary vertex. By design, spiral filters are *anisotropic* and have proven to generalize functions on 3D meshes better than spectral methods, as demonstrated by Bouritsas *et al.* [**141**] and Gong *et al.* [**146**]. In our analyses, an ablation study demonstrates an improvement upon AD classification performance using spiral filters, in comparison to spectral filters; originally used in our preliminary work in chapter 3 [**96**].

4.2.3. Generative networks on brain graphs

Several studies have recently investigated using GDL [**138**] for synthesizing brain-related graphs [**153, 154, 155, 156**] using generative adversarial network (GAN) [**132**] inspired frameworks. Other types of generative networks, namely autoencoder-based architectures, have also demonstrated success for neuroimaging applications, such as the work of Choi *et al.* [**157**], where a generative model is developed using chronological age and apoE4 genetic traits as conditional features for synthesizing PET scans in the context of AD and brain aging. In their study, a variational autoencoder (VAE) [**158**] is used to demonstrate the significant effect apoE4 genetic information had in predicting age-related metabolic changes in synthesized PET scans that are then compared to ground-truth follow-up scans.

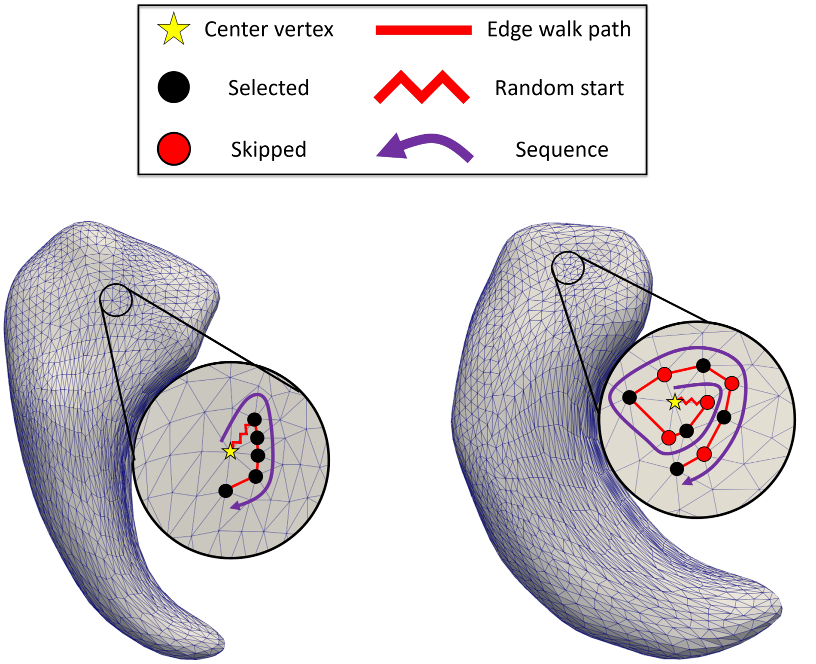
Autoencoders are NNs trained to minimize the reconstruct error between their inputs and outputs, separated by *encoder* and *decoder* halves. Traditionally, autoencoders have been used for unsupervised dimensionality reduction or feature learning, since their objective functions for training are typically designed to minimize the reconstructions of its inputs (i.e. mean absolute error).

Variational autoencoders (VAEs) [**158**], similarly aim to reconstruct inputs, while also attempting to constrain the latent space of the encoder output to an assumed underlying probabilistic distribution (such as a multivariate Gaussian). Using this assumption, the total objective function used to train VAEs minimize a reconstruction loss term and a latent space regularization term, typically the Kullback–Leibler (KL) divergence [**159**], as a measure of the disparity between the embedding and assumed prior distribution $\mathcal{N}(\mathbf{0}, \mathbf{I})$. Once trained, VAEs are valuable in their utility as a generative framework, where new samples can be synthesized by sampling from the assumed prior distribution. CoMA [**4**] is built upon a VAE framework for meshes, using spectral GCNs. Their results demonstrate remarkable performance in synthesizing a diversity of facial expressions on 3D morphable meshes, all registered to a common template topology.

As a generative framework, one drawback to VAEs is the lack of control in *targeted* data generation. This can be problematic for tasks dependent on generating specific *types* of samples. *Conditional variational autoencoders (CVAEs)* [**148**] offer more control by combining variational inference from VAEs with additional conditional priors, with respect to each sample, using a simple concatenation step prior to decoding. Based on CoMA and the success of point cloud generation for neuroanatomical shapes [**89**], a CVAE framework composed of spiral convolutional learning blocks is used in this study to generate 3D trimeshes of neuroanatomical regions by conditioning on AD diagnosis.

4.3. Methods

The mesh notation from Table 1.1 and trimesh extraction preprocessing steps described previously in subsection 3.2.2 of chapter 3 are used again in the methods of this study to deal with trimeshes. However, due to computational limitations in dealing with incorporating the cortical surfaces, this study only analyzes and predicts on the subcortical regions (i.e. amygdala, caudate, etc.). Additionally, the trimesh adaptation of Grad-CAM [**120**] discussed in subsection 3.2.5 is adapted to



(a) Spiral on AD mesh (b) Dilated spiral on HC mesh

Figure 4.1. Examples of clockwise spiral sequences established on left hippocampi trimeshes from randomly selected scans of a subject with Alzheimer's disease (left) and a healthy control (right). Note that in using dilation, the receptive field of the kernel supports exponential expansion without increasing the support-size/length of the spiral kernel [**146**]. In each example, a spiral sequence of 6 selected vertices are generated including the center vertex.

this study's discriminative framework, given Grad-CAM's modularity for any type of CNN architecture.

4.3.1. Spiral sequences on trimeshes

First, we provide an illustrated clarification of spiral sequences on 3D morphable trimeshes of brain regions (Figure 4.1), which are at the core of the learning framework introduced by Gong *et al.* [**146**]. Given an arbitrary trimesh and an arbitrarily-selected vertex we'll refer to as the *center vertex*, a spiral sequence can be naturally enumerated by following a spiral pattern around the center vertex. First, a spiral orientation is fixed (clockwise or counter-clockwise) and a random starting direction is selected from the center vertex. Following the convention of Gong *et al.* [**146**], orientations for all spiral generations were fixed to *counter-clockwise* in our analyses and a *random* starting direction was initialized for each vertex.

Specifically, a k-ring and a k-disk around a center vertex v is defined as:

$$\begin{aligned}
0 - \text{ring}(v) &= \{v\}\,, \\
k - \text{disk}(v) &= \bigcup_{i=0,\dots,k} i\text{-ring}(v), \\
(k+1)\text{-ring}(v) &= \mathcal{N}\left(k\text{-ring}(v)\right) \setminus k\text{-disk}(v), \qquad (4.1)
\end{aligned}$$

where $\mathcal{N}(V)$ is the neighborhood/set of all vertices adjacent to any vertex in the set of vertices V. A spiral sequence of length L at vertex v is defined as $S(v, L)$; a *canonically ordered set* of L vertices from a concatenation of k rings. Only part of the last ring is concatenated in this definition, in order to ensure a fixed-length serialization. Formally, the spiral sequence is defined as:

$$S(v, L) \subset \{0\text{-ring}(v), 1\text{-ring}(v), \dots, k\text{-ring}(v)\}\,. \qquad (4.2)$$

The spiral sequences defined in SpiralNet++ [**146**] show remarkable advantages to a high-level feature representation of each vertex in a consistent and robust way when spirals are *frozen* during training. By this we mean that spiral sequences are generated *only once* for each vertex on the template trimesh, instead of randomly generated sequences for every vertex per epoch, as was done by Lim *et al.* [**160**].

4.3.2. Spiral convolution

The idea of a convolutional neural network (CNN) applied on 2D/3D images defined on standard Euclidean grids [**33**] is dependent on using 2D/3D rectangular convolutional filters that slide across images and map $F_{\text{in}} \mapsto F_{\text{out}}$ features with respect to every pixel/voxel, as discussed in Appendix B of the chapter 6.5. An extension of this application on data types in irregular domains such as graphs, is typically expressed using [**161, 162**] or message passing [**163**] schemes.

Using the mesh notation convention established in Table 1.1, with $\mathbf{x}_i^{(k-1)} \in \mathbb{R}^{F_{\text{in}}}$ denoting the feature vector of F_{in} features at vertex v_i and $\mathbf{w}_{ij}$ denoting the (optional) E_{in} features on edge eij connecting vertex v_i and vertex v_j at layer $(k-1)$, message passing NNs are typically defined such that

$$\mathbf{x}_i^{(k)} = \gamma^{(k)}\left(\mathbf{x}_i^{(k-1)}, \mathop{\square}_{j\in\mathcal{N}(i)} \phi^{(k)}\left(\mathbf{x}_i^{(k-1)}, \mathbf{x}_j^{(k-1)}, \mathbf{e}_{ij}^{(k-1)}\right)\right), \tag{4.3}$$

where $\square$ represents a differentiable permutation-invariant operation (i.e. sum, mean, max, etc.), and $\gamma^{(k)}$ and $\phi^{(k)}$ denote differentiable kernel functions such as a multilayer perceptron (MLP). CNNs defined for data types that exist in standard Euclidean grids have a clear one-to-one mapping. However for data types in irregular domains such as graphs, correspondences are defined using neighborhood connectivity for each vertex and weight matrices dependent on the kernel functions, γ and ϕ at each layer.

Using the spiral sequence serialization discussed in the previous section (subsection 4.3.1), we can define convolution on trimeshes in a *canonical* manner to Euclidean CNNs that is *anisotropic* by design. Following the convention of Gong *et al.* [**146**], the spiral convolution operator is defined as

$$\mathbf{x}_i^{(k)} = \gamma^{(k)} \left(\underset{v_j \in S(v_i, L)}{||} \right), \tag{4.4}$$

where γ denotes a MLP and $||$ is the concatenation operation applied on the shared features of the vertices of spiral sequence $S(v_i, L)$ centered at vertex v_i.

The *dilated* extension [**164**] of spiral convolution using a dilated spiral sequence (depicted in Figure 4.1) can also be applied to trimeshes by uniformly sub-sampling during spiral generation, with the pre-processing parameter d denoting a uniform sampling of every $d-1$ vertices along the spiral sequence until L vertices are selected.

4.3.3. Trimesh sampling (up/down-sampling)

Traditional Euclidean CNNs typically use a hierarchical multiscale learning structure, typically employed for learning global and local context, using a combination of convolutional layers and pooling/up-sampling operations in-network. To mimic this behavior, we use trimesh sampling/coarsening operators introduced by Ranjan *et al.* [**4**] that define analogous down-sampling and up-sampling of trimesh vertices within a NN.

The 3D trimesh samples in this study use $F = 3$ input features per vertex, each feature corresponding to the x, y, and z coordinates for every vertex in a subject's native 3D space. However, convolutions applied on trimesh features within a NN can result in features with different dimensionality. Therefore, in this section we use F to generalize our explanation to trimeshes with arbitrary feature dimensions across their vertices.

The in-network down-sampling of a trimesh, with N vertices, is performed using the down-sampling matrix, $\mathbf{D} \in \{0,1\}^{M \times N}$, and up-sampling with $\mathbf{U} \in \mathbb{R}^{N \times M}$, for $N > M$. The sparse down-sampling matrix is obtained by contracting vertex pairs iteratively that maintain surface error approximations using quadric matrices [**5**]. The vertices of the down-sampled trimesh are essentially a subset of the original trimesh's vertices, $\mathcal{V}_d \subset \mathcal{V}$. Each element of $(\mathbf{D})_{p,q} \in \{0,1\}$ denotes whether the q^{th} vertex is kept during down-sampling with $(\mathbf{D})_{p,q} = 1$, otherwise discarded with $(\mathbf{D})_{p,q} = 0 \ \forall p$.

To remain loss-less, the up-sampling operator is built during the generation of the down-sampling operator. Vertices retained during down-sampling are kept for up-sampling such that $(\mathbf{U})_{q,p} = 1$ iff $(\mathbf{D})_{p,q} = 1$. Vertices $v_q \in \mathcal{V}$ that are discarded during down-sampling, for $(\mathbf{D})_{p,q} = 0, \forall p$, are mapped into the down-sampled trimesh surface by using barycentric coordinates, $\tilde{p} = w_i i + w_j j + w_k k$, where $v_{i,j,k} \in \mathcal{V}$ and $w_i + w_j + w_k = 1$. Using these weights, we update $\mathbf{U}$ such that

$$\mathbf{U}(q,i) = w_i, \quad \mathbf{U}(q,j) = w_j, \quad \mathbf{U}(q,k) = w_k,$$

otherwise, $\mathbf{U}(q,l) = 0$.

The features on the vertices retained from down-sampling a trimesh are obtained via the sparse matrix multiplication

$$\mathbf{Y} = \mathbf{D}\mathbf{X} \in \mathbb{R}^{M \times F}, \tag{4.5}$$

for $\mathbf{X} \in \mathbb{R}^{N \times F}$. By analogy, the vertices on the output trimesh of an up-sampling operation are obtained as an inverse to down-sampling via the sparse matrix multiplication

$$\mathbf{X} = \mathbf{U}\mathbf{Y} \in \mathbb{R}^{N \times F}. \tag{4.6}$$

4.3.4. Spiral brain trimesh networks

4.3.4.1. Residual learning blocks (ResBlocks). Motivated by the success of residual DL frameworks [**114**] for image recognition, the NNs used in this study are also composed of residual learning blocks (ResBlocks) depicted in Figure 4.2. A spiral convolutional layer maps $F_{\text{in}} \mapsto F_{\text{out}}$ features for every vertex in the input trimesh using MLPs applied on the spiral sequence, $S(v_i, L)$ of each vertex v_i. Analogous to traditional convolution with padding to preserve the size of input feature maps, spiral convolution on trimeshes also preserves dimensionality since $S(v_i, L)$ is defined for every input vertex. Therefore, the number of vertices, N, is still preserved in the output vertex feature matrix, $\mathbf{X}_{\text{out}} \in \mathbb{R}^{N \times F_{\text{out}}}$. Through trial-and-error, we found that using the ResBlock sequence defined in Figure 4.2 yielded the best results,

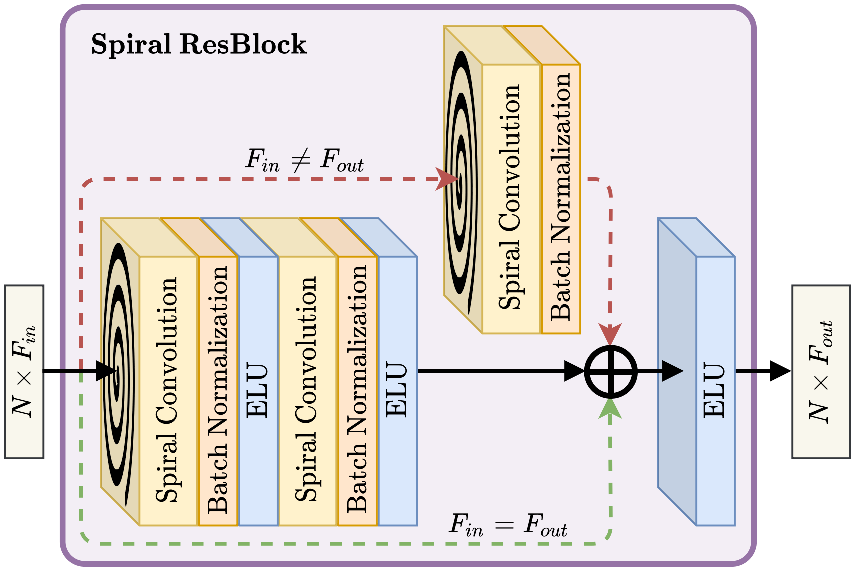


Figure 4.2. Residual learning block (ResBlock) module used in SpiralNet inspired architecture. Batch normalization (BN) (depicted in orange) is applied after spiral convolution (depicted in yellow). The top (red) branch of the ResBlock uses spiral convolution followed by BN as an identity linear mapping tool to map the F_{in} features of the input vertices to the F_{out} features acquired by the main branch. Otherwise, the input of the residual learning block (ResBlock) is added to the main branch output (green). An element-wise ELU$(\cdot)$ function is used within the hidden layers and as the final activation of the ResBlock.

which still incorporates batch normalization (BN), GCN layers (this time spiral), and non-linear activation functions.

An important hyperparameter for training deep NNs is the choice of activation function for the hidden layers and output layer. In their work, He *et al.* [**114**] used the rectified linear unit (ReLU) activation function defined as

$$\mathrm{ReLU}(x) = x^{+} = \max\left(0, x\right),$$

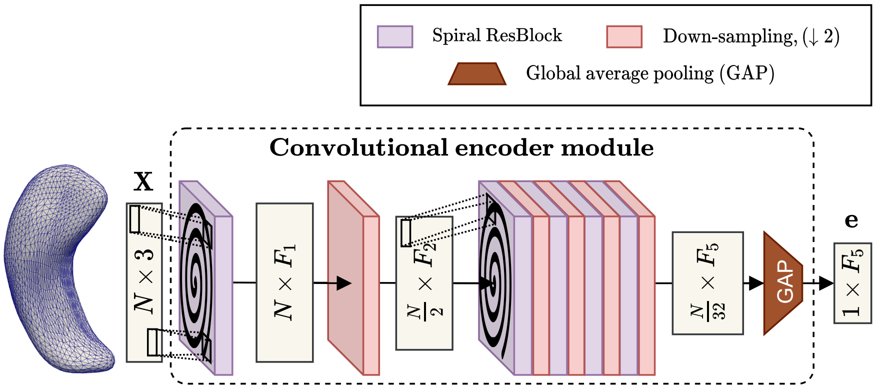


Figure 4.3. Trimesh mesh encoder module made up of a sequential stack of alternating spiral convolution and down-sampling layers (five each). The i^{th} ResBlock maps F_i features onto the vertices of the respective input. Each down-sampling layer coarsens the input vertex count down by a factor of two. After the final down-sampling layer, global average pooling (GAP) is applied over to reduce the output embedding down to $\mathbb{R}^{F_5}$.

within their residual learning framework. DL architectures using ReLU are prone to suffering from the common "dying ReLU" problem where hidden layer outputs heavily saturate to zero [**165**], leading to zero-valued gradients, thus making learning more difficult. We circumvent this by using the exponential linear unit (ELU) activation function [**166**] instead, defined as

$$\text{ELU}(x) = \begin{cases} x, & \text{if } x > 0 \\ \alpha\left(e^{x} - 1\right), & \text{if } x \leq 0 \end{cases}, \tag{4.7}$$

for $\alpha = 1$ in this study.

4.3.4.2. Convolutional mesh encoder. Using the ResBlocks introduced in subsubsection 4.3.4.1 and trimesh down-sampling, described in subsection 4.3.3, we introduce the convolutional *encoder* module used by our discriminative and generative SpiralNets, illustrated by Figure 4.3. As illustrated, the input feature matrices are embedded to $\mathbb{R}^{F_5}$ latent vectors using the encoder defined as the sequential stack:

$$\begin{aligned}&\{\mathrm{ResBlock}(L_1, d_1, F_1) \to \mathrm{MS}(\downarrow 2) \to \mathrm{ResBlock}(L_2, d_2, F_2) \to \\ &\mathrm{MS}(\downarrow 2) \to \cdots \to \mathrm{ResBlock}(L_5, d_5, F_5) \to \mathrm{MS}(\downarrow 2) \to \mathrm{GAP}_N\} \,,\end{aligned}$$

where

- L_r, d_r, and F_r refer to the spiral lengths, dilation, and number of filters for all convolutional layers in the r^{th} ResBlock,
- MS($\downarrow$ 2) is shorthand for "mesh sampling (MS) down by a factor of 2" (down-sampling),
- and GAP_N is the global average pooling operation [**4**] over N vertices.

On meshes, GAP is essentially just an averaging operation over the node dimension, as depicted in Figure 4.3.

Note that since the input trimesh is down-sampled five times within the module, each time by a factor of 2, the number of vertices after the final down-sampling operation is $\frac{N}{2^5} = \frac{N}{32}$. This module is used as the first step for both our discriminative and generative networks, described in subsubsection 4.3.4.4 and subsubsection 4.3.4.5.

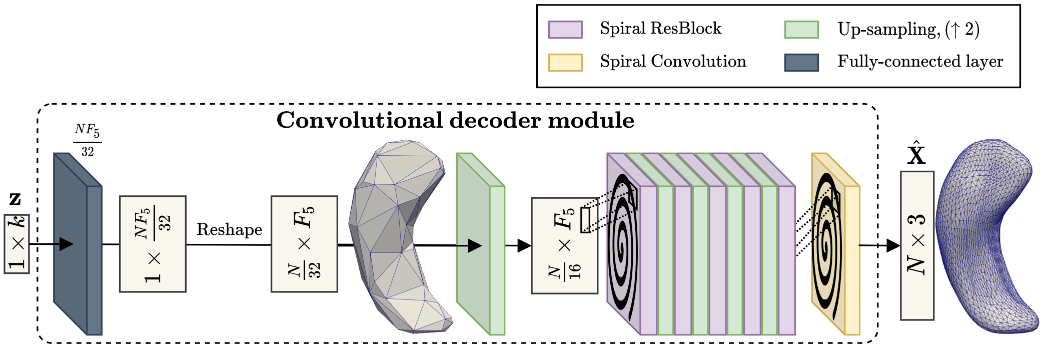


Figure 4.4. The mesh decoder module first uses a MLP layer and reshaping to map the input vector, $\mathbf{z} \in \mathbb{R}^k$ to a feature matrix for trimeshes at the coarsest level in $\mathbb{R}^{\frac{N}{32} \times F_5}$. Alternating up-sampling and ResBlock layers (five each) are used after. An additional spiral convolutional layer with three filters and no activation function is used to project the penultimate $N \times F_1$ feature matrix back to $N \times 3$ for the respective 3D trimesh reconstruction.

4.3.4.3. Convolutional mesh decoder. The convolutional trimesh *decoder* module, depicted in Figure 4.4, applies a synonymous backwards transformation of the encoder module described in subsubsection 4.3.4.2. Following Figure 4.4 and starting with an arbitrary vector $\mathbf{z} \in \mathbb{R}^k$, first a MLP maps $\mathbf{z} \mapsto \mathbb{R}^{\frac{NF_5}{32}}$. This output is then reshaped to get a feature matrix in $\in \mathbb{R}^{\frac{N}{32} \times F_5}$, representing the F_5 features on the $\frac{N}{32}$ vertices at the coarsest level of our trimeshes. The rest of the decoder module is defined as the sequential stack:

$$\begin{aligned}&\{\mathrm{MS}(\uparrow 2) \to \mathrm{ResBlock}(L_5, d_5, F_5) \to \mathrm{MS}(\downarrow 2) \to \mathrm{ResBlock}(L_4, d_4, F_4) \to \\ &\mathrm{MS}(\uparrow 2) \to \cdots \to \mathrm{ResBlock}(L_1, d_1, F_1) \to \mathrm{SpiralConv}(l_1, d_1, 3)\}\,,\end{aligned}$$

where L_r, d_r, and F_r are the *same* corresponding values used in the encoder module from subsubsection 4.3.4.2. An additional spiral convolution (SpiralConv) with 3

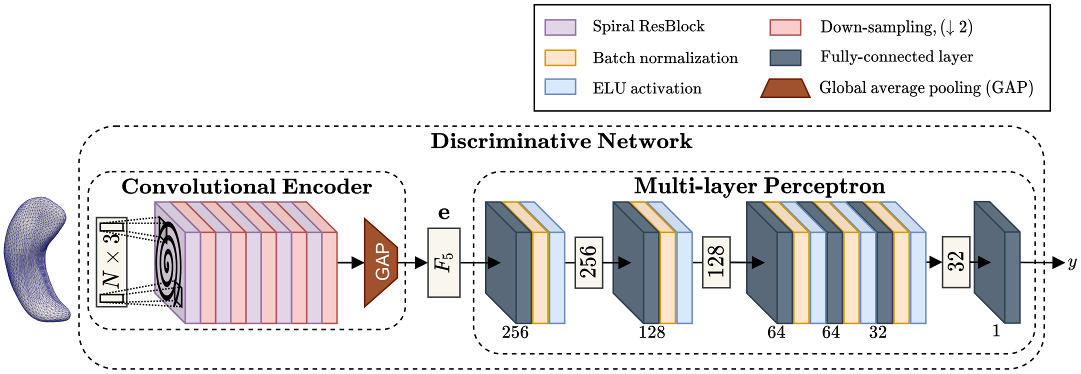


Figure 4.5. End-to-end discriminative spiral network given a 3D trimesh input with feature matrix $\mathbf{X} \in \mathbb{R}^{N\times3}$. Batch normalization (BN) is used after each MLP layer, followed by an ELU$(\cdot)$ activation. Given the output of the convolutional encoder, $\mathbf{e} \in \mathbb{R}^{F_5}$, the MLP predicts the target, y, from the embedding for a particular sample. For binary classification, we apply a sigmoid function after the final layer to output a probability for each sample.

filters is used at the end of the decoder module (with no activation) to obtain the reconstruction: $\hat{\mathbf{X}} \in \mathbb{R}^{N\times3}$, with three features per vertex (corresponding x, y, and z coordinates). This module is only utilized within the generative network described in subsubsection 4.3.4.5, where the task is to output 3D trimeshes.

4.3.4.4. Discriminative network. Following the point cloud discriminative network convention outlined by Gutiérrez-Becker *et al.* [**89**], we construct our own discriminative networks using the encoder module (Figure 4.3) in series with a MLP that uses BN and ELU element-wise activation functions after each MLP layer, as depicted in Figure 4.5. The goal of this network is to learn trimesh features given an input feature matrix, $\mathbf{X} \in \mathbb{R}^{N\times3}$, and a spiral convolutional operator that exploits the locally-Euclidean topology of 3D trimesh manifolds. Global average pooling (GAP) is then applied on the learned trimesh features, and later used within a MLP for predicting the target variable, y.

In this study, we use the discriminative network for binary classification, therefore we apply a sigmoid function, $\sigma(y) = \frac{1}{1+e^{-y}}$, on the predicted targets to get the probability of a positive label given the corresponding 3D trimesh manifold. Traditionally, for binary classification tasks such as disease prediction, the positive binary label, (1), pertaining to the pathological class, is typically the class in opposition to the healthy control (HC) label, (0). Our discriminative network can be trained in an end-to-end supervised manner by optimizing a standard binary cross-entropy (BCE) loss

$$\mathcal{L}_{\text{BCE}} = -\frac{1}{B}\left(y_i \log\left(\hat{y}_i\right) + \left(1 - y_i\right)\log\left(1 - \hat{y}_i\right)\right), \tag{4.8}$$

where y_i and $\hat{y}_i$ are the ground-truth labels and predicted probabilities (output of sigmoid) respectively for a given sample i in a batch of B samples.

4.3.4.5. Generative network (CVAE). Based on the CoMA architecture by Ranjan *et al.* [**4**], our CVAE model uses a convolutional decoder on trimesh samples that share a topology at different hierarchical levels of coarsening, described in subsection 4.3.3. Following Figure 4.6, first a convolutional encoder, E, is used to compress input samples, $\mathbf{X} \in \mathbb{R}^{N\times 3}$, down to the latent vectors, $\mathbf{e} = E(\mathbf{X}) \in \mathbb{R}^{F_5}$. Next, $\mathbf{e}$ is mapped to a "mean vector," $\mu \in \mathbb{R}^k$ and a "standard deviation vector," $\sigma \in \mathbb{R}^k$, using two parallel MLP layers. These vector outputs are then used for variational inference during training with the "reparameterization trick" for VAEs

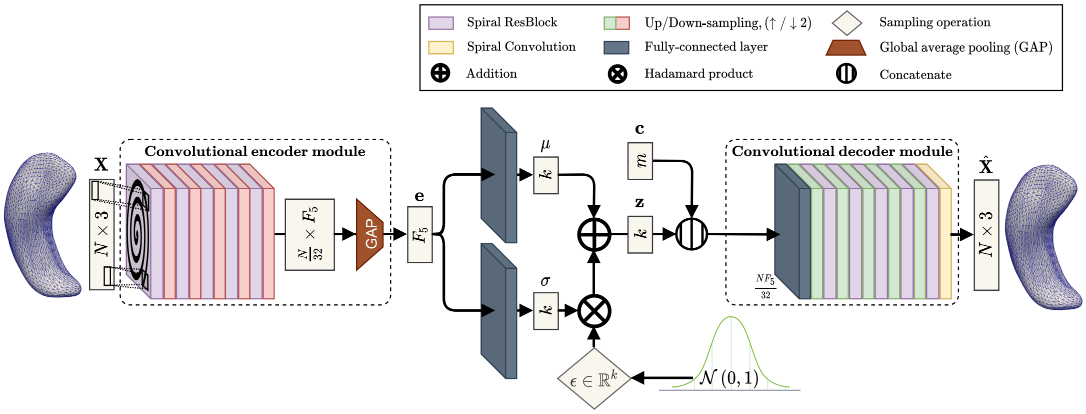


Figure 4.6. End-to-end generative model based on spiral convolutional CVAE architecture. During inference, a trimesh sample, $\mathbf{X} \in \mathbb{R}^{N\times 3}$, is first encoded to $\mathbf{e} \in \mathbb{R}^{F_5}$, using the encoder, E. This encoding is then used to sample, $\mathbf{z} \in \mathbb{R}^{k}$, from the prior distribution, $Q\left(\mathbf{z}|\mathbf{X}\right)$, assumed to be a multivariate Gaussian. Next $\mathbf{z}$ is concatenated with the conditional vector, $\mathbf{c} \in \mathbb{R}^{m}$, and a reconstruction is generated using the decoder $D\left([\mathbf{z}, \mathbf{c}]\right) = \hat{\mathbf{X}} \in \mathbb{R}^{N\times 3}$. During generation, we sample from $\mathcal{N}\left(0, 1\right)$ for each varied component of $\mathbf{z}$, concatenate the sample with a given conditional $\mathbf{c}$, and start at the decoder to generate a new trimesh sample, $D\left([\mathbf{z}, \mathbf{c}]\right)$.

[**158**]. Taking these, we vary each component of the latent vector as $z_i = \mu_i + \epsilon\sigma_i$, for $i = 1, 2, \ldots, k$ and $\epsilon \sim \mathcal{N}\left(0, 1\right)$, therefore assuming a multivariate Gaussian distribution, $Q\left(\mathbf{z}|\mathbf{X}\right)$, from which we can sample from.

Next we obtain and concatenate a random sample from our multivariate Gaussian, $\mathbf{z}$, with the associated conditional vector $\mathbf{c}$, to generate the trimesh reconstruction, $\hat{\mathbf{X}} = D\left([\mathbf{z}, \mathbf{c}]\right)$. As done by Ranjan *et al.* [**4**] for CoMA, our SpiralResNet CVAE is trained by minimizing the loss

$$\mathcal{L}_{\text{gen}} = \left\|\mathbf{X} - \hat{\mathbf{X}}\right\|_1 + w_{\text{KL}}\text{KL}\left(\mathcal{N}\left(\mathbf{0}, \mathbf{I}\right) || \, Q\left(\mathbf{z}|\mathbf{X}\right)\right), \tag{4.9}$$

with $w_{\mathrm{KL}} = 1 \times 10^{-3}$, selected ad-hoc, acting as weight on the KL divergence term of the total loss function $\mathcal{L}_{\mathrm{gen}}$. The first term (reconstruction) minimizes the mean absolute error (MAE) between the obtained reconstruction and ground-truth sample, and the second term (KL divergence) acts a regularizer on the latent space by adding the constraint of a unit Gaussian prior with zero-mean on the latent space distribution, $Q\left(\mathbf{z}|\mathbf{X}\right)$.

Once trained, synthesizing new samples is simple. Since the KL divergence term of the loss function attempts to enforce/constrain the latent space to a unit Gaussian, we generate new samples with our decoder by sampling $\mathbb{R}^k$ vectors from the unit Gaussian prior and concatenating them with conditional prior vectors, $\mathbf{c} \in \mathbb{R}^m$, which function to bias the type of sample we want to synthesize.

4.4. Experiments

We evaluate our discriminative and generative SpiralResNets for several supervised and unsupervised tasks respectively. First, we introduce the 3D structural neuroimaging dataset and describe our convention for assigning the appropriate *in-vivo* diagnosis labels for each trimesh sample (subsubsection 4.4.1.1). Next, we detail the preprocessing parameters used within our experiments for generating the spiral sequences at each level of trimesh coarsening (subsubsection 4.4.1.2).

In subsection 4.4.2, we conduct an experiment with our discriminative model to analyze the efficacy of incorporating input data from multiple subcortical regions in binary AD/MCI classification. Our results demonstrate a clear advantage to the joint modeling of multiple subcortical regions, as opposed to using a single hemisphere or single region. In subsubsection 4.4.2.2, we provide a baseline comparison to alternative shape-based operators, in place of spiral convolution, for the same binary classification tasks. In subsubsection 4.4.2.3, CAMs are generated for samples that are correctly classified as AD by our SpiralNet classifier. These CAMs fall in accordance with previous reports of morphological changes observed in the brain that are correlated with AD pathology. Our CAMs support our classification results by producing visual transparency into our discriminative SpiralResNet's reasoning for true positive (TP) AD classification.

Lastly, in subsection 4.4.3, we evaluate the effect of conditioning our generative models on AD diagnosis with respect to each subcortical region. Our generative network's results demonstrate our model captures morphological differences in the presence of AD for some of the subcortical regions, particularly the hippocampi and amygdala nuclei, which are in accordance with previous autopsy reports that highlight patterns of atrophy in those regions in the presence of AD.

4.4.1. Dataset and pre-processing

4.4.1.1. ADNI dataset. In this study, we use 8,665 T1-weighted 3D MRI volumes from the Alzheimer's Disease Neuroimaging Initiative (ADNI) dataset, corresponding to 1,266 unique subjects. For each scan, we associate the healthy control (HC), mild cognitive impairment (MCI), or Alzheimer's disease (AD) labels given up to 2 months after the corresponding scan in ADNI. This is done as a precaution to ensure that each diagnosis had clinical justification.

Each discriminative model in this work is designed to classify pathological (AD/MCI) scans apart from HCs. To ensure that scans from the same subject do not appear in different sets, all data splits (training/testing/validation) in this study shuffle samples by *subject* identifiers instead of *scan* identifiers. We randomly split our data into training/testing sets (85/15%) across subjects, and use a 5-fold cross-validation across the subjects within the training set in our analyses.

Trimeshes are extracted from each T1-weighted MRI sample using the trimesh extraction preprocessing method described in subsection 3.2.2 of the previous chapter's methodology. Each subcortical region is represented using an independent trimesh surface with the corresponding number of vertices, edges, and faces described in Table 3.1, per hemisphere. Using the trimesh [**167**] library in the Python [**168**] programming language, a trimesh object for one hemisphere of a subcortical region is represented using a vertex feature matrix, $\mathbf{X} \in \mathbb{R}^{N \times 3}$, described in Table 1.1, and a corresponding set of faces, $\mathcal{F}$, which is a set of three-element tuples where each tuple indexes the vertices that make up the corresponding triangular face on the mesh.

The vertex feature matrix of a trimesh sample containing a single bilateral subcortical region (i.e. LH/RH hippocampi) is constructed using a row-wise concatenation (vertical stacking) of the vertex feature matrix for each hemisphere of the corresponding subcortical region. The sets of faces for each hemisphere of the same subcortical region are merged to create one set of faces for the bilateral subcortical region sample type. For trimesh samples representing a single hemisphere or multiple subcortical regions, the corresponding vertex feature matrix and face sets are obtained using the same vertical stacking and merging process. Each trimesh sample type can be interpreted as an undirected graph described by a vertex feature matrix, $\mathbf{X}$, and a corresponding set of faces, $\mathcal{F}$. Therefore, $14,848$ vertices are used to represent a single trimesh sample for all subcortical regions, $7,424$ vertices for a single hemisphere, and $2N$ vertices per subcortical region, for each corresponding N in Table 3.1.

4.4.1.2. Spiral sequence and mesh-sampling generation. Following the encoder module described in subsubsection 4.3.4.2 and depicted in Figure 4.3, the topology of spiral sequences at each level of trimesh coarsening is only pre-processed *once*. In-order, the spiral lengths, L_r, used for the spiral filters within the r-th ResBlock of the encoder are $\{L_r\}_{r=1}^{5} = \{12, 12, 12, 12, 9\}$, with the corresponding dilation parameters, $\{d_r\}_{r=1}^{5} = \{2, 2, 2, 1, 1\}$. These parameters are used in reverse-order for the ResBlocks within the the convolutional decoder, depicted in Figure 4.4.

Following the steps in subsection 4.3.3, down/up-sampling matrices were generated *once* to represent surfaces in this study at multiple hierarchical levels while still preserving context at each level. Again following the structure of the encoder (Figure 4.3), we specifically up/down-sample trimeshes within the architecture by a factor of 2 for each trimesh sampling operation. At each level of coarsening, spiral sequences are generated once using the template trimesh.

4.4.2. Discriminative model predictions

Discriminative models and hyperparameter tuning were evaluated using a 5-fold cross-validation on the training set, as explained in subsubsection 4.4.1.1, for two separate experiments. In our first experiment, we analyze the efficacy of incorporating input data from multiple subcortical regions for binary AD/MCI classification, in comparison to input data from a single hemisphere or single region using a SpiralResNet classifier. In our second experiment, we analyze the performance of alternative *shape*-based classifiers in comparison to our proposed SpiralResNet model. We report the results on the test set for each classification task. The number of filters, per convolutional layer, at the r^{th} ResBlock, within the encoder is $\{F_r\}_{r=1}^{5} = \{32, 64, 64, 128, 128\}$. A binary cross-entropy (BCE) loss function was optimized to train all discriminative models using the *AdamW* [**169**] optimizer with a learning rate of 2×10^{-4}, learning rate decay of 0.99 for every step, and a batch size of 16 samples per batch over 200 epochs. In addition to the BCE loss, the parameters of the networks were also ℓ_2-regularized with a weight decay of 1×10^{-5}.

4.4.2.1. Subcortical region ablation study. First, we perform binary classification tasks on different combinations of subcortical regions to classify scans using the diagnostic labels provided by ADNI. The first task is to classify HC scans apart from those belonging to subjects with AD, meanwhile the second task looks at HC vs. MCI classification. For each task, we use the same discriminative SpiralResNet (architecture hyperparameters and number of learnable parameters from subsubsection 4.3.4.4), and train each model on the same task, each with a varied combination of input regions. Classifiers are trained and compared with: (a) single-region (both hemispheres), (b) single-hemisphere, and (c) all-region trimesh inputs for each sample.

Table 4.1a summarizes the results of the experiments on **Alzheimer's disease (AD) classification** across variations of subcortical region inputs. The discriminative model's performance gradually improves with the inclusion of more subcortical regions. In particular, an improvement in classifier performance is observed when an entire hemisphere (an input with multiple subcortical regions), is used versus using both hemispheres of a single subcortical region. The discriminative model performs best when *all* subcortical regions (the largest input option) are used as input. The discriminative model trained on the left hemisphere (LH) slightly outperforms the model trained on the right hemisphere (RH) in both Area Under the Curve (AUC) statistics, which may also be indicative of the way AD pathology is typically diagnosed. The LH of the human brain is tightly associated to language function (i.e. grammar, vocabulary, and literal meaning) [**170**], which is often used as a metric for the clinical diagnosis of AD.

AD follows a different trajectory than normal aging [**171**]. Language and memory problems like forgetfulness can be correlated with normal aging, however the types of memory problems that occur with AD dementia are more severe and typically begin to interfere with activities of daily living (ADLs), which is not a part of normal aging. One example: forgetting where you put your glasses, can be indicative of disorganization, forgetfulness, or normal aging. However, forgetting *what* glasses are used *for*

Table 4.1. Binary classification results using same SpiralResNet discriminative network. Precision, recall, and F1-score are reported with respect to a classification threshold of 0.5. For a global measure over different thresholds we also report the Area Under the Curve (AUC) of the receiver operating characteristic (ROC) curve (ROC-AUC) and the AUC for the Precision-Recall curve (PR-AUC) for each case.

(a) healthy control (HC) vs. Alzheimer's disease (AD)

Region	**Threshold = 0.5**			**AUC**	
	Precision	*Recall*	*F1*	*ROC-AUC*	*PR-AUC*
all regions	**0.877**	0.834	**0.855**	**0.906**	**0.895**
left hemisphere	0.827	0.700	0.758	0.893	0.874
right hemisphere	0.737	0.798	0.766	0.887	0.863
amygdala	0.788	**0.850**	0.818	0.900	0.891
caudate	0.524	0.655	0.582	0.699	0.592
hippocampus	0.682	0.722	0.702	0.812	0.708
nucleus accumbens	0.610	0.674	0.640	0.774	0.690
pallidum	0.543	0.644	0.589	0.700	0.556
putamen	0.642	0.637	0.639	0.780	0.705
thalamus	0.611	0.723	0.662	0.780	0.703

(b) healthy control (HC) vs. mild cognitive impairment (MCI)

Region	**Threshold = 0.5**			**AUC**	
	Precision	*Recall*	*F1*	*ROC-AUC*	*PR-AUC*
all regions	0.613	**0.712**	0.659	0.612	**0.693**
left hemisphere	0.629	0.616	0.622	0.589	0.649
right hemisphere	0.645	0.631	0.637	**0.622**	0.691
amygdala	0.643	0.561	0.599	0.607	0.689
caudate	0.628	0.565	0.595	0.578	0.635
hippocampus	0.597	0.705	0.647	0.549	0.622
nucleus accumbens	0.573	0.625	0.598	0.503	0.597
pallidum	0.597	0.698	0.643	0.533	0.617
putamen	0.602	0.551	0.575	0.529	0.618
thalamus	**0.646**	0.677	**0.661**	0.617	0.593

(their utility) is not a part of normal aging. Like many anomaly detection problems in medical imaging, where it is important to anticipate pathological events that occur less times than the healthy control, *precision-recall* statistics (see Table 4.1a and Table 4.1b) are often stronger for measuring classification performance when there is a class imbalance and the class of interest belongs to the smaller population.

There exists strong evidence for certain patterns of atrophy for different neuroanatomical regions at different stages of AD progression [**172**]. Early involvement of the entorhinal cortex, hippocampus, and amygdala in AD progression have been reported consistently in the literature [**173, 174, 175, 176**]. Our results in Table 4.1a suggest a stronger performance in AD classification given the *shape* of the amygdala or hippocampus alone compared to the other subcortical regions. Most importantly, these results also demonstrate that a holistic approach incorporating multiple subcortical regions improves AD classification.

Table 4.1b demonstrates the results of **Mild Cognitive Impairment Classification**. An expected drop in performance occurs for MCI classification, compared to AD. This behavior is expected due to the MCI group's variability, given its detection being more symptomatic and it is also sub-divided into several stages. Detecting MCI is important because people with MCI are more likely to develop AD than those without. Unlike the fluidity of the MCI pathological spectrum, AD progression is endemic and symptoms worsen with time. However, methods for slowing down and mitigating the effects of AD make its early detection favorable.

4.4.2.2. SpiralResNet classifier baseline comparison. Given the improvement in AD classification using input data from multiple subcortical regions for our discriminative model, we compare our model's performance with other baseline *shape-based* classifiers on the same dataset. We evaluate four different methods to perform the same discriminative tasks as subsubsection 4.4.2.1: (1) the discriminative SpiralResNet in this work, (2) the same discriminative module with spectral graph convolution in-place of spiral convolution, (3) the end-to-end discriminative network by [**89**], and (4) a MLP trained on the latent space features of the generative network in this work.

Spectral networks set-up

The spectral GCN [**3**] (referred to as ChebyNets) comparison, demonstrates an improvement in performance with batch normalization (BN) and a residual learning architecture by training and evaluating multiple learning architectures. We construct (1) a ChebyNet using the same architecture as the discriminative SpiralResNet in Figure 4.5, but with spectral GCN layers, BN, and ELU activations in place of the Spiral ResBlocks, and another network using "ChebyNet ResBlocks," where spiral convolution operations within a ResBlock are replaced with GCN layers. For a fair comparison, we use the same network depth as the SpiralResNet discriminator, the same number of output features per convolutional layer, and a Chebyshev polynomial of degree $K = 6$ for each spectral convolutional layer [**3**]. The second MLP-half of each ChebyNet model follows the same MLP architecture used within our discriminative SpiralResNet (Figure 4.5).

Point cloud networks set-up

To utilize the same dataset on this method, we drop the edges of our 3D trimeshes and treat the surface vertices as point clouds representing the surface/shape of the subcortical regions. The shared MLPs within the architecture of the discriminative model constructed by Gutiérrez-Becker *et al.* [**89**] to operate on point clouds, are identically implemented using 1D convolutional layers with a kernel size of 1 [**137, 147**]. For consistency in adopting the PointNet-inspired model for a fair comparison, we use the PointNet layers described by Gutiérrez-Becker *et al.* [**89**], and construct the same discriminative network in Figure 4.5, with point cloud operations in place of spiral operations. We construct a PointNet discriminator with (1) 1D convolutional layers + no activation following Gutiérrez-Becker *et al.* [**89**], in place of the ResBlocks, (2) the same PointNet model in addition to BN + ELU activations after each convolutional layer, and (3) a final variant with "PointNet ResBlocks" following the same style as the SpiralResNet ResBlocks and "ChebyNet ResBlocks" in the spectral set-up. The second MLP-half of each PointNet model follows the same MLP architecture used within our SpiralResNet discriminative model (Figure 4.5).

Generative model latent space set-up

A generative model (subsubsection 4.3.4.5) was constructed with $\{F_r\}_{r=1}^{5} = \{128, 128, 128, 128, 256\}$ for the corresponding output feature maps of the model's encoder and decoder SpiralResNet ResBlocks. We found it best to compress trimesh samples down to a latent space using $\mathbb{R}^{16}$ components for each subcortical region, therefore resulting in $\mathbf{z} \in \mathbb{R}^{112}$ for all subcortical regions. A binary one-hot encoding vector is used for the condition vector $\mathbf{c} \in \mathbb{R}^2$, with respect to the diagnoses for each sample.

The generative network was trained by optimizing the loss function in Equation 4.9 and using ℓ_2-regularization, weighed by 1×10^{-5}, on the network's learnable parameters. The AdamW [**169**] optimizer is used with a learning rate of 2×10^{-4}, learning rate decay of 0.99 for every step, and a batch size of 8 samples per batch over 500 epochs of training. Once trained, a MLP following the same architecture as the second MLP in the discriminative network (Figure 4.5), is trained on the latent space shape descriptors (i.e. $\mathbf{z}$) of the corresponding samples, using the same data splits as the other baseline comparisons. This MLP is also trained using the AdamW [**169**] optimizer, with the same training parameters as the rest of the discriminative baseline models.

AD model comparison

For the AD binary classification task, the model comparison results in Table 4.2a demonstrates that our discriminative SpiralResNets used in the previous ablation study (subsubsection 4.4.2.1) outperforms all the baseline models in precision, recall, and F1 score for a 0.5 binary classification threshold. Our SpiralResNet model also outperforms the baseline models in both AUC statistics, particularly the PR-AUC indicating an overall improvement in precision, recall, and F1 score across multiple classifier thresholds in $[0, 1]$.

The spectral classifier without residual connections (ChebyNet in Table 4.2a) performs the worst overall. However, with the addition of the residual learning

Table 4.2. Baseline comparison of binary classifiers for HC vs. AD/MCI classification.

(a) healthy control (HC) vs. Alzheimer's disease (AD)

Model	**Threshold = 0.5**			**AUC**	
	Precision	*Recall*	*F1*	*ROC-AUC*	*PR-AUC*
SpiralResNet (Ours)	**0.877**	**0.834**	**0.855**	**0.906**	**0.895**
Generative (Ours)	0.703	0.771	0.735	0.851	0.769
ChebyNet	0.487	0.644	0.555	0.664	0.580
ChebyResNet	0.740	0.757	0.748	0.869	0.837
PointNet	0.791	0.798	0.795	0.803	0.786
PointNet+BN+ELU	0.802	0.776	0.789	0.798	0.774
PointResNet	0.842	0.814	0.828	0.836	0.822

(b) healthy control (HC) vs. mild cognitive impairment (MCI)

Model	**Threshold = 0.5**			**AUC**	
	Precision	*Recall*	*F1*	*ROC-AUC*	*PR-AUC*
SpiralResNet (Ours)	**0.613**	0.712	0.659	0.541	**0.693**
Generative (Ours)	0.595	0.776	0.673	0.524	0.629
ChebyNet	0.602	0.820	**0.694**	0.542	0.612
ChebyResNet	0.591	**0.827**	0.689	0.521	0.610
PointNet	0.590	0.789	0.676	0.528	0.615
PointNet+BN+ELU	0.595	0.826	0.692	**0.557**	0.639
PointResNet	0.601	0.702	0.648	0.542	0.616

framework by using "ChebyNet ResBlocks," we see an improvement in performance across all metrics for the ChebyResNet model; in fact, it ranks second-highest in both AUC scores behind our spiral model. In our prior work [**96**], ChebyResNets were used for the same AD binary classification task on the same subcortical regions used in this study, in addition to the corresponding white and pial cortical surface trimeshes for each sample. In that work, ChebyResNet outperformed the baseline classifiers, demonstrating an improvement in performance by directly learning on surface trimeshes with spectral graph convolution. In this work, ChebyResNet outperforms the PointNet variants, indicating again an improvement in performance over non-surface trimesh approaches.

The bare PointNet model, without activation functions or a residual framework, performed better across all metrics (shown in Table 4.2a), in comparison to the bare ChebyNet classifier. The PointNet models progressively improves overall with the addition of BN and ELU activations, and with the residual learning framework. The PointResNet model does outperform the ChebyResNet model in precision, recall, and F1-score given a 0.5 binary classification threshold, however not in AUC statistics taken over several thresholds in $[0, 1]$.

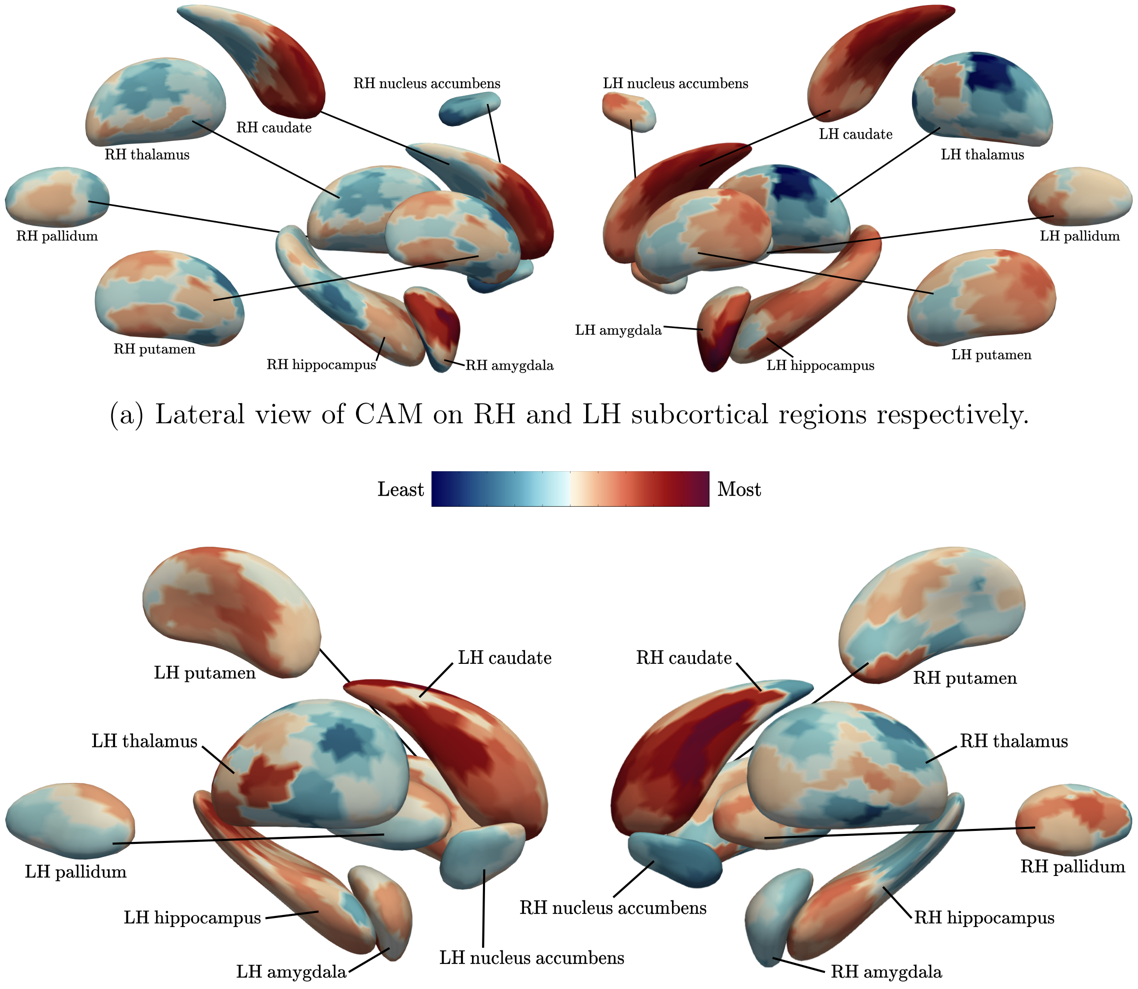
MCI model comparison

Like our region ablation experiment, we see a drop in performance for all discriminative models in binary MCI classification. The same network set-up used for AD classification was used in this evaluation, treating MCI as the positive label. Our SpiralResNet classifier achieves the highest PR-AUC in MCI classification when compared to the baseline methods. The overall drop in performance for MCI classification for all models in this experiment is the same behavior analyzed in the previous experiment.

4.4.2.3. Class activation maps (CAMs) for binary AD/HC classification. Using the pre-trained SpiralResNet classifier trained on all the subcortical regions, we generate class activation maps (CAMs), using our Grad-CAM adaptation on trimeshes, for each AD sample in the test set that is correctly classified by our model (true positive (TP) predictions), given a 0.5 classifier threshold. CAMs for the TP samples are then averaged and projected onto the vertices of the subcortical template [**113, 177**]. Trimesh faces are colored using an interpolation based on the CAM values at the vertices of each corresponding triangle (each face has three corresponding vertices). The color-map scale used to visualize the TP CAM in Figure 4.7a and Figure 4.7b highlights areas along the surface by their magnitude of influence, ordered from least to greatest, in binary AD classification with our trained discriminative model.

Aligning with our discriminative SpiralResNet model's results in the region ablation study, we observe a strong involvement of hippocampus and amygdala shape in AD vs. HC classification. In AD, it has been demonstrated that cortical atrophy occurs earlier and progresses faster in the LH than in the RH [**178, 179**]. Wachinger *et al.* [**85, 86, 78**] demonstrated a significant leftward asymmetry in cortical thinning (mainly in the temporal lobe and superior frontal regions) with an increase in hippocampal asymmetry, which remains consistent with previous findings demonstrating an asymmetric distribution of amyloid-β [**180**], a protein in the brain that is thought to be toxic and naturally occurs at abnormal levels in the brains of subjects living with AD.

Both caudate regions are also highlighted as indicative of TP classifications, again with a similar leftward asymmetry. In particular, there is an emphasis on the tail of the left caudate nucleus. This observation falls in line with the findings of [**181**] where both the left and right caudate nucleus were smaller in volume for patients



(a) Lateral view of CAM on RH and LH subcortical regions respectively.

(b) Medial view of CAM on RH and LH subcortical regions respectively.

Figure 4.7. Average of class activation map (CAM) for true positive (TP) predictions by the SpiralResNet discriminative network proposed in this study. A CAM is generated for each TP prediction and their average is projected onto the subcortical region template trimesh by [**113**]. Provided are lateral (a) and medial (b) views of the CAM projected on the template, which follows the color-scale map which at the center of the two subfigures.

with dementia compared to age-matched HCs; in fact, their findings show that the left caudate volume difference was significant in AD subjects ($p < 0.01$). In a recent study looking at shape differences in the ventricles of the brain with respect AD, Ferrarini *et al.* [**182**] show that the areas adjacent to the anterior corpus callosum, splenium of the corpus callosum, *amygdala*, thalamus, tails of the *caudate* nuclei, and the head of the *left caudate* nucleus are all significantly affected by AD, which are also highlighted within our generated CAMs.

Volume reductions in the putamen, hippocampus, and thalamus volume were observed by De Jong *et al.* [**101**], adhering to the potential left putamen involvement depicted in Figure 4.7b. On the left hippocampus region particularly, we see widespread involvement of the region with most of the predictive activity occurring at the tail of the left hippocampus and roughly around the CA1 subfield, also reported by Gutiérrez-Becker *et al.* [**89**].

On average, we observe an asymmetry towards the CAMs of the LH regions as more indicative of AD than the RH, even when trained on both hemispheres at once. Our ablation study also demonstrates an improvement in classifier AUC performance (Table 4.1a) with using the LH versus the RH in AD classification. Several studies point towards a left-lateralization of brain atrophy in AD [**178, 179**], however Derflinger *et al.* [**102**] argue that brain atrophy in AD is asymmetric rather than lateralized and that data suggesting leftward lateralization may be a result of selection bias. This may be due to the fact that clinical scores used to diagnose AD are primarily language-based, resulting in a potential bias towards a selection of patients already with left-lateralized atrophy [**183**].

4.4.3. Diagnostic conditioning on generative CVAE model

Differences in output generation with respect to AD diagnosis was done with the point cloud generative models of Gutiérrez-Becker *et al.* [**89**]. Shape variations their model associates to the presence of AD are measured with point-to-point metrics like ℓ_1 distance. Choi *et al.* [**157**] also experiment with modifying their CVAE's condition vectors to generate synthetic PET images and forecast future age-related metabolic changes. Predicted regional metabolic changes were correlated with the real changes in their follow-up data. In this work, we observe changes in trimesh surface area, A, and volume, V, with respect to the HC and AD labels, given the same latent vectors for the set of HC samples. The shape descriptors learned by our generative model that are used in our discriminative model evaluation (Table 4.2a) demonstrate potential in encoding complex shape variations using a low-dimensional embedding.

For our final evaluation, we use the same CVAE architecture used by our generative model in subsubsection 4.4.2.2 to construct a generative CVAE model with respect to each subcortical region, using $\mathbf{z} \in \mathbb{R}^{16}$ as the dimension of the latent space for each model. For each CVAE model we use a binary one-hot encoding with respect to the AD vs. HC labels in our dataset as the condition vector, $\mathbf{c} \in \mathbb{R}^2$, to analyze the effect of conditioning on AD diagnosis *per region*. Each CVAE model is trained following the same training parameters and AdamW optimizer used to train the generative network in our baseline classifier comparison (subsubsection 4.4.2.2).

First we train each generative network on the entire dataset of HC and AD samples. Next, we extract the latent space embedding (i.e. $\mathbf{z} \in \mathbb{R}^{16}$ for each subcortical region) of each HC sample in the dataset. With the latent space shape descriptor of each region for each HC sample, we analyze the effect of changing the HC label to AD before the decoding step of each generative network to see how diagnosis affects the generated trimesh output.

Based on the literature regarding changes in the hippocampus shape as a result of AD, Figure 4.8 qualitatively depicts some of the hippocampus results in four randomly selected (originally HC) samples. Qualitatively, we observe a "thinning" in hippocampus volume for each hemisphere, particularly shown in the examples of the second (LH) and third (RH) columns in Figure 4.8. The histograms spread throughout Figure 4.9-4.12 quantitatively depict the observed corresponding volumes, V, and surface areas, A, with using the HC samples and changing the diagnosis during decoding. The volume of a watertight trimesh is determined using a surface integral, and the surface area is determined as the sum of the areas of all the triangles on a trimesh.

Given that the diagnosis labels are categorical and we are analyzing the effect of conditioning the generative shape model using these labels, we use the non-parametric Kruskal-Wallis H-test [**184**] to measure the statistical significance of differences in the output of the model with respect to each label. For each histogram, we report the corresponding H-value and p-value. For the left putamen, left pallidum, and right pallidum, differences in the volumes of generated outputs are not statistically significant ($p > 0.05$). For the left caudate, left nucleus accumbens,

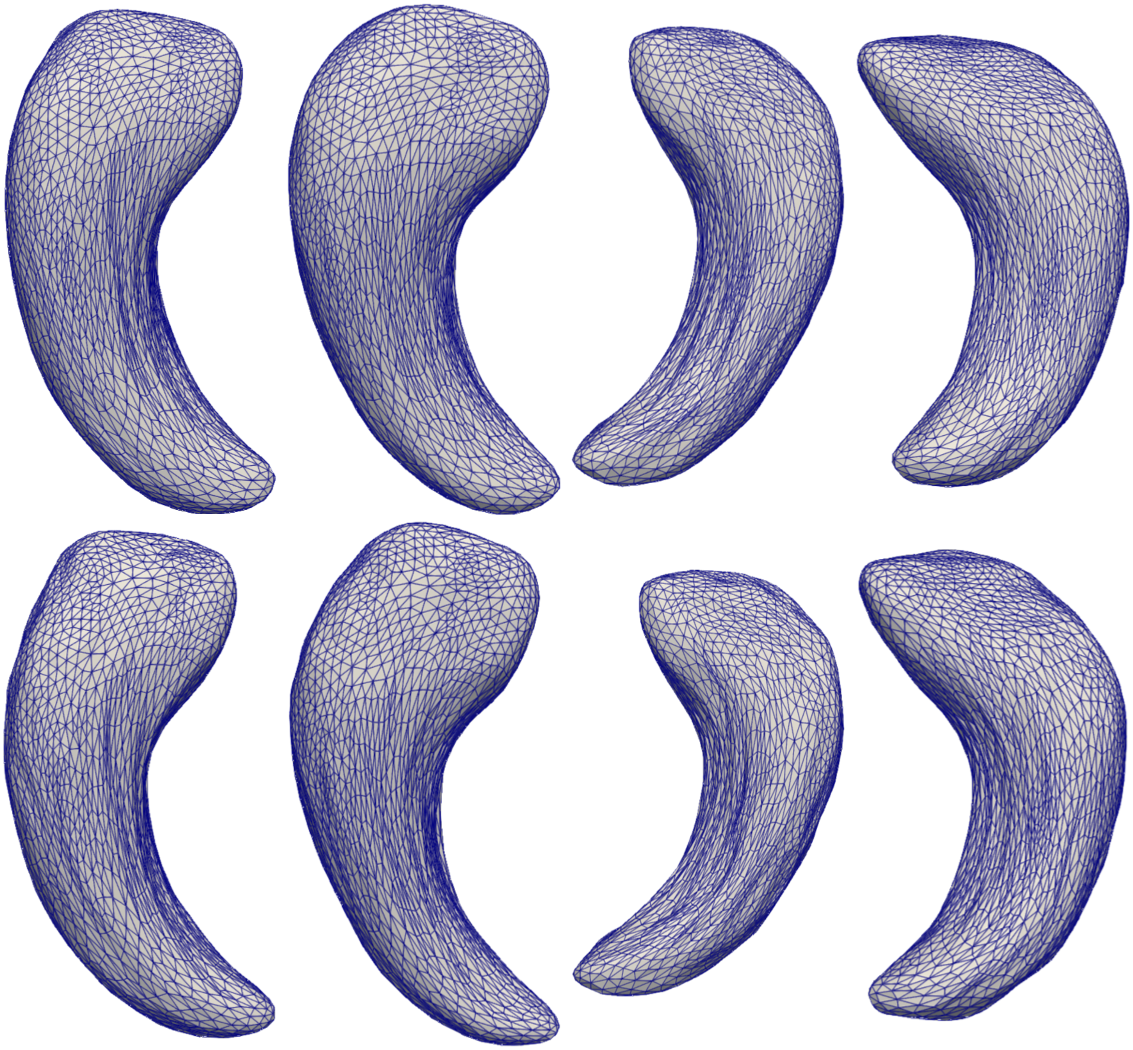


Figure 4.8. Dorsal views of the left and right hippocampus surfaces generated using proposed generative CVAE model on ADNI dataset. For a given latent space vector, $\mathbf{z}$, a 3D trimesh is generated by conditioning on the HC (top row) or AD (bottom row) label that is passed along to the decoder along with $\mathbf{z}$. Each column corresponds to a different HC sample.

left/right pallidum, right putamen, and left/right thalamus, there is no statistical significance ($p > 0.05$) in the differences in surface area.

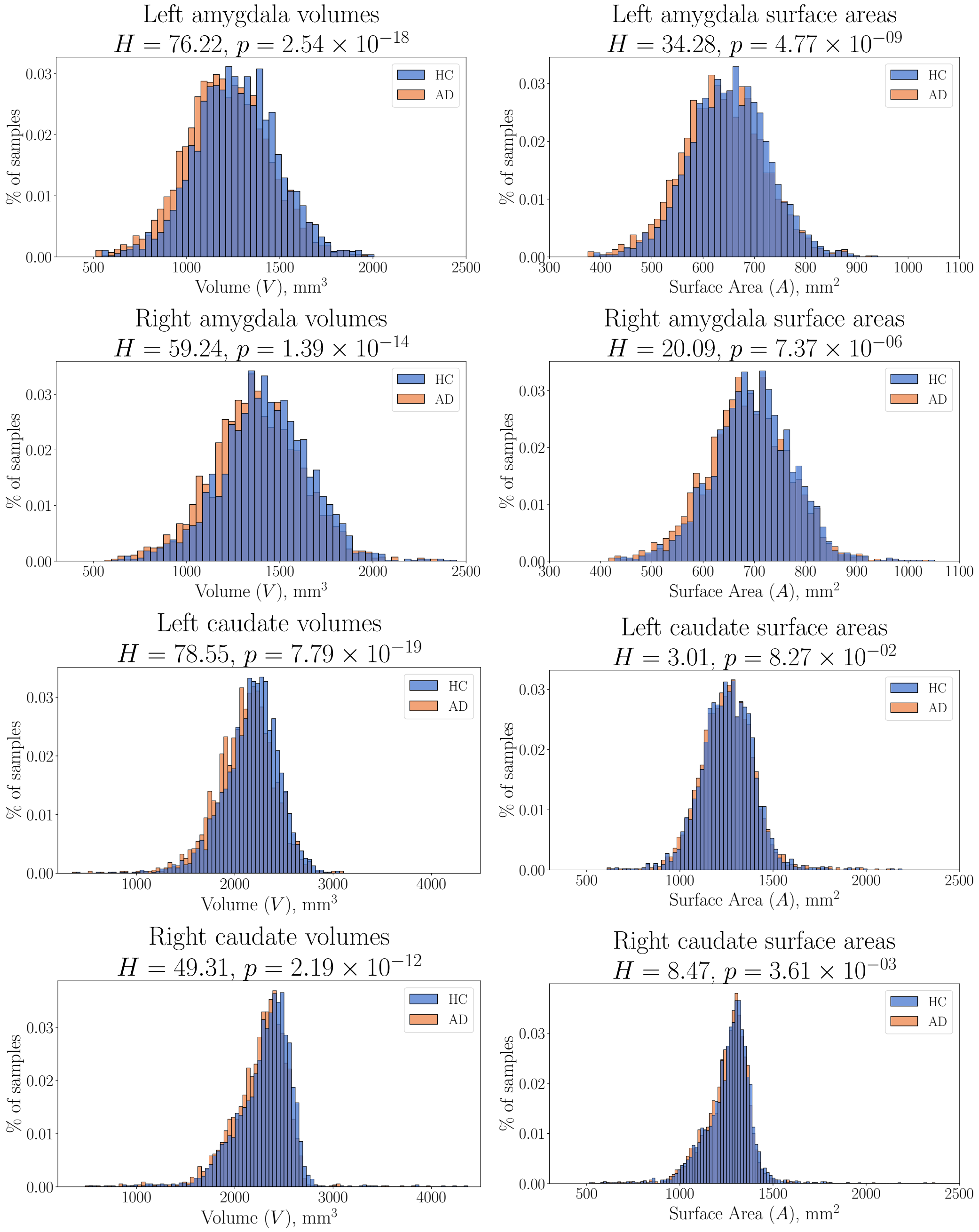


Figure 4.9. Observed changes in output volume and surface area for amygdala (first two rows) and caudate (bottom two rows).

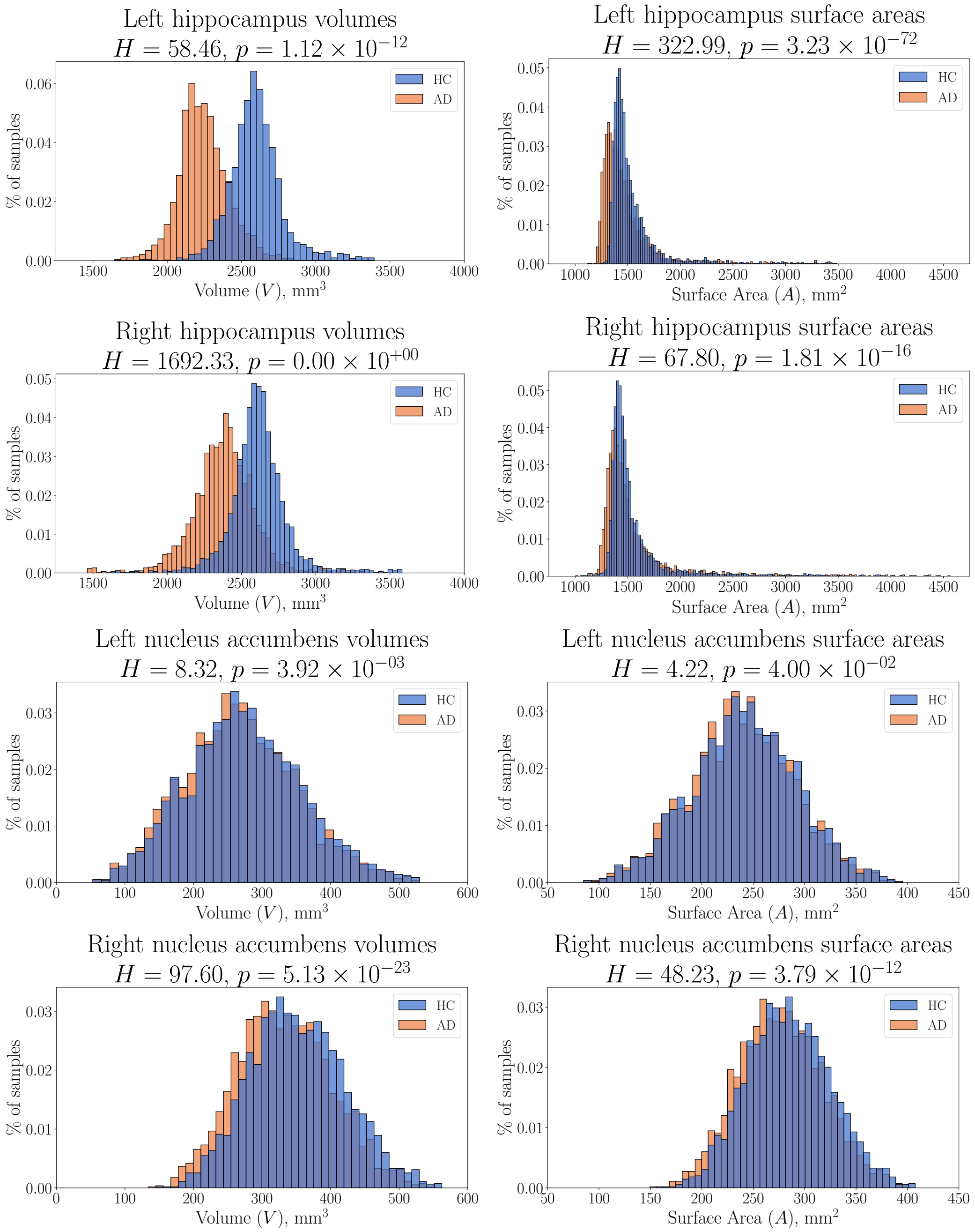
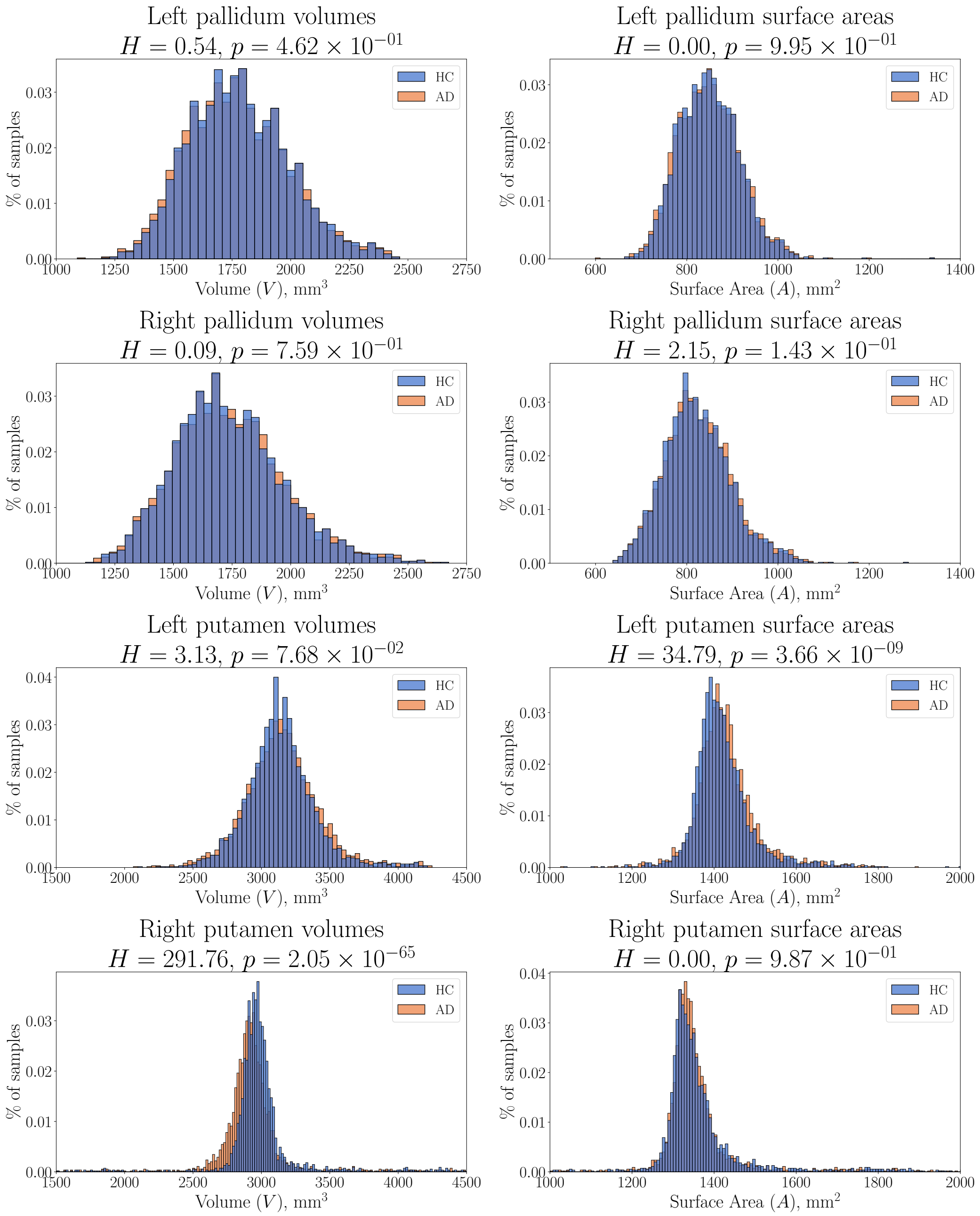


Figure 4.10. Observed changes in output volume and surface area for hippocampus (first two rows) and nucleus accumbens (bottom two rows).



Left pallidum volumes $H = 0.54$, $p = 4.62 \times 10^{-01}$

Left pallidum surface areas $H = 0.00$, $p = 9.95 \times 10^{-01}$

Right pallidum volumes $H = 0.09$, $p = 7.59 \times 10^{-01}$

Right pallidum surface areas $H = 2.15$, $p = 1.43 \times 10^{-01}$

Left putamen volumes $H = 3.13$, $p = 7.68 \times 10^{-02}$

Left putamen surface areas $H = 34.79$, $p = 3.66 \times 10^{-09}$

Right putamen volumes $H = 291.76$, $p = 2.05 \times 10^{-65}$

Right putamen surface areas $H = 0.00$, $p = 9.87 \times 10^{-01}$

Figure 4.11. Observed changes in output volume and surface area for pallidum (first two rows) and putamen (bottom two rows).

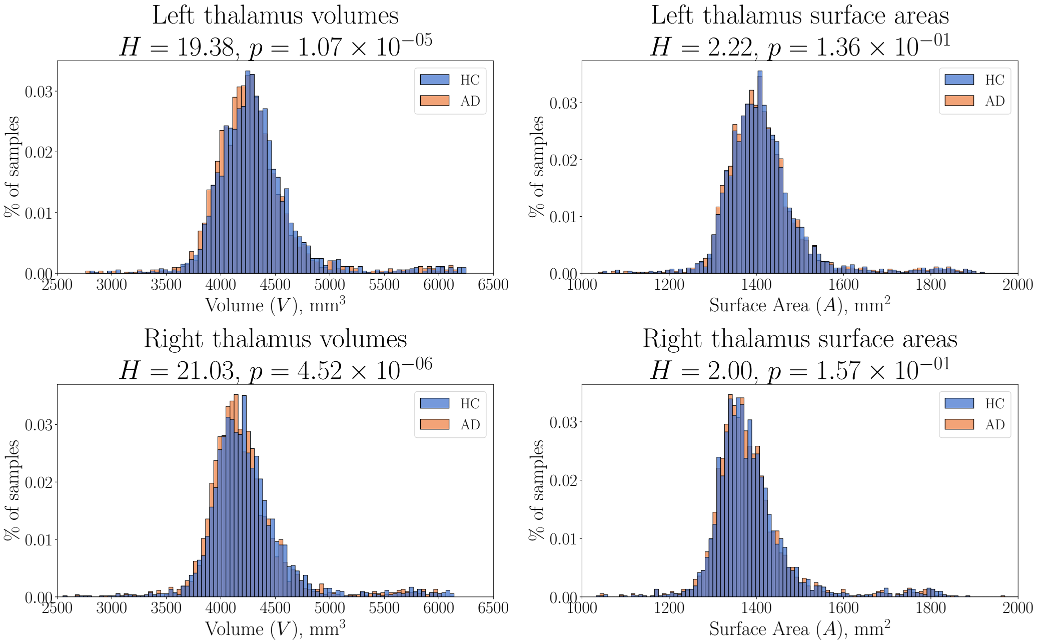


Figure 4.12. Observed changes in output volume and surface area for thalamus.

For each of the remaining subcortical regions, a reduction in volume and surface area is the most common observation, especially in both hippocampi ($p \ll 0.001$). We hypothesized our generative model would learn to reduce the hippocampus and amygdala regions, areas that are highly correlated with language, memory, frontal executive function scores. Our results for the remaining regions are in accordance with the expected shrinking of each region in the presence of AD, coinciding with previous autopsy reports in AD progression [**173, 185**].

4.4.4. Summary of experiments

Our results for *in-vivo* AD vs. HC classification with SpiralResNets on brain surface trimeshes demonstrate the powerful discriminative advantage in learning surface representations of subcortical brain regions. Spiral CNNs are demonstrated to outperform recent methods which operate on point cloud representations or use spectral graph convolution on the same template-registered trimeshes in this study. To the best of our knowledge, the SpiralResNet method proposed in this study is a state-of-the-art (SOTA) approach that exploits the locally-Euclidean properties of vertices distributed across a surface to design learnable anistropic filters that improve AD classification with respect to subcortical region *shape*. Our results demonstrate a clear advantage to incorporating multiple subcortical regions, as opposed to input data from a single subcortical region or hemisphere.

The CAMs obtained using our discriminative SpiralResNets draw direct correspondences with the literature regarding localized areas of deformation related to AD pathology. Paired with our discriminative SpiralResNet, our framework combines localized contextual visualization together with classification results. More often, a modular visualization method that provides context to a discriminative model's predictions *without making architectural changes to the model*, is highly desirable for establishing appropriate trust in predictive models.

Furthermore, the results of our generative SpiralResNet demonstrate the potential for using diagnosis in the condition vector, as a means to add more specificity to the type of output that is generated. Our generative framework illustrates a potential application for generating synthetic training data that would be beneficial for improving deep learning frameworks that benefit from increased dataset sizes. Significant volume and surface area changes with respect to AD diagnosis were identified, particularly in the amygdala, caudate, nucleus accumbens, right putamen, thalamus, and most importantly the hippocampus, an area of the brain highly correlated with AD. Our prior work using spectral filters [**96**] utilizes the same subcortical regions, in addition to the cortex, to perform the same AD classification task. However, during our analysis, we observed frequent graphics processing unit (GPU) memory issues with training a CVAE SpiralResNet using the cortical surface. A major degradation in the output quality of reconstructed/generated cortical surfaces with our generative framework was also observed. As Gutiérrez-Becker *et al.* [**89**] point out, modeling a region with a more complex geometry, e.g., the cortex, requires a larger number of points that may lead to GPU memory constraints. Additionally, the gyrification of the cortical surface is much more complex and may require additional methods that generalize better to 3D mesh regions with complex sulci and gyri.

4.5. Conclusions and Future Work

To the best of our knowledge, no existing works have investigated brain shape in regards to AD pathology using discriminative *and* generative SpiralResNets that learn and operate *directly on surface trimeshes* by way of geometric deep learning. Our framework is constructed by a variety of modular computational blocks that are used by both our discriminative and generative SpiralResNets. Notably, our convolutional encoder learns effective shape descriptors that can be used for AD classification by our discriminative SpiralResNet. Our first analysis demonstrates an improvement in AD classification performance using the *same model* with varying input types: (a) single subcortical region, (b) subcortical regions within a single hemisphere, and (c) bilateral subcortical regions. Our results demonstrate a clear advantage to the joint modeling of multiple subcortical regions for *in-vivo* AD classification.

Our discriminative SpiralResNet also outperforms alternative shape descriptor methods in our baseline comparison. Additionally, our adaptation of Grad-CAM to 3D trimeshes provides context as to which subcortical brain regions are driving our AD classification results. Our class activation maps (CAMs) are in accordance with the literature on morphological changes observed in the brains of subjects with AD. Our CAMs make our classification results more transparent by producing visual explanations. Improving clinical confidence and reliability in automated discriminative methods, can be approached by contextualizing a model's reasoning about its beliefs and actions for clinicians to trust and use.

Additionally, our generative SpiralResNet's decoder module is able to reconstruct 3D trimesh inputs from their low-dimensional shape descriptors obtained by the encoder. More importantly, in using a variational approach, we're able to learn a probabilistic latent space that can be sampled from to generate synthetic samples for each subcortical region with respect to phenotype information, in particular: AD diagnosis. The endemic nature of medical imaging data, particularly within neuroimaging, attributes to scarcity of open-access neuroimaging databases. Our generative SpiralResNet is able to generate realistic-looking synthetic examples, which may be used to train other deep learning approaches that often require large datasets and annotated data is limited.

Our proposed discriminative SpiralResNet can be further tailored to fuse other phenotypic data for AD classification; including but not limited to: chronological age, sex assignment at birth, genotype data, etc. Phenotype features can also be used as additional conditional priors in our generative framework, adding additional constraints for synthesizing personalized samples. Natural extensions of this work could include (1) expanding the classification task to sub-typing different stages of mild cognitive impairment (early vs. late), (2) using spiral convolution within a recurrent neural network framework for longitudinal predictions related to AD, and (3) experimenting with generating template-registered 3D trimeshes from MRI volume inputs using a spiral convolutional decoder framework to automate the trimesh extraction preprocessing steps.

4.6. Cortical Extension of AD/MCI Classification Using SpiralResNets

The subcortical brain region ablation study performed using SpiralResNets demonstrated the efficacy of a incorporating input data from multiple brain regions as opposed to studying individual regions for Alzheimer's disease (AD) vs. healthy control (HC) binary classification. However, technical limitations at the start of this study previously limited us to being able to incorporate additional brain shape information from the surface of the cortex due to its size.

Upon moving to an upgraded desktop environment with access to a single NVIDIA$_{®}$ TITAN V graphics processing unit (GPU) card, incorporating cortical information lead to improvements in AD/MCI vs. HC classification. Upon immediate retraining of SpiralResNets on trimeshes incorporating cortical surface features, the first problem we encountered was in being able to fit both the neural network (NN) and batches of data into working memory with a single GPU card. To successfully circumvent this issue, we decreased the batch size to 4 samples, as opposed to 16 samples per batch when just working with subcortical surface information.

4.6.1. Initial roadblock: no improvement in AD vs. HC classification

The same binary AD binary classification tasks performed for the discriminative study was performed again with the incorporation of cortical ribbon trimeshes. The optimal SpiralResNet architecture for the classification task on the subcortical regions did not improve in performance and worsened across some of the data splits in a 10-fold cross-validation, with the inclusion of cortical features.

During the preliminary stages of this extension, individual hemispheres of the cortex were analyzed for generating cortical surface trimeshes using the SpiralResNet CVAE. Quantitatively, the reconstruction errors obtained in training the CVAEs on individual cortical hemispheres were acceptable, however qualitatively, the reconstructed and generated cortical trimeshes were noisy, and poorly reconstructed the cortical folding pattern. We hypothesize that this may be due to the high frequency complexity of the cortical ribbon folding pattern and that are generative SpiralResNets are doing a poorer job at capturing high frequency features and mainly captures low frequency general-shape features. For this reason, we hypothesized that this difficulty in effectively learning high frequency cortical features may carry over to the performance of SpiralResNet classifier.

4.6.2. GoogLeNet Inception architecture

A major milestone in the ongoing development for state-of-the-art (SOTA) convolutional neural networks (CNNs), was the recent heavily-engineered GoogLeNet (Inception v1 [**186**]) architecture introduced by Szegedy *et al.* which used several tricks to boost speed and accuracy. Its popularity and impressive performance has lead to a constant evolution of several versions of the network architecture, with each version built out as an iterative improvement over the previous iteration (Inception v2/v3 [**115**] Inception v4/-ResNet [**187**]). The key innovation to these works are in the Inception module. Inception modules are blocks of parallel convolution layers with filters that vary by size with respect to each convolutional layer, the results of which are then concatenated. This allows the model to learn not only parallel convolutional filters of the same size, but also parallel filters of differing sizes, allowing learning at multiple scales at each Inception block.

Salient parts of images can have a large variation in size. For example, the area occupied by a dog is different for each image in Figure 4.13. Given this issue of huge variability in the locality of information in images, determining and fine-tuning

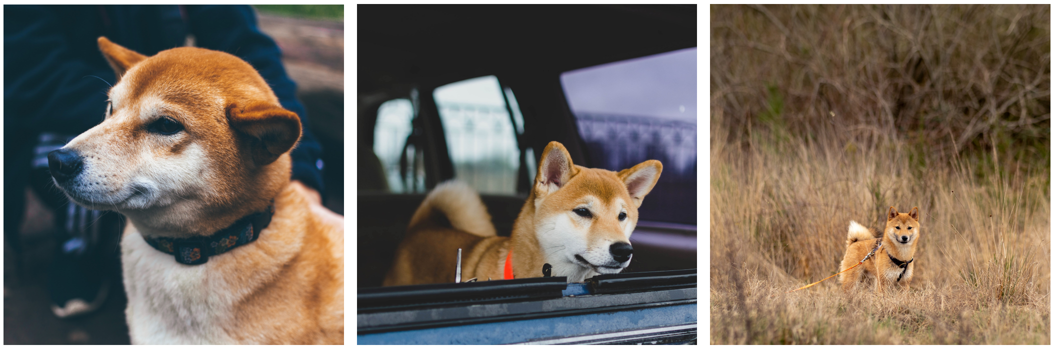


Figure 4.13. Shiba Inu dogs occupying varying areas of 2D images. From left to right: a dog occupying most of an image, a dog occupying part of an image, and a dog occupying a small portion of an image.

relevant hyperparameters, such as *filter size*, can become challenging. With CNNs, larger kernels are preferred for globally-distributed information, and a smaller kernel is preferred for locally-distributed information.

Nearly every iteration of the Inception architecture shares a common "stem" structure at the root of their NN architecture before the Inception modules. The stem begins with a *very wide* convolutional layer, composed of significantly more filters than subsequent convolutional layers in the architecture. The wide convolutional layer at the stem of Inception architectures, followed by pooling immediately after, provides a large receptive field at the start of the architecture that is useful for capturing global context in finer detail for images.

4.6.3. Extended SpiralResNet AD Classification Results

Motivated by the Inception architectures [**186, 115, 187**], we replace the first convolutional ResBlock from our discriminative SpiralResNet architecture in Figure 4.5,

Table 4.3. Cortical extension results for HC vs. AD classification.

Structure	**Threshold = 0.5**			**AUC**	
	Precision	*Recall*	*F1*	*ROC-AUC*	*PR-AUC*
Full brain	**0.920**	**0.899**	**0.909**	**0.906**	**0.901**
Cortex	0.631	0.850	0.724	0.866	0.812
Cortex LH	0.702	0.794	0.745	0.864	0.824
Cortex RH	0.643	0.640	0.642	0.775	0.720

with a wide spiral GCN layer (with 128 filters) and a down-sampling layer (by a factor of 2) instead to mimic the wide convolutional stem at the start of the Inception network scheme. Using this approach improved the performance of the SpiralResNet for the AD vs. HC classification, outlined in Table 4.3.

Asymmetric thinning of the cerebral cortex in adults was shown to be accelerated for those living with AD [**188**], coinciding with whole-brain analyses by Wachinger *et al.* [**189**] who also report asymmetries in the hippocampus and amygdala for those with AD. Furthermore, the results of the ablation study in subsubsection 4.4.2.1 demonstrated that multi-regional input data improves classifier performance. This observed improvement in classification performance with the cortex supports this idea further.

4.6.4. Further Work

Extracting and analyzing the class activation map (CAM) for the best-performing SpiralNet model, incorporating cortical and subcortical information, is all that is left to do. This would be highly beneficial to observe the difference in how much subcortical information is weighed in on our model when cortical features are (not) available. Additionally, it would be highly beneficial to see if the CAMs on the cortex draw correspondences with the literature on patterns of cortical atrophy in AD pathology, particularly posterior cortical atrophy (PCA) [**190**].

CHAPTER 5

Discrete, recurrent, and scalable patterns in human judgement underlie affective picture ratings

Abstract

Operant keypress tasks show lawful relationships in human preference behavior (i.e., approach/avoidance) and have been analogized to “wanting”. It is unknown if non-operant rating tasks where each action does not have a consequence, analogous to “liking”, show similar lawful relationships. We studied non-operant, picture-rating data from three independent cohorts (N = 501, 506, and 4,019 participants) using the same 7-point Likert scale for negative to positive preferences. Non-operant picture ratings produced similar value, limit, and trade-off functions to those reported for operant keypress tasks, all with individual $R^2 > 0.80$. These functions were discrete in mathematical formulation, recurrent across all three independent cohorts, and scaled between individual and group curves. Behavioral features extracted from the non-operant, picture-rating task argue for lawfulness and demonstrate a simple, quick, and low-cost framework quantitatively assessing human preference without forced choice decisions, games of chance, or operant keypressing. This framework can be easily deployed on any digital device worldwide.

5.1. Introduction

Preference can be defined as the variable extent an organism shows an inclination to act or behave by approaching or avoiding events in the world, based on the rewarding or aversive effects of these events [**191, 192, 193**]. Preference-based behavioral variables that measure the intensity and patterns of approach/avoidance behavior with an operant keypress task based on reinforcement reward theory [**194, 195**] show lawful relationships in humans when using visual [**196, 197**] and auditory stimuli [**198**] (see Figure 5.1). These lawful behavioral relationships [**199**] have been associated with activation in brain reward circuitry by use of model-based functional MRI [**200, 201**], imaging genetics [**202, 203**], and quantitative morphometry [**204**].

The keypress task used in such studies was derived from an operant framework [**194, 195**] where each keypress had an incremental consequence on stimulus view time [**200, 197**]; this has been well-validated across multiple studies [**200, 205, 202, 196, 197, 206, 204, 203, 207, 201, 208, 209**]. The keypress task can be analogized to the construct of "wanting" as opposed to "liking" [**200, 210**], and leads to variables that quantify the average (mean) magnitude (K), variance (σ), and the pattern (i.e., Shannon entropy (H)) of participants' keypress-based behavior. We refer to this methodology, and the multiple relationships between these variables and features based on their graphical relationships, as relative preference theory (RPT) Figure 5.1. Two of the graphs produced for RPT mimic known functions with distinct variables from prospect theory [**211**] and the mean-variance function described by [**212**] for portfolio theory.

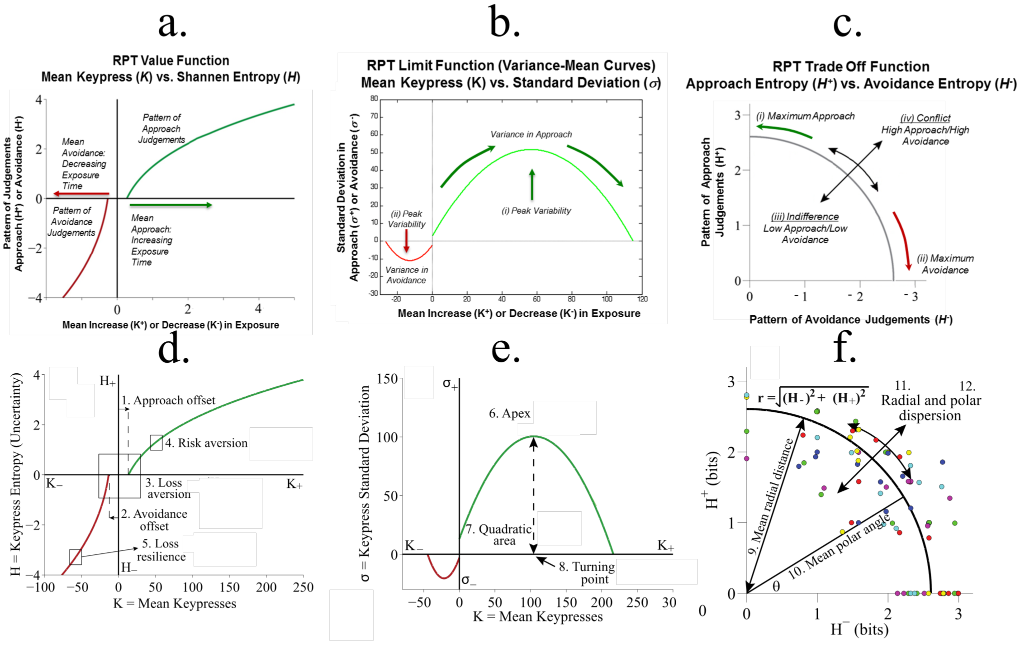


Figure 5.1. RPT curves and their behavioral interpretations.

RPT is characterized in part by features that describe relationships between these three behavioral variables: $\{K, H, \sigma\}$. These relationships include: (1) a value function plotting the Shannon entropy ($H_{\pm}$), against the average ratings ($K_{\pm}$) for approach or avoidance toward a suite of objects. This function is referred to as a value function given it calibrates “wanting” or “liking” (depending on the task structure) against the pattern of previous judgements and is consistent with the prospect theory value function. Standard features of these curves, shown in the diagram, include loss aversion (LA) and risk aversion (RA) from the literature on behavioral economics. The corollary of RA is also shown, herein referred to as loss resilience (LR). Two offsets are also noted that are clear in the individual data, relating to an "approach offset" (β_+) and "avoidance offset" (β_-). (2) A variance-mean relationship is observed between the average ratings ($K_\pm$) plotted against the corresponding standard deviation of rating responses ($\sigma_\pm$). This relationship is characterized by increasing variance up to a peak followed by decreasing variance back to baseline. This function describes limits to preference or its "saturation" (see Figure 5.1b). Standard features of this curve include the apices of the quadratic fits, the "turning points" ($\rho_\pm$) or value of $K_\pm$ at which $\sigma_\pm$ is maximal/minimal, and the quadratic areas ($QA_\pm$) of these curves bounded by the K-axis. (3) A trade-off function between the approach entropy (H_+) and avoidance entropy (H_-) was also identified, defining how bundles of approach judgments were balanced with bundles of avoidance judgments as a quantifiable trade-off between approach and avoidance (see Figure 5.1c). This trade-off function can be characterized by the mean polar angle of the trade-off curves (θ), the standard deviation of this polar angle (its dispersion) (σ_θ), the mean radial distance for the trade-off curves (r), and its corresponding standard deviation (the dispersion in r) (σ_r).

To date, RPT has only been discussed in an operant framework where effort traded for viewing time can be considered a model of "wanting" (e.g., [**200**]). In a recent study, RPT was compared to a prospect theory framework in which ratings were made under conditions of uncertainty during a game of chance (i.e., anticipation phase of the trial), and under conditions of certainty when the outcome was revealed (i.e., outcome phase of the trial). During anticipation, ratings produced statistically similar loss aversion (LA) measures to those of keypressing with an RPT analysis, whereas during the outcome phase, ratings showed no overweighting of losses relative to gains [**197**]. LA was specifically defined by [**213**] to describe an overweighting of negative judgements relative to positive ones under conditions of risk. These observations raised the hypothesis that in a non-operant model where actions have no consequences (i.e., "liking"), a rating task with no uncertainty might produce RPT-like curves, but not show the same degree of overweighting of losses relative to gains which characterize LA during uncertainty in prospect theory. Demonstrating that rating tasks show consistent law-like patterns, but a reduction in the overweighting of negative outcomes for LA, has potential implications for online digital behavior. Specifically, an absence of strong LA in the context of liking responses, but presence with wanting responses, provides a potential hypothesis for why digital behavior that is not effort or operant based, might reflect less concern for negative consequences.

These considerations led us to analyze three separate cohorts of human participants using a rating task and picture stimuli from the International Affective Picture Set (IAPS) [**214, 215**]. Three questions were asked: (1) Would ratings in the absence of the operant framework, produce preference relationships similar to what has been observed with keypressing (e.g., Figure 5.1)? (2) If ratings produced RPT-like curves, would these functions be (i) mathematically discrete, (ii) recurrent across cohorts, and (iii) scale from individual to group data? Namely, would functions from ratings meet three of the primary criteria raised by [**199**] for lawfulness? (3) How consistent would rating-based curves be to each other if they came from distinct cohorts and experimental sessions? Would features of these functions based on behavioral economics and prospect theory (i.e., LA and risk aversion (RA)) or based on Markowitz's decision utility around variance-mean functions [**212**] be similar between distinct cohorts? Furthermore, would the extracted behavioral features from these functions potentially differ from previously published ones computed from keypressing tasks?

To quantify the similarity of potential rating-based curves across distinct experiments, we framed 15 metrics that included LA, RA, and the equivalent of RA on the avoidance value function (Figure 5.1a). Twelve other metrics were defined from standard curve features such as offsets (Figure 5.1a), apices, x-axis values for the apices, and quadratic areas (Figure 5.1b), along with mean and variance measures for any angles or radial distances (Figure 5.1c) (see Methods). These 15 extracted "features" were not considered definitive for reconstituting each function but could be psychologically interpreted (see Methods).

From this work, we found that the broad set of features extracted from rating task curves in this study provide a potential framework for summarizing human reward and aversion judgements. This set of summary metrics, derived from a simple rating task on a digital device, could characterize human preference at the big-data scale, potentially across the 83.72% of the world's population that currently owns a smartphone [**10**], or the 85% of Americans with a smartphone (at least 97% own a cellphone of some kind) [**11**].

5.2. Methods

5.2.1. Participants

In all three studies, rating and survey responses were collected online to meet demographic criteria established by the United States (US) census. One study involved 501 participants for the Emotion and Behavior Study (EBS), an online study of US adult (i.e., $\geq$ 18 years of age) consumers, conducted by Research Results, Inc. (Boston, MA) in 2016. The second study consisted of 506 participants randomly sampled from the general US population using a participant database accessed by Gold Research Inc. (San Antonio, Texas) for the Automated Mental Health Assessment Study (AMHA), referred to as the AMHA-1 cohort. Questionnaire responses for AMHA-1 were collected between the end of February 2021 and the beginning of March 2021, approximately one year following the official COVID-19 pandemic declaration in the US. The third study involved 4,019 participants, also randomly sampled from the general US population using a participant database accessed by Gold Research Inc. for the AMHA study from November 2021, referred to as the AMHA-2 cohort. All participants provided informed consent for their response data to be used, and data released to Northwestern University from Gold Research Inc. and Research Results Inc. were anonymized.

Participant demographic information including: gender identity, age group (in years), employment status, education level, handedness, and race/ethnicity are summarized in Table 5.1, including percent compositions for each group within each corresponding demographic measure.

Table 5.1. Demographics of rating experiment participants across three distinct cohorts observed in this study.

		Cohort Counts (Percentage %)		
Demographic	**Group**	*EBS*	*AMHA-1*	*AMHA-2*
Gender identity	Male	225 (44.91)	262 (51.78)	1,592 (39.61)
	Female	276 (55.09)	243 (48.02)	2,408 (59.92)
	Other or prefer not to answer	0 (0.00)	1 (0.20)	19 (0.47)
Age group (years)	0-17	0 (0.00)	0 (0.00)	0 (0.00)
	18-24	79 (15.77)	67 (13.24)	245 (6.10)
	25-34	107 (21.36)	110 (21.74)	457 (11.37)
	35-44	90 (17.96)	128 (25.30)	603 (15.00)
	45-54	107 (21.36)	90 (17.79)	702 (17.47)
	55-64	102 (20.36)	73 (14.43)	977 (24.31)
	≥ 65	16 (3.19)	38 (7.51)	1,035 (25.75)
High education level	Some high school	19 (3.79)	12 (2.37)	112 (2.79)
	High school graduate	103 (20.56)	114 (22.53)	839 (20.88)
	Some college	201 (40.12)	114 (22.53)	1,172 (29.16)
	Bachelor's degree	106 (21.16)	114 (22.53)	912 (22.69)
	Some graduate school	9 (1.80)	23 (4.55)	200 (4.98)
	Graduate degree	40 (7.98)	54 (10.67)	666 (16.57)
	Post-doctoral training	23 (4.59)	75 (14.82)	118 (2.94)
Handedness	Right	423 (84.43)	393 (77.67)	3,446 (85.74)
	Left	70 (13.97)	80 (15.81)	478 (11.89)
	Both	8 (1.60)	33 (6.52)	95 (2.36)
Race ethnicity	White/Caucasian	353 (70.46)	355 (70.16)	3,340 (83.11)
	African American	80 (15.97)	61 (12.06)	277 (6.89)
	Hispanic/Latinx	20 (3.99)	34 (6.72)	139 (3.46)
	Asian or Pacific Islander	16 (3.19)	15 (2.96)	145 (3.61)
	Native American or Alaskan Native	6 (1.20)	7 (1.38)	28 (0.70)
	Mixed racial background	24 (4.79)	27 (5.34)	37 (0.92)
	Other race	1 (0.20)	4 (0.79)	21 (0.52)
	Prefer not to answer	1 (0.20)	3 (0.59)	32 (0.80)

Note: Demographic data for subject populations studied by rating task is shown per cohort. For each demographic, group counts and their relative percentages within the respective cohort are provided.

5.2.2. Picture stimuli

Stimulus sets across the rating task consisted of images from the International Affective Picture System (IAPS) [**214, 215**], a well-validated emotional stimulus set. For all cohorts, six categories of pictures were used: (1) sports, (2) disasters, (3) cute animals, (4) aggressive animals, (5) nature (beach vs. mountains), and (6) food, with eight pictures per category (48 pictures in total). Pictures had a maximum size of $1,204 \times 768$ pixels in all three studies. All picture stimulus sets reported in the present study are collectively referred to as "IAPS stimuli" throughout the text.

5.2.3. Picture rating task ("liking" assessment)

Participants were prompted for the rating task while completing an online digital survey, which contained questionnaires regarding participant demographic information and research questionnaires for depression symptoms using the Patient Health Questionnaire (PHQ-9) [**216**]; trait anxiety using the Spielberger State-Trait Anxiety Inventory (STAI) [**217**]; a broad array of mental health, neurological, and medical issues using the MGH Phenotype Genotype Project in Addiction and Mood Disorders symptom questionnaire (MGH-SQ); and behavioral health disorders (e.g., internalizing or externalizing psychiatric disorders, substance use disorders, or crime/violence problems) from the GAIN-SS short screen assessment [**218**]. For the picture rating task, the instructions presented to participants for each study were based on the following instructions used for the EBS study:

> “The next part of this survey involves looking at pictures and then responding how much you like or dislike the image. Please rate each image on a scale from -3 (Dislike Very Much) to +3 (Like Very Much). Zero (0) is neutral... meaning you have no feelings either way. The images are a set of photographs that have been used by scientists around the world for over 20 years.
>
> It is important you rate each picture based on your initial emotional response.
>
> There are no right or wrong answers... just respond with your feelings and rate the pictures very quickly. Please click ‘Next’ to begin.”

Each picture was presented as shown in Figure 5.2, where the ratings below each picture were selectable using the mouse cursor or keyboard arrows. There was no time limit for assigning ratings to each picture, but participants were requested to rate each picture as quickly as possible, and they were not able to change their response after selecting a rating. After each rating selection was made, the next picture was automatically loaded and presented.

5.2.4. Data quality screening

Data integrity was assessed for all data from the three studies. Quality assurance was conducted based on four exclusion criteria for picture rating tasks and survey data (survey-based non-rating data is not described herein), which reduced the analysis to 281 participants for the EBS cohort, 366 for AMHA-1, and 3,476 for AMHA-2. These four exclusion criteria were:

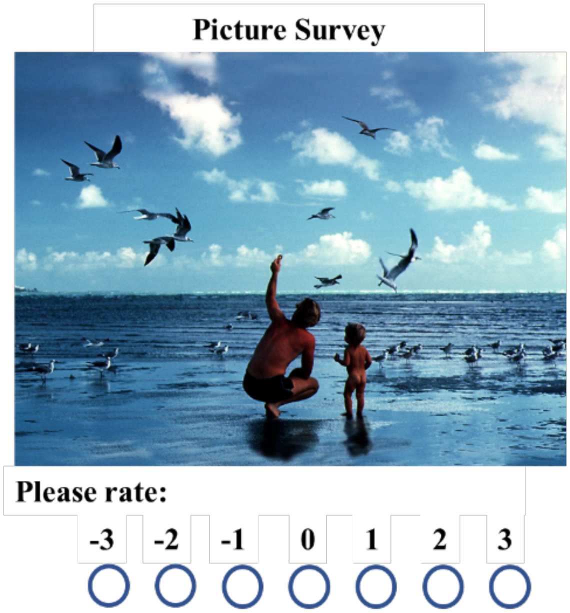


Figure 5.2. Example of the format of the picture rating task. Unlike the keypress task from prior studies, this task involved no operant consequence to its action. Individuals made ratings with no change in viewing time or other consequence, rating along a 7-point Likert-like scale from -3 to +3.

(1) participants selected the same response throughout any section of the questions/tasks (e.g., selecting option "1" for all questions),

(2) participants indicated they had ten or more clinician-diagnosed illnesses (data not described here),

(3) participants showed minimal variance in a picture rating task (i.e., all pictures were rated the same or varied only by one point), and

(4) if *both* education level and years of education did not match *and* if they completed the questionnaire in less than 500 seconds (800 seconds for the AMHA-2 cohort).

Further quality assurance involved assessment of RPT variables and curves from the picture rating tasks. Variables that were quantified included the average magnitude (K), variance (σ), and the pattern or information (i.e., Shannon entropy (H)) related to participants' preference behavior. K reflected the average (mean) of positive ratings a subject made (K_+) or negative ratings (K_-) within each picture category. Other metrics included the variance in positive ratings (σ_+) or negative ratings (σ_-), along with the Shannon entropy (i.e., information; see [**219**]) of positive ratings (H_+) or negative ratings (H_-) for stimuli within each category. The Shannon entropy is a core variable in information theory that characterizes the degree of uncertainty across a set of responses [**219**]; it quantifies the pattern of judgements made to a set of stimuli and could thus be considered a memory variable. Collectively, these variables capture judgments about the valence of judgement (positive vs. negative or approach vs. avoidance) as well as its magnitude (intensity of rating) to describe relative preferences [**220, 196**] (Figure 5.1).

When evaluating data quality, raw data was assessed for cases when $K = 0$ for a given category (i.e., cases where the subject made all neutral ratings to neither approach nor to avoid any stimulus in the category). Computing the Shannon entropy, H, for a given picture category requires that $K > 0$ given that when $K = 0$, the H computation results in evaluating $\log\left(\frac{0}{0}\right)$ which is undefined. In these cases, the Shannon entropy was set to $H = 0$ for categories in which the subject rated “0” for all the stimuli.

Before carrying out the RPT analyses, and fitting models to participants' ratings, data was further screened for additional criteria beyond when $K = 0$ for a given category. The complete set of model fit inclusion/exclusion criteria was as follows:

(1) Valid entropy (H) calculations (see prior paragraph),

(2) exclusion of data points lying beyond three times the interquartile range (IQR), below the first quartile or above the third quartile (i.e., removing extreme outliers),

(3) and coherence of model fits between individual and group data. This last criterion required that the curve direction for individual subject fits be consistent with the curve direction of the group-level statistical fits (and boundary envelopes), and therefore corroborate most of the observed subject data.

Criteria (3) and (4) are necessary operational definitions for quality assurance given the potential for convergence failures with curve fitting. For the Automated Mental Health Assessment Study (AMHA)-2 cohort, criteria (2) was not implemented given the potential for greater variance than with the prior two studies due to the COVID-19 pandemic; in lieu of a 3×interquartile range (IQR) threshold, a threshold was set for the two curve features with a small number of very extreme outliers: loss aversion ($LA > 200$, resulting in $N = 42$ exclusions) and positive quadratic area ($QA_+ > 100$, resulting in $N = 5$ exclusions) (see definitions in Relative preference analysis next).

In total, six types of model fitting were performed for the rating data: group and individual models for the (K, H) data, (K, σ) data, and (H_-, H_+) data distributions. For the group data, we generated group-level data fits along with boundary envelopes (power-law fits and logarithmic fits for group (K, H) data), and quadratic fits for group (K, σ) data to guide the focus of statistical testing based on the power-law fits (K, H), and quadratic fits (K, σ) for individual data. Individual data then followed these fits based on logarithmic and simple power-law fits for individual (K, H) value functions, quadratic fits for individual (K, σ) limit functions, and radial fits for individual (H_-, H_+) trade-off distributions [**196, 198**].

5.2.5. Relative preference analysis

Initial analysis of picture rating data involved qualitative assessment of the mean positive and negative ratings for each category of picture to confirm there were no major deviations in IAPS stimuli ratings across the three studies.

For the relative preference analysis, we replicated the methodology described in detail by [**196, 198, 208**]. We used the iterative modeling approach of [**221**] to identify RPT patterns in the data and three of the four putative signatures of lawfulness, as described previously with visual [**220, 196**] and auditory stimuli [**198**]. We thus sought "discrete" mathematical fitting of patterns within the data, "recurrence" of patterns across the three distinct experimental cohorts, and "scalability" of the observed patterns. We utilized datasets that met stringent criteria for quality assurance, then assessed the graphical structure between the three behavioral variables, $\{K, H, \sigma\}$. For the rating tasks, these variables reflected the mean positive ratings or negative ratings within a picture category $(K_\pm)$, the Shannon entropy of positive/negative ratings within a category $(H_\pm)$, and the standard deviation $(\sigma_\pm)$ of positive/negative ratings within a category. Graphical analyses sought to determine the presence of functions, manifolds, or boundary envelopes to individual, and separately, group data that were graphically similar to RPT functions, manifolds, and boundary envelopes [**220, 196, 197, 198**].

Formal testing of discreteness, recurrence, and scaling was done as follows. To assess if mathematical fitting was discrete, the goodness of fit for the (K, H) value functions and (K, σ) limit functions, across the three experiments, were characterized by R^2, and adjusted R^2 statistics; then tabulated by location and dispersion estimates. Given prior keypress findings of discreteness with $R^2 > 0.7$, we assessed if definable functions for individual data and manifold fits (and/or boundary envelopes) for group data had clear parameter estimates and showed $R^2 > 0.7$. For recurrence, we assessed if similar individual and group models were observed for each of the three independent populations, and if the extracted RPT features ($N = 15$) for individual functions were similar across the three groups. Lastly, scale invariance and simple power-law fitting was assessed by performing linear regressions following logarithmic transformations of both the K- and H-axes. If the resulting fits characteristically demonstrated asymptotic behavior ($0 < a < 1$, given $H(K) = bK^a$), this implied that substantial changes in the input variable, K, produced only minor changes in the output, H. The same asymptotic behavior was assessed with the logarithmic fits to the (K, H) data, with the difference that in this case the fits were obtained by performing a linear regression of H against K after the logarithmic transformation of K alone.

5.2.5.1. (K, H) value functions. We evaluated mean positive or negative ratings across stimuli within a picture category ($K_{\pm}$) and the Shannon entropy of these ratings ($H_{\pm}$). We used the following approach to compute the Shannon entropy separately for the positive (approach) and negative (avoidance) ratings in each category. First, consider an ensemble of numbers for either positive or negative rating responses, $\mathbf{a}$, across stimuli within a single picture category: $\mathbf{a}_{\pm} = \{a_1, a_2, \ldots, a_N\}$, for N pictures within the given category. We can then define the relative proportions of the positive and negative responses for the individual stimuli, p_i, such that

$$p_i = \frac{a_i}{\sum_{j=1}^{N} a_j}. \tag{5.1}$$

Using these normalized proportions of the rating responses, the Shannon entropy of the response pattern can be computed for an individual picture category as follows:

$$H_{\pm} = \sum_i p_i \log_2 \left(\frac{1}{p_i} \right). \tag{5.2}$$

After computing the values of $K_{\pm}$ and $H_{\pm}$ for each picture category, (K, H) value functions were generated by plotting the Shannon entropy, $H_{\pm}$, against the mean ratings ($K_{\pm}$), for all picture categories for an individual subject. (K, H) data were also plotted across multiple participants to visualize data at the group level.

At the group level, we assessed if (K, H) best-fit parameters could be approximated using the logarithmic function, $H(K) = a \log_{10}(K) + b$, or power-law functions, $H(K) = bK^a$; we also confirmed they contained boundary envelopes that conformed well to either logarithmic functions of power-law functions. At the individual subject level, we assessed fits for the same logarithmic and power-law functions to the (K, H) data for approach and avoidance across picture categories for individual participants. The best-fit parameters for the logarithmic and power-law functions were achieved by performing a simple linear regression on the plots for H vs. $\log_{10}(K)$, and $\log_{10}(H)$ vs. $\log_{10}(K)$, respectively.

5.2.5.2. (K, H) limit functions. The second relationship considered was that between the mean ratings, $K_{\pm}$, and the standard deviation of ratings across stimuli within a category, $\sigma_{\pm}$. (K, σ) limit functions were generated by plotting values of σ against K for all picture categories in an individual subject or by pooling the data together across participants in a group analysis. At both the individual and group level, we found that (K, σ) limit functions were well characterized by quadratic functions of the form $\sigma = aK^2 + bK + c$. For the group data, we fit quadratic boundary envelopes to the (K, σ) data much in the same manner performed for the (K, H) value functions. For individual subject analysis, we fit quadratic functions directly to the (K, σ) data using the `polyfit()` function in MATLAB$^{®}$.

5.2.5.3. (H_-, H_+) trade-off plots. (H_-, H_+) trade-off (or opponency) plots were defined by plotting the Shannon entropy for positive ratings, H_+, against the Shannon entropy for negative ratings, H_-, for all picture categories in each stimulus set. These plots were generated either across categories for an individual subject, or by pooling data across all participants in the cohort to generate a group-level plot. For both the individual subject- and group-level data, (H_-, H_+) data conformed to a radial distribution about the origin of the trade-off plot, such that $r = \sqrt{(H_+)^2 + (H_-)^2}$, or equivalently, $H_+ = \sqrt{r^2 - (H_-)^2}$. Radial fits were estimated for individual participants as well as the group-level data by computing the mean radial distance, r, across all (H_-, H_+) data in the trade-off plot.

5.2.5.4. Feature extractions from (K, H), (K, σ), and (H_-, H_+) functions. To help characterize the (K, H), (K, σ), and (H_-, H_+) functions, we applied two standard definitions from behavioral economics (LA and RA) along with the RA computation applied to the avoidance arm of the value function (Figure 5.1a), referred to as loss resilience (LR) herein. Twelve other features that reflect standard curve feature analyses were utilized (Figure 5.1): positive offset (β_+), negative offset (β_-), positive apex (α_+), negative apex (α_-), positive turning point (ρ_+), negative turning point (ρ_-), mean polar angle of the (H_-, H_+) curve (θ), standard deviation of the polar angle (σ_θ), mean radial distance of points on the (H_-, H_+) plot (r), and the standard deviation of the radial distances (σ_r). These simple metrics are not exhaustive but allow interpretation of the functions based on RPT, prospect theory, and Markowitz's decision utility [**220, 211, 196, 197, 198, 212, 213, 208**]. Descriptions of these 15 curve features are provided briefly in what follows.

(K, H) extracted feature definitions

Features for the (K, H) plots were framed by the (K, H) function being considered concave relative to the K-axis. The RPT features that were extracted from these graphs were: risk aversion (RA), loss resilience (LR), loss aversion (LA), and the positive and negative offsets ($\beta_\pm$).

- Risk aversion (RA): risk aversion is extracted as the ratio of the second derivative of the (K_+, H_+) curve to its first derivative, which also produces a curve. To produce a unitary value for comparison across cohorts, we calculated RA for $K_+ = 1.5$. Informally, RA measures the degree to which an individual prefers a likely reward in comparison to a better more uncertain reward. RA is a common notion in economics that studies decision-making under uncertainty [**222**].
- Loss resilience (LR): loss resilience is defined to be the absolute value of the ratio of the second derivative of the (K_-, H_-) curve to its first derivative, which also produces a curve. For prediction, we calculated LR at $K_- = -1.5$. Informally, LR is the degree to which an individual prefers to lose a small defined amount in comparison to losing a greater amount with more uncertainty associated with this loss.
- Loss aversion (LA): loss aversion is the absolute value of the ratio of the linear regression slope of $(\log K_-, \log H_-)$ to the linear regression slope of $(\log K_+, \log H_+)$. It intuitively measures the degree to which an individual person outweighs losses to gains. LA is a fundamental measure in prospect theory [**211**], which informally states that humans have a cognitive bias to overweight losses relative to gains in the presence of uncertainty.

- Positive offset (β_+): the positive offset is the value of K_+ when setting $H_+ = 0$. β_+ intuitively measures the ante one needs to engage in a game of chance and models the amount of a bid an individual is willing to make to enter a game of chance (e.g., an "ante" in poker).
- Negative offset (β_-): the negative offset is the value of K_- when setting $H_- = 0$. β_- intuitively measures how much insurance an individual might need against bad outcomes. It mirrors the "ante", but in the framework of potential losses.

(K, σ) extracted feature definitions

Features for the (K, σ) curves, were framed by the (K, σ) curves being considered as quadratic functions, that were concave relative to the K-axis. The (K, σ) curve models the relationship between variance (risk) and mean value. It can also be framed by the following question: Would an individual prefer a dollar with probability one, or value drawn from a normal distribution with a mean of two and variance of two? The RPT features that are extracted from this curve include: the positive and negative apices ($\alpha_\pm$), the positive and negative turning points ($\rho_\pm$), and the positive and negative quadratic areas ($QA_\pm$).

- Positive apex (α_+): the positive apex is the value of σ_+ for the derivative $\frac{d\sigma_+}{dK_+} = 0$. Intuitively, this represents the maximum variance for approach behavior. In this sense, α_+ models where increases in positive value transition from a relationship with increases in risk, to a relationship with decreases in risk. **[212]** described decision utility similarly, so that the positive apex models when variance changes from weighing against a decision to facilitating a decision.

- Negative apex (α_-): the negative apex is the value of σ_- when the derivative $\frac{d\sigma_-}{dK_-} = 0$. Intuitively, this represents the maximum variance for avoidance behavior. Like with the positive apex, this transition point is important to consider for avoidance decisions in the context of decision utility by [**212**].
- Positive turning point (ρ_+): the positive turning point is the value of K_+ when the derivative $\frac{d\sigma_+}{dK_+} = 0$. Intuitively, this represents the rating intensity with maximum variance for approach behavior, potentially when an individual decides to approach a goal-object.
- Negative turning point (ρ_-): the negative turning point is the value of K_- when the derivative $\frac{d\sigma_-}{dK_-} = 0$. Intuitively, this represents the rating intensity with maximum variance for avoidance behavior, potentially when an individual decides to avoid a goal-object.
- Positive quadratic area (QA_+): the positive quadratic area is the area under the curve (AUC) of the first quadrant of the (K_+, σ_+). This variable represents the relationship between K_+ and σ_+ and can be thought of as a quantity that measures the amount of value an individual associates to positive stimuli.
- Negative quadratic area (QA_-): the negative quadratic area is the AUC of the third quadrant for the curve (K_-, σ_-). This variable represents the

relationship between K_- and σ_- and can be thought of as quantity that measures the aversive value an individual associates to negative stimuli.

(H_-, H_+) extracted feature definitions

Features for the (H_-, H_+) curve were framed as the (H_-, H_+) function being considered a trade-off function between the H_- and H_+ variables, that can commonly look like a semi-circular fit in individuals (e.g., Figure 5.1c). The RPT features extracted from this curve include: the mean polar angle (θ), its standard deviation (σ_θ), mean radial distance (r), and its corresponding standard deviation (σ_r).

- Mean polar angle (θ): the mean polar angle is the mean of the polar angles of the points in the (H_-, H_+) plane. Intuitively, this measures the mean balance for the entropy, or patterns, in approach vs. avoidance behavior. It signifies the balance in approach and avoidance judgments across multiple categories of goal-object (e.g., picture ensembles in this case).
- Polar angle standard deviation (σ_θ): the standard deviation of the polar angles of the points in the (H_-, H_+) plane. Intuitively, this measures the standard deviation in the patterns of approach and avoidance behavior. This variance represents the spread of positive and negative preferences across a set of potential goal-objects and can be considered a measure of the breadth of an individual's (or a group's) portfolio of preference.
- Mean radial distance (r): the mean radial distance measures the average Euclidean distance of the data points in the (H_-, H_+) curve to the origin. This measure defines how individuals can have strong positive and negative

preferences (i.e., biases) for the same thing, reflecting conflict, or have low positive and negative preferences for something, reflecting indifference. This gets at the consistency of compatibility of approach and avoidance, and how an individual can both like and dislike something or be indifferent to both its positive and negative features.

- Radial distance standard deviation (σ_r): this is simply the standard deviation of the radial distances of the data points in the (H_-, H_+) plane to the origin. This final measure is interpreted through how the points in the (H_-, H_+) plane vary regarding the radial distance from the origin. The variance in this radial distance will reflect how much an individual goes between having conflicting preferences and having indifferent ones.

5.2.6. Comparison of features between rating experiments

For each of the three subject populations, the mean and standard deviation (SD) were computed for each of the fifteen features, along with standard error of the mean (SEM) and the 95% confidence intervals (CIs) for the corresponding means. Violin plots [**2**] for each of the RPT features were also generated (Figure 5.5) to provide a visual comparison of the distribution, interquartile range (IQR), and 95% CIs, with respect to the corresponding median, for each RPT feature across all cohorts. The primary framework of comparison was assessment of overlap in the 95% CI for the corresponding means, and violin plots for the medians. A quantitative comparison of

RPT features across the three cohorts was also performed using rank-based, nonparametric Kruskal-Wallis H-test [**184, 223**] statistics, followed by post-hoc nonparametric pairwise multiple comparisons using Dunn's test [**224**] and Kolmogorov-Smirnov (K-S) test [**225**] statistics. This was done for all fifteen features, although only seven of these features reflected dimensionless units, and covariances around demographic differences in the cohorts could not be incorporated into such analyses. Given this last factor, we assessed age distributions for each sample, where clear skewing existed, and ran univariate linear regressions for the fifteen features against age in that cohort to assist with interpreting results.

5.3. Results

5.3.1. Group-level assessment of positive and negative ratings by picture category

For each of the three groups of participants studied, we summed the total number of positive and negative ratings made per picture category, and their mean, as a qualitative assessment that there were no major deviations in IAPS stimuli ratings across the three studies. As can be seen in Table 5.2, the mean positive and negative ratings for each category of picture were in close alignment across the EBS, AMHA-1 and AMHA-2 groups.

Table 5.2. Picture ratings summary statistics across cohorts

	Sum of ratings (Mean rating per picture)					
	Avoidance (negative ratings)			*Approach (positive ratings)*		
Category	*EBS*	*AMHA-1*	*AMHA-2*	*EBS*	*AMHA-1*	*AMHA-2*
Aggressive animals	-6,748 (-2.3354)	-4,784 (-2.3719)	-65,383 (-2.5629)	1,102 (1.7225)	3,024 (1.9194)	6,686 (1.9861)
Nature	-352 (-1.7325)	-466 (-1.6944)	-3,013 (-1.8483)	7,460 (2.2487)	7,527 (2.2516)	60,939 (2.3263)
Cute animals	-396 (-1.7071)	-456 (-1.7757)	-3,723 (-1.8818)	8,058 (2.3523)	8,111 (2.3284)	66,348 (2.4463)
Disaster	-7,203 (-2.4161)	-5,255 (-2.4392)	-68,694 (-2.5960)	940 (1.7608)	2,904 (1.9926)	5,117 (1.8846)
Nude/exposed bodies	-3,092 (-2.1251)	-1,697 (-2.1025)	-23,676 (-2.2657)	3,529 (2.1478)	5,227 (2.0948)	26,998 (2.1937)
Sports	-1,700 (-1.8305)	-1,298 (-1.8180)	-13,971 (-1.9396)	3,657 (1.7883)	5,248 (1.9844)	28,672 (1.9020)

Legend: Summary and descriptive statistics for the "total sum responses across the cohort" and "mean response", stratified by positive or negative valence of ratings (approach, +, or avoidance,−). The total sum of approach (+) and avoidance (−) responses is shown per cohort for each category of picture shown from the IAPS stimulus set, along with the average response for each picture, per category.

5.3.2. Group-level (K, H), (K, σ), and (H_-, H_+) analyses

We first investigated the relationships between mean ratings and the Shannon entropy of category distributions for ratings. Group-level analyses were performed in two ways:

(1) Envelope fits as done previously (e.g., [**196, 198**]), and
(2) statistical fits of group data to constrain the fits tested subsequently with individual data (Table 5.3).

For envelope fitting of the value function, power-law and logarithmic boundary envelopes were fit to the approach (K_+, H_+) and avoidance (K_-, H_-) rating data such that they formed an outer bound containing 95% of the data; they were both observed to provide robust approximations of the edge of the distribution. For all three experiments, group data was fit by boundary envelopes to similar extents (all $p < 0.05$) by both logarithmic and power-law functions. When we examined functional fitting of the group data, we observed statistically significant fits (all $p < 0.05$) by both logarithmic and power-law functions (Table 5.3a-c). R^2 values were all > 0.70 and extended up to 0.90 across the three cohorts.

Next, we examined the relationship between the average category ratings and the standard deviation of ratings in each category, which we refer to as the mean-variance relationship (Figure 5.1b). Boundary envelopes enclosing 95% of the approach (K_+, σ_+) and avoidance (K_-, σ_-) data were fit to the EBS data. The quadratic boundary envelopes effectively approximated the edge of the mean-variance plots. The same analytic approach was performed for AMHA-1 and AMHA-2 data,

Table 5.3. Group fitting parameters for IAPS rating experiment across three cohorts

		Cohort					
		EBS		*AMHA-1*		*AMHA-2*	
Curve set	**Curve**	R^2	**Parameters**	R^2	**Parameters**	R^2	**Parameters**
(K, H) (power law)	(K_+, H_+)	0.732	$a = 0.440$ $b = 0.296$	0.784	$a = 0.475$ $b = 0.287$	0.767	$a = 0.461$ $b = 0.282$
	(K_-, H_-)	0.766	$a = 0.473$ $b = 0.274$	0.809	$a = 0.502$ $b = 0.261$	0.827	$a = 0.510$ $b = 0.250$
$(\log(K), H)$ (logarithmic)	$(\log(K_+), H_+)$	0.857	$a = 2.210$ $b = 2.047$	0.890	$a = 2.316$ $b = 2.013$	0.875	$a = 2.304$ $b = 1.967$
	$(\log(K_-), H_-)$	0.871	$a = 2.330$ $b = 1.932$	0.892	$a = 2.324$ $b = 1.905$	0.902	$a = 2.394$ $b = 1.851$
(K, σ) (quadratic)	(K_+, σ_+)	0.750	$a = -0.476$ $b = 1.468$ $c = 0.067$	0.746	$a = -0.470$ $b = 1.439$ $c = 0.091$	0.812	$a = -0.518$ $b = 1.588$ $c = 0.068$
	(K_-, σ_-)	0.817	$a = -0.553$ $b = 1.667$ $c = 0.065$	0.862	$a = -0.580$ $b = 1.754$ $c = 0.055$	0.864	$a = -0.609$ $b = 1.827$ $c = 0.069$

Legend: Group power law, (K, H), and logarithmic, $(\log(K), H)$ fits along with quadratic, (K, σ), fits are listed for the rating data across the three cohorts, stratified by valence of ratings (approach, +; or avoidance, −). For each fit, the coefficient of determination (goodness of fit), R^2, and the corresponding fitting parameters of the group approach (+) and avoidance (−) curves is reported per cohort.

showing outer bounds containing 95% of the data, providing robust approximations of the edge of the distribution, and broadly corroborating the behavior of analogous distributions for reported keypress-based RPT variables [**196, 198**], with $p < 0.05$ for all three datasets. When we examined functional fitting of the group data, we observed statistically significant fits (all $p < 0.05$) for all three cohorts (Table 5.3a-c). Importantly, all curves depicting limit functions with group data showed concave fits (relative to the absolute value of the $\{K, H, \sigma\}$ variables), thereby setting a constraint used for the individual data. R^2 values for group fitting varied from 0.75 to 0.86 across the three cohorts (Table 5.3a-c).

Lastly, we examined the (H_-, H_+) trade-off distributions characterizing the relationship between the patterns of approach and avoidance across tasks. This analysis sought to assess whether the pattern of approach preference behavior (i.e., positive ratings) scaled in proportion to the avoidance preference behavior (i.e., negative ratings) for pictures within the same categories. Specifically, we fit radial functions to test for symmetry in the distribution of H_- and H_+ values across categories within each individual subject. Ratings-based group-level (H_-, H_+) distributions were broader than distributions we have reported previously for keypress-based implementations of the IAPS relative preference task, consistent with the increased variance in the rating graph features for mean polar angle and mean radial distance.

5.3.3. Individual subject (K, H) value functions

After investigating the shape of distributions at the group level, we assessed (K, H) data and value function fitting at the individual subject level. As observed for group data, individual participants' (K, H) value functions for the rating were well fit by concave logarithmic or power-law functions (Table 5.4, Figure 5.4a-5.4c). Goodness of fit was assessed by computing R^2 values, and adjusted R^2 values (accounting for degrees of freedom) for each subject's model fit (Table 5.4). R^2 values were all > 0.80 and ranged from 0.84 to 0.96. Average goodness of fit values were quite similar between the three cohorts (Table 5.4).

Overall, fewer participant exclusions were noted across the three studies for logarithmic fitting of the (K, H) approach data when compared to (K, H) power-law model fits due to either insufficient data for fitting of joint RPT distributions or invalid parameter estimates (see Methods of this chapter).

5.3.4. Individual subject (K, σ) limit functions

The consistency of mean-variance relationships across participants in the rating experiment was assessed by fitting quadratic functions to each individual subject's (K, σ) distributions for approach and avoidance rating data. Concave quadratic fits across individual participants' (K, σ) data are displayed in Figure 5.4a-5.4c. Following Figure 5.4a-5.4c, **(A)** is the plotted value functions comparing mean rating intensity (K) to rating entropy (H) in individual participants, where K and H values were computed for each of six picture categories, for either approach (K_+, H_+) or

Table 5.4. Goodness of fit summary statistics for individual value and limit functions

Curve set	Curve	Summary statistic	Mean ± Std. dev. (per cohort)		
			EBS	*AMHA-1*	*AMHA-2*
$(\log(K), H)$	$(\log(K_+), H_+)$	R^2	0.89 ± 0.14	0.89 ± 0.16	0.92 ± 0.13
		R^2_{adj}	0.85 ± 0.21	0.85 ± 0.23	0.87 ± 0.21
	$(\log(K_-), H_-)$	R^2	0.94 ± 0.11	0.92 ± 0.12	0.96 ± 0.07
		R^2_{adj}	0.90 ± 0.18	0.88 ± 0.21	0.80 ± 0.29
$(\log(K), \log(H))$	$(\log(K_+), \log(H_+))$	R^2	0.84 ± 0.20	0.86 ± 0.17	0.89 ± 0.17
		R^2_{adj}	0.76 ± 0.32	0.80 ± 0.26	0.80 ± 0.29
	$(\log(K_-), H_-)$	R^2	0.92 ± 0.12	0.91 ± 0.14	0.95 ± 0.10
		R^2_{adj}	0.87 ± 0.21	0.85 ± 0.24	0.90 ± 0.20
(K, σ)	(K_+, σ_+)	R^2	0.87 ± 0.18	0.85 ± 0.19	0.90 ± 0.13
		R^2_{adj}	0.78 ± 0.30	0.75 ± 0.31	0.84 ± 0.21
	(K_-, σ_-)	R^2	0.92 ± 0.12	0.91 ± 0.14	0.95 ± 0.10
		R^2_{adj}	0.87 ± 0.21	0.85 ± 0.24	0.90 ± 0.20

Legend: Individual logarithmic, $(\log(K), H)$, and linear $(\log(K), \log(H))$ fits, along with quadratic, (K, σ), fits are listed for the rating data across the three cohorts. Linear, logarithmic, and quadratic correlations were performed in each subject across the data relating to approach ratings for the six categories of IAPS stimuli, and across the data relating to avoidance responses; participants needed data from at least three of the experimental conditions (aggressive animals, nature, cute animals, disaster, nudity, and sports) to be fitted. The mean and standard deviation are listed for the coefficient of determination, R^2 and its corresponding adjusted value, $R^2_{\text{adj.}}$.

avoidance (K_{-}, H_{-}) rating behavior within a single representative subject. The dark green and red traces indicate power-law fits to approach and avoidance data for each subject. Additionally, **(B)** looks at the limit functions comparing K to the standard deviation of approach or avoidance ratings (σ) across picture categories in individual participants. Approach and avoidance data for individual participants were fit to quadratic functions (see Methods of this chapter). Lastly, **(C)** is the trade-off plot comparing entropy for approach (H_{+}) and avoidance (H_{-}) ratings across six picture categories in individual participants. The dotted black line denotes $r = \log_2(8)$ and each subject is shown as a radial fit where $r = \sqrt{(H_{+})^2 + (H_{-})^2}$.

As with the (K, H) data, the goodness of fit was assessed by computing R^2 values, and adjusted R^2 values (accounting for degrees of freedom) for each subject's model fit (Table 5.4). R^2 values varied from 0.85 to 0.94, which was considered very high across participants.

5.3.5. Individual subject (H_{-}, H_{+}) trade-off functions

Trade-off distributions (H_{-}, H_{+}) for individual participants' rating patterns across pictures were also examined. This analysis sought to assess whether the pattern of participants' approach preference behavior (i.e., positive ratings) scaled in proportion to the pattern of participants' avoidance preference behavior (i.e., negative ratings) for pictures within the same categories (e.g., nature scenes). Specifically, we fit radial functions to test for trade-offs in the distribution of H_{-} and H_{+} values across categories within each individual subject. Figure 5.4a-5.4c display radial fits across individual participants' (H_{-}, H_{+}) data and highlight the (H_{-}, H_{+}) data points and fit for a representative subject from each experiment.

5.3.6. Extracted curve features computation and comparison of rating results

Summary statistics for LA, RA, and the 13 other RPT graph features obtained from each participant in each experiment are summarized in Table 5.5. Significantly, there was a consistent overlap among the 95% CIs for the median in the majority of the RPT features (Figure 5.5). Kernel density estimates show the shapes of individual distributions for each cohort, while the box plots within each violin plot describes the median and the corresponding IQRs and 95% CIs for the median. Unlike violin plots, box plots don't allow us to see variations in the data, particularly for multimodal distributions (those with multiple peaks). As is clear in Figure 5.5, the majority of RPT features' distributions showed significant deviation from normality in each cohort, so further analyses used nonparametric statistics, which are tabulated in Table 5.6 and Table 5.7. LA values were lower in all three cohorts than has been reported for keypress experiments and prospect theory-based experiments from [**226, 213**]. It should be noted that $LA < 2.0$ for the rating experiments suggests that participants did not show LA per se, but potential reward sensitivity. LA values did differ between the three cohorts by the Kruskal-Wallis H-test, while the post-hoc

Dunn's assessment confirmed that this difference only occurred between the AMHA-1 and AMHA-2 cohorts. The additional post-hoc two-sample K-S test assessment concluded that the distributions for LA across the three cohorts did not differ.

Table 5.5. RPT curve metrics for IAPS rating experiments across cohorts

		Cohort		
RPT metric	**Statistic**	*EBS*	*AMHA-1*	*AMHA-2*
Loss aversion (LA)	*Mean ± SD*	0.88 ± 1.148	1.51 ± 2.090	0.88 ± 0.365
	SEM	0.0738	0.1647	0.0063
	95% CI	$[0.74, 1.03]$	$[1.19, 1.84]$	$[0.86, 0.89]$
Risk aversion (RA)	*Mean ± SD*	0.35 ± 0.122	0.32 ± 0.120	0.34 ± 0.125
	SEM	0.0072	0.0090	0.0022
	95% CI	$[0.33, 0.36]$	$[0.31, 0.34]$	$[0.34, 0.35]$
Loss resilience (LR)	*Mean ± SD*	0.32 ± 0.122	0.30 ± 0.130	0.32 ± 0.134
	SEM	0.0073	0.0098	0.0024
	95% CI	$[0.29, 0.32]$	$[0.28, 0.31]$	$[0.32, 0.33]$
Positive offset (β_+)*	*Mean ± SD*	0.15 ± 0.101	0.16 ± 0.097	0.17 ± 0.106
	SEM	0.0046	0.0052	0.0018
	95% CI	$[0.14, 0.16]$	$[0.15, 0.17]$	$[0.17, 0.18]$
Negative offset (β_-)*	*Mean ± SD*	-0.19 ± 0.101	-0.19 ± 0.101	-0.21 ± 0.108
	SEM	0.0048	0.0061	0.0018
	95% CI	$[-0.20, -0.18]$	$[-0.20, -0.18]$	$[-0.21, -0.21]$
Positive apex (α_+)*	*Mean ± SD*	1.27 ± 0.327	1.27 ± 0.274	1.33 ± 0.290
	SEM	0.0154	0.0152	0.0049
	95% CI	$[1.24, 1.30]$	$[1.24, 1.30]$	$[1.32, 1.34]$
Negative apex (α_-)*	*Mean ± SD*	1.39 ± 0.378	1.40 ± 0.382	1.52 ± 0.349
	SEM	0.0174	0.0216	0.0059
	95% CI	$[1.35, 1.42]$	$[1.35, 1.44]$	$[1.51, 1.53]$
Positive turning point (ρ_+)*	*Mean ± SD*	1.48 ± 0.269	1.48 ± 0.203	1.48 ± 0.195
	SEM	0.0131	0.0116	0.0034
	95% CI	$[1.45, 1.50]$	$[1.45, 1.50]$	$[1.47, 1.48]$
Negative turning point (ρ_-)*	*Mean ± SD*	1.40 ± 0.275	1.38 ± 0.301	1.48 ± 0.103
	SEM	0.0131	0.0176	0.0019
	95% CI	$[1.37, 1.42]$	$[1.34, 1.41]$	$[1.48, 1.49]$

Continued on next page

Table 5.5 – *Continued from previous page*

RPT metric	**Statistic**	*EBS*	*AMHA-1*	*AMHA-2*
Positive quadratic area (QA_+)*	*Mean ± SD*	2.63 ± 0.867	2.59 ± 0.783	2.71 ± 0.759
	SEM	0.0412	0.0437	0.0130
	95% CI	$[2.55, 2.71]$	$[2.51, 2.68]$	$[2.68, 2.73]$
Negative quadratic area (QA_-)*	*Mean ± SD*	2.68 ± 1.031	2.66 ± 1.028	3.05 ± 0.815
	SEM	0.0477	0.0583	0.0139
	95% CI	$[2.58, 2.77]$	$[2.55, 2.78]$	$[3.02, 3.08]$
Polar angle (θ)	*Mean ± SD*	52.53 ± 14.654	58.89 ± 16.927	50.04 ± 11.499
	SEM	0.6560	0.8872	0.1950
	95% CI	$[51.24, 53.82]$	$[57.14, 60.63]$	$[49.66, 50.42]$
Polar dispersion (σ_θ)	*Mean ± SD*	40.73 ± 7.024	34.40 ± 15.146	40.73 ± 7.024
	SEM	0.3203	0.7917	0.3203
	95% CI	$[40.10, 41.36]$	$[32.84, 35.95]$	$[40.10, 41.36]$
Radial distance (r)	*Mean ± SD*	2.52 ± 0.256	2.60 ± 0.239	2.52 ± 0.256
	SEM	0.3203	0.7917	0.3203
	95% CI	$[2.50, 2.55]$	$[2.57, 2.62]$	$[2.50, 2.55]$
Radial dispersion (σ_r)	*Mean ± SD*	0.47 ± 0.298	0.38 ± 0.276	0.47 ± 0.298
	SEM	0.0133	0.0145	0.0133
	95% CI	$[0.44, 0.49]$	$[0.35, 0.40]$	$[0.44, 0.49]$

Legend: RPT features of the (K, H), (K, σ), and (H_-, H_+) curves of the IAPS picture rating data across the three distinct cohorts. Fifteen features were identified using common engineering methods, including five features from the value function, six from the limit function, and four from the trade-off function (see Methods of this chapter). For each of the three datasets, the mean and standard deviation (SD) are listed for the fifteen features, along with standard error of the mean (SEM) and the 95% confidence intervals (CIs) for the corresponding means. Unstarred features of the (K, H), (K, σ), and (H_-, H_+) curves are in dimensionless units, facilitating comparison across cohorts.

RA represents a function as shown in Figure 5.6. The results for comparing RA across subjects using a defined point on these functions is shown in Table 5.6.

Table 5.6. Nonparametric Kruskal-Wallis H-test comparison of RPT metrics across distinct cohorts for IAPS picture rating experiment

RPT metric	H-value	$p - value$
Loss aversion (LA) (logfit)	6.98785	3.0381×10^{-2}
Risk aversion (RA)	4.07003	1.3068×10^{-1}
Loss resilience (LR)	13.3089	1.2883×10^{-3}
Positive offset (β_+)	10.2435	5.9655×10^{-3}
Negative offset (β_-)	20.1216	4.2722×10^{-5}
Positive apex (α_+)	14.3479	7.6627×10^{-4}
Negative apex (α_-)	74.5914	6.3486×10^{-17}
Positive turning point (ρ_+)	2.28294	3.1935×10^{-1}
Negative turning point (ρ_-)	6.16587	4.5824×10^{-2}
Positive quadratic area (QA_+)	5.38677	6.7652×10^{-2}
Negative quadratic area (QA_-)	65.4338	6.1835×10^{-15}
Polar angle (θ)	9.9913	6.7672×10^{-3}
Polar dispersion (σ_θ)	125.001	7.1836×10^{-28}
Radial distance (r)	40.4942	1.6099×10^{-9}
Radial dispersion (σ_r)	89.6117	3.4759×10^{-20}

These results demonstrate that RA distributions across all three cohorts did not differ statistically using the Kruskal-Wallis H-test, as well as the post-hoc Dunn's test and two-sample K-S test assessments (Table 5.7).

Loss resilience (LR), which is computed the same way as RA using the avoidance curve, and represents the function shown in the third (lower-left) quadrants for Figure 5.6a-c. The three rating experiments produced similar functional forms for LR curves; it should be noted that the LR and RA curves with the rating experiment were similar in means and contained overlapping confidence intervals for the mean, as shown in Table 5.5. Additionally, there was statistically significant difference in LR across the three cohorts according to Table 5.6, however the post-hoc assessments

Table 5.7. Post-hoc pairwise Dunn's test and two-sample Kolmogorov-Smirnov (K-S) test statistics of RPT metrics across distinct cohorts for IAPS picture rating experiment

	Dunn's test p-values (Holm-Bonferroni corrected)			**Two-sample K-S test p-values**		
RPT metric	AMHA-1/ AMHA-2	AMHA-1/ EBS	EBS/ AMHA-2	AMHA-1/ AMHA-2	AMHA-1/ EBS	EBS/ AMHA-2
Loss aversion (LA) (logfit)	0.025	0.097	0.800	0.051	0.150	0.630
Risk aversion (RA)	0.130	0.290	0.790	0.052	0.140	0.790
Loss resilience (LR)	0.019	0.590	0.019	0.017	0.840	0.027
Positive offset (β_+)	0.400	0.017	7.50×10^{-3}	0.190	8.90×10^{-4}	0.001
Negative offset (β_-)	0.078	0.330	1.20×10^{-4}	0.077	0.500	1.10×10^{-3}
Positive apex (α_+)	0.580	0.094	4.80×10^{-4}	0.270	0.015	1.70×10^{-3}
Negative apex (α_-)	0.400×10^{-6}	0.320	2.00×10^{-13}	4.300×10^{-6}	0.360	1.10×10^{-9}
Positive turning point (ρ_+)	0.420	0.420	0.820	0.099	0.057	0.490
Negative turning point (ρ_-)	0.670	0.670	0.057	0.061	0.620	2.90×10^{-4}
Positive quadratic area (QA_+)	0.790	0.790	0.082	0.380	0.410	0.073
Negative quadratic area (QA_-)	1.10×10^{-5}	0.410	9.60×10^{-12}	3.60×10^{-4}	0.640	1.60×10^{-7}
Polar angle (θ)	0.340	0.015	9.60×10^{-3}	0.018	0.026	5.30×10^{-6}
Polar dispersion (σ_θ)	6.00×10^{-13}	0.890	1.40×10^{-18}	2.10×10^{-11}	0.560	3.80×10^{-15}
Radial distance (r)	1.500×10^{-6}	0.180	5.10×10^{-5}	1.30×10^{-5}	0.055	2.90×10^{-4}
Radial dispersion (σ_r)	9.600×10^{-13}	0.090	1.70×10^{-10}	1.10×10^{-11}	8.10×10^{-4}	1.60×10^{-9}

Legend: As a post-hoc assessment to the three-way nonparametric H-test comparison in Table 5.6, this table contains the p-values from a pairwise comparison between cohorts using Dunn's test (p-values are corrected using the Holm-Bonferroni method), as well as p-values from multiple pairwise comparisons of the distributions for each RPT metric across the cohorts using two-sample Kolmogorov-Smirnov (K-S) nonparametric tests.

both highlight that differences in LR for the AMHA-1 and EBS cohorts were not statistically significant ($p > 0.05$, Table 5.7).

Positive and negative offsets, β_+ and β_- respectively, are clearly present in both the logarithmic and power-law fits to the value function across the three rating experiments. For comparison they were computed from the logarithmic fit and did not significantly differ across cohorts. It should be noted that prospect theory does not allow for offsets to the value function and sets an inflection point connecting the positive and negative arms of the value function so there cannot be offsets for both functions [**211, 213**]. The current data confirms prior findings using an analysis of RPT features showing the existence of clear offsets to the value function from the origin when using (K, H) variables, and a discontinuity along the K-axis intercepts between the approach and avoidance data [**196, 198, 208**]. Although the Kruskal-Wallis H-test results in Table 5.6 show statistical significance in both β_+ and β_-, the post-hoc assessments in Table 5.7 don't demonstrate statistically significant differences in the distributions between AMHA-1 and AMHA-2 for β_+ and β_-, and AMHA-1 and EBS for just β_-.

Apex ($\alpha_\pm$), turning point ($\rho_\pm$), and quadratic area ($QA_\pm$) features are standard metrics of parabolic fits for the limit function (i.e., the fit for the mean-variance curves). The apices ($\alpha_\pm$), significantly differed across the three cohorts, however a number of the post-hoc assessments do not depict statistically significant differences (see Table 5.7). For the positive turning point, ρ_+, there was not a statistically significant difference across all cohorts (see Table 5.6), however the negative turning point, ρ_{-}, statistically differed in the Kruskal-Wallis three-way comparison. The post-hoc assessments for ρ_{-} shows that differences in the distributions were not statistically significant except for the K-S test's results in comparing the AMHA-2 and EBS cohorts; in contrast to these observations, Figure 5.5 shows overlapping 95% CIs for the median. For the positive quadratic area, QA_{+}, there were not statistically significant differences in approach and avoidance (Table 5.6), however for QA_{-} the Kruskal-Wallis comparison (Table 5.6) and post-hoc assessments (Table 5.7) indicate no statistical significance in differences for just the AMHA-1 and EBS cohorts. Qualitatively, the limit curves for approach and avoidance appear symmetric to each other relative to the H-axis (information/value) for the rating experiment; this is not the case for published keypress data [**196, 197, 198, 201, 208**].

Trade-off curve features showed consistent statistical differences for all four features across the three cohorts for the three-way Kruskal-Wallis comparison (Table 5.6), as well as CIs for the mean (Table 5.5). The mean polar angle, θ, for the rating experiments was > 45 degrees, consistent with a slight weighting of the rating assessments toward approach. The consistency between approach and avoidance variables is encoded in the radial distance feature, r, (i.e., if the increases in approach balanced decreases in avoidance, or if an individual felt more conflict, namely increases in both approach and avoidance). The radial distance features for all three experiments were just slightly within the semi-circle described for $r = \log_2(8)$, a theoretical frame for ensembles of eight pictures (please see [**196**]).

5.3.7. Extracted curve features and age

Distributions of age across the three samples showed generally flat distributions for the EBS and AMHA-1 cohorts, and a progressive increase in older subjects for AMHA-2 per violin plots (Figure 5.7). Univariate linear regressions between the 15 RPT features and age were consequently run for the AMHA-2 cohort, showing significant effects, after correction for multiple comparisons. These results are listed in Table 5.8 for each regression, the standardized β, adjusted R^2, and p-value associated with the overall regression was reported. Eleven out of the 15 RPT features showed trend effects with age. Only four features did not show a significant relationship with age after correction for multiple comparisons ($p < \frac{0.05}{15} \approx 0.0033$): LA, negative offset (β_{-}), positive turning point (ρ_{+}), and radial distance (r), which have implications for interpreting differences across the three cohorts, particularly where differences were suggested in RPT features between the EBS and AMHA-1 cohorts on one hand and the AMHA-2 cohort on the other. Overall, differences in features for the AMHA-2 cohort from the other two cohorts appear to be driven by differences in age between the cohorts.

5.4. Discussion

Across the three distinct cohorts collected with the same pictures and procedures, this study found that: (1) picture ratings without an operant framework produced RPT curves with a similar mathematical form as those produced in an operant context (where each action has a consequence by changing the viewing time). (2)

Table 5.8. Summary statistics of univariate linear regression across each RPT feature versus age for AMHA-2 cohort

RPT metric	**Standardized** β	$R^2_{\textbf{adj}}$	p**-value**
Loss aversion (LA) (logfit)	5.331×10^{-3}	-2.614×10^{-4}	0.7542
Risk aversion (RA)	0.1046	1.064×10^{-2}	1.373×10^{-9}
Loss resilience (LR)	0.1212	1.439×10^{-2}	2.675×10^{-12}
Positive offset (β_+)	-9.869×10^{-2}	9.441×10^{-3}	1.266×10^{-8}
Negative offset (β_-)	2.272×10^{-2}	2.210×10^{-4}	0.1862
Positive apex (α_+)	-0.1098	1.175×10^{-2}	1.898×10^{-10}
Negative apex (α_-)	5.176×10^{-2}	2.378×10^{-3}	2.859×10^{-3}
Positive turning point (ρ_+)	5.328×10^{-3}	-2.667×10^{-4}	0.7564
Negative turning point (ρ_-)	9.564×10^{-2}	8.857×10^{-3}	2.193×10^{-8}
Positive quadratic area (QA_+)	-9.996×10^{-2}	9.629×10^{-3}	5.095×10^{-9}
Negative quadratic area (QA_-)	6.124×10^{-2}	3.462×10^{-3}	3.117×10^{-4}
Polar angle (θ)	-8.792×10^{-2}	7.429×10^{-3}	4.259×10^{-7}
Polar dispersion (σ_θ)	0.2123	4.479×10^{-2}	1.789×10^{-34}
Radial distance (r)	-4.306×10^{-2}	1.554×10^{-3}	1.302×10^{-2}
Radial dispersion (σ_r)	0.1744	3.012×10^{-2}	4.255×10^{-24}

Rating-based RPT curves produced were discrete and recurrent across cohorts and scaled from individual to group data, meeting three of four criteria from [**199**] for lawfulness. (3) RPT curves across the three cohorts showed high symmetry between liking and disliking assessment, which qualitatively differs from what is observed with operant keypress tasks. (4) Several features of these RPT curves were consistent across the three cohorts, but in some cases, differed relative to other experimental contexts, such as age. Of note in this regard, age still did not affect some RPT curve features such as LA, which is an observation that has previously been reported with keypress data [**197**].

As with the operant keypress procedure in other studies, the picture rating task produced data that showed RPT-like relationships (e.g., Figure 5.1). The rating task value functions, like those observed with keypressing, followed the pattern observed with prospect theory [**211, 213**], and the limit functions followed those described by [**212**] for risk-reward curves. Individual R^2 values across the three studies ranged from 0.84 to 0.96, in line with previously published results in different cohorts using a keypress paradigm [**196, 198, 208**]. Extracted features from the corresponding curves for the three rating experiments showed statistically consistent, and visually similar patterns. In particular, the 95% CIs of the medians were broadly overlapping for the majority of RPT features, except for several differences between the EBS and AMHA-1 studies on one side and the AMHA-2 study on the other. Given the AMHA-2 study had a major difference in the proportion of older subjects (i.e., 55-70 years of age), and eleven of the 15 features used to compare cohorts were significantly associated with age for the AMHA-2 cohort, these differences likely relate to age distribution differences between cohorts.

The behavioral finance measure of risk aversion (RA) did not statistically differ between cohorts, whereas the same metric applied to the avoidance curves (i.e., referred to as LR herein) statistically differed between the AMHA-2 and the AMHA-1 and EBS cohorts, while the AMHA-1 and EBS cohorts were similar. Altogether, observations point to a greater symmetry in the valuing of positive (approach) and negative (avoidance) aspects of the stimuli when ratings are performed with no behavioral consequence, as opposed to prior studies based on keypressing. Supporting this observation, asymmetries in the mean-variance (K, σ) curves observed in prior operant keypress experiments [**198, 208**] were not evident in individual or group data from the rating task. These observations with RA and LR support the hypothesis raised by the LA results, suggesting rating-based tasks may reflect a lower regard for negative consequences.

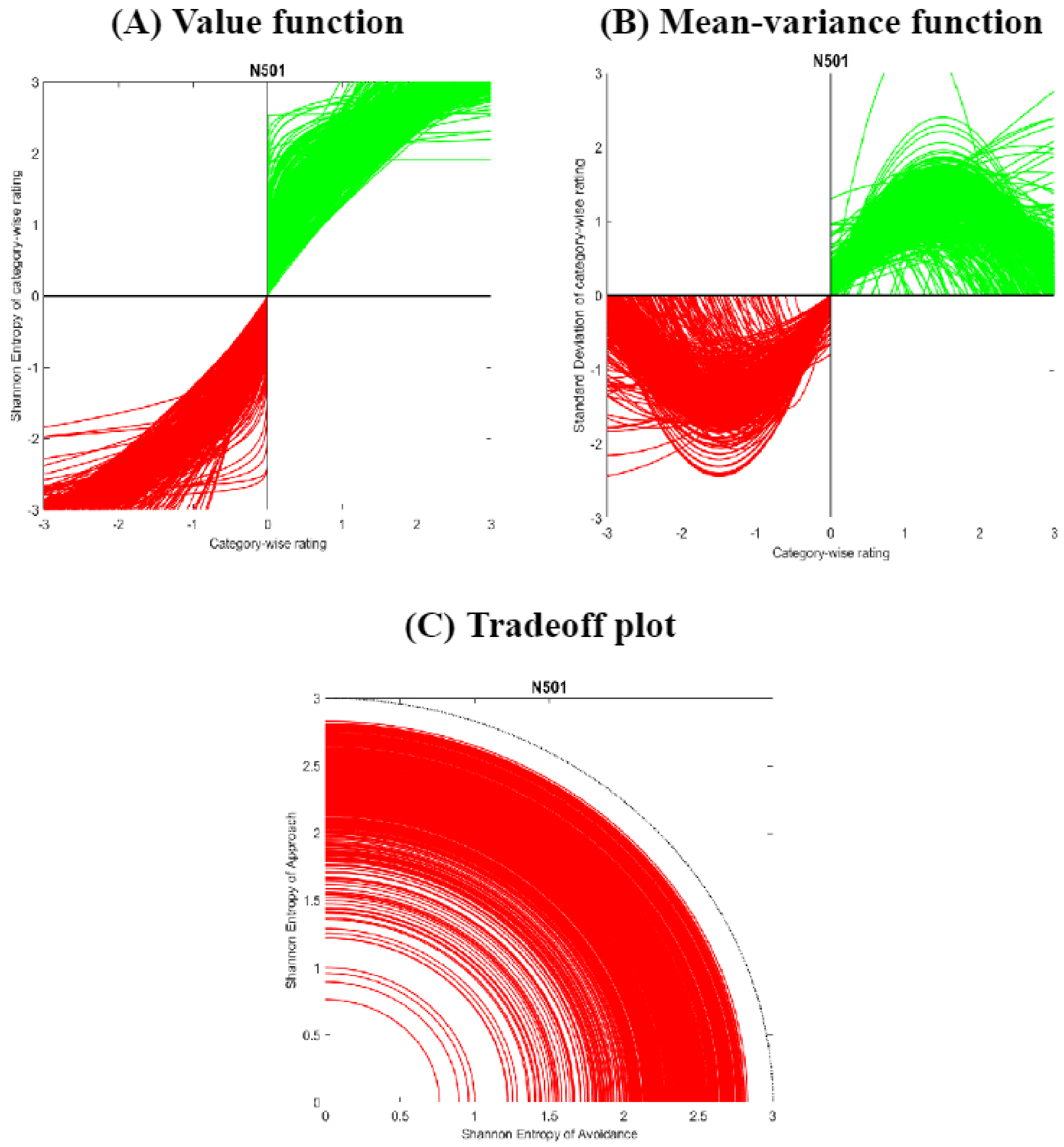
Although prospect theory [**211, 213**] considers the value function to be continuous with an inflection point, the rating task produced offsets, consistent with prior keypress experiments and RPT analyses [**196, 198, 201, 208**]. These offsets are suggestive of other psychological phenomena, such as the ante in poker, where a player must place a bet in the pot to enter the card game (e.g., β_{+}), or an insurance premium paid to counter potential bad outcomes (e.g., β_{-}). Further work is warranted to frame these findings.

Further research is also needed to deal with caveats to the current work, including that demographic matching between cohorts was not perfect, and the rating experiment used a unitary Likert-like scale as opposed to two scales wherein approach and avoidance assessments could be assessed independently. In line with caveat (1), we hypothesize demographic matching between cohorts may have contributed towards the statistical differences observed with the AMHA-2 when compared to AMHA-1 and EBS, such as differences in the distribution of age groups for AMHA-2 compared to the other cohorts. AMHA-2 specifically contains a right-skewed distribution of participants from an older-age group demographic (see Figure 5.7), whereas the distribution of age groups for AMHA-1 and EBS are more uniform.

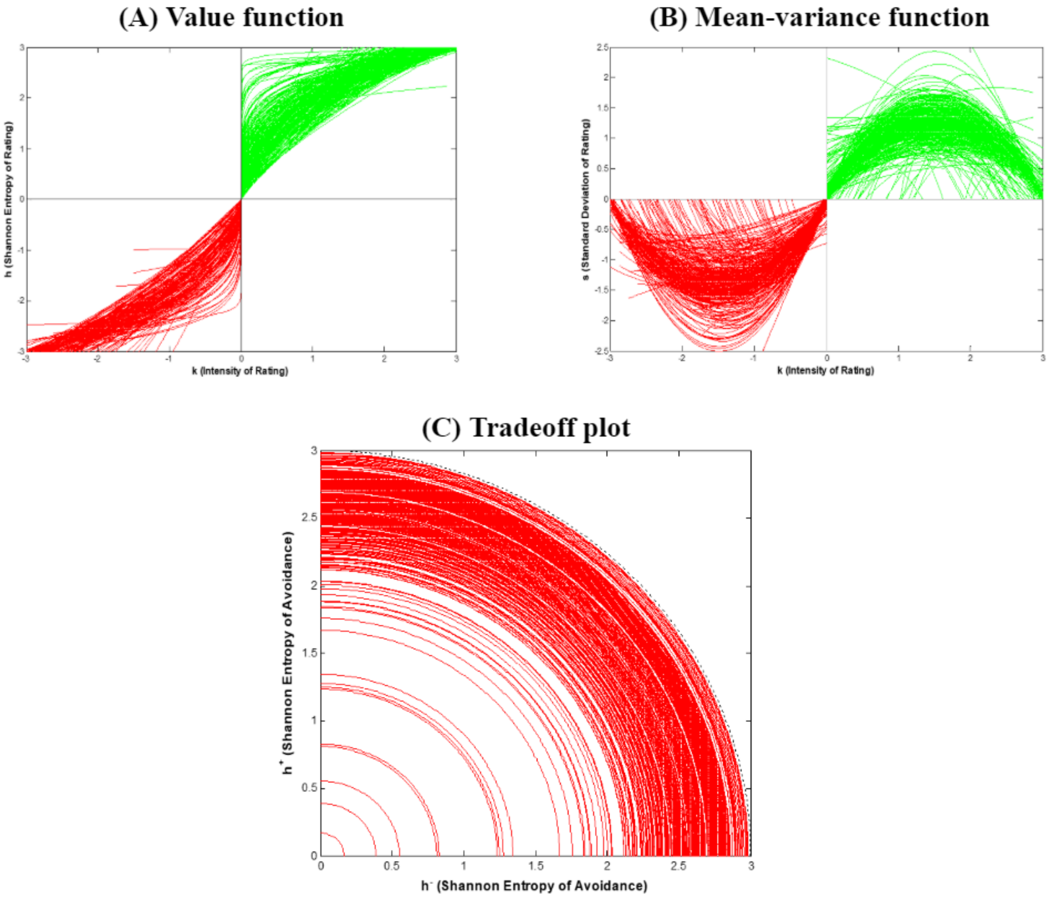
With replication, the current findings, using cohorts calibrated to the US Census, contributes to the big data psychology movement where the experimental environment cannot be as well-controlled as in a lab setting, but which can be conducted at a large-scale in much shorter time windows and with a major decrease in research team size and experimental time. Large-scale brain imaging and genetic studies are now quite common (e.g., The Connectome Project, ABCD Project, UK Brain Bank) and involve the collection of dense phenotyping data, although these studies are not primarily focused on human psychology and collect data over an extended time window with large human research teams. Studies with Amazon's Mechanical Turk have argued for extension of task-based psychology studies to the web with small research teams [**227, 228, 229, 230, 231**], although there has been some critique of such practices [**232, 233, 234**]. The current work points to the opportunity for testing computational behavior at larger scales than can be performed in the lab, allowing for greater sampling in the natural variance in measures.

In summary, the results of this study argue, that preference assessments made through a short and simple rating task can be modelled quantitatively with R^2 (goodness of fits) above 0.80, for RPT-based value functions, limit functions, and trade-off functions. Rating-based curves meet three of the four strict criteria for lawfulness set out by [**199**]. Lastly, these curves appear to differ from those produced from operant keypressing [**198**], particularly with the issue of overweighting of perceived negative stimuli relative to positive ones when individuals must trade effort for exposure to the stimulus; these observations support the hypothesis that judgments lacking consequence may reflect lower aversion to negative outcomes, and reduce inhibitions against negative digital behavior at large. Given the simplicity of the rating task application and its analysis, this approach to preference quantitation could be easily applied to the 83.72% of the world's population that currently owns a smartphone [**10**], or the 85% of Americans with a smartphone (at least 97% own a cellphone of some kind) [**11**].

Figure 5.3. Individual RPT fits of the three distinct cohorts for the IAPS picture rating task.



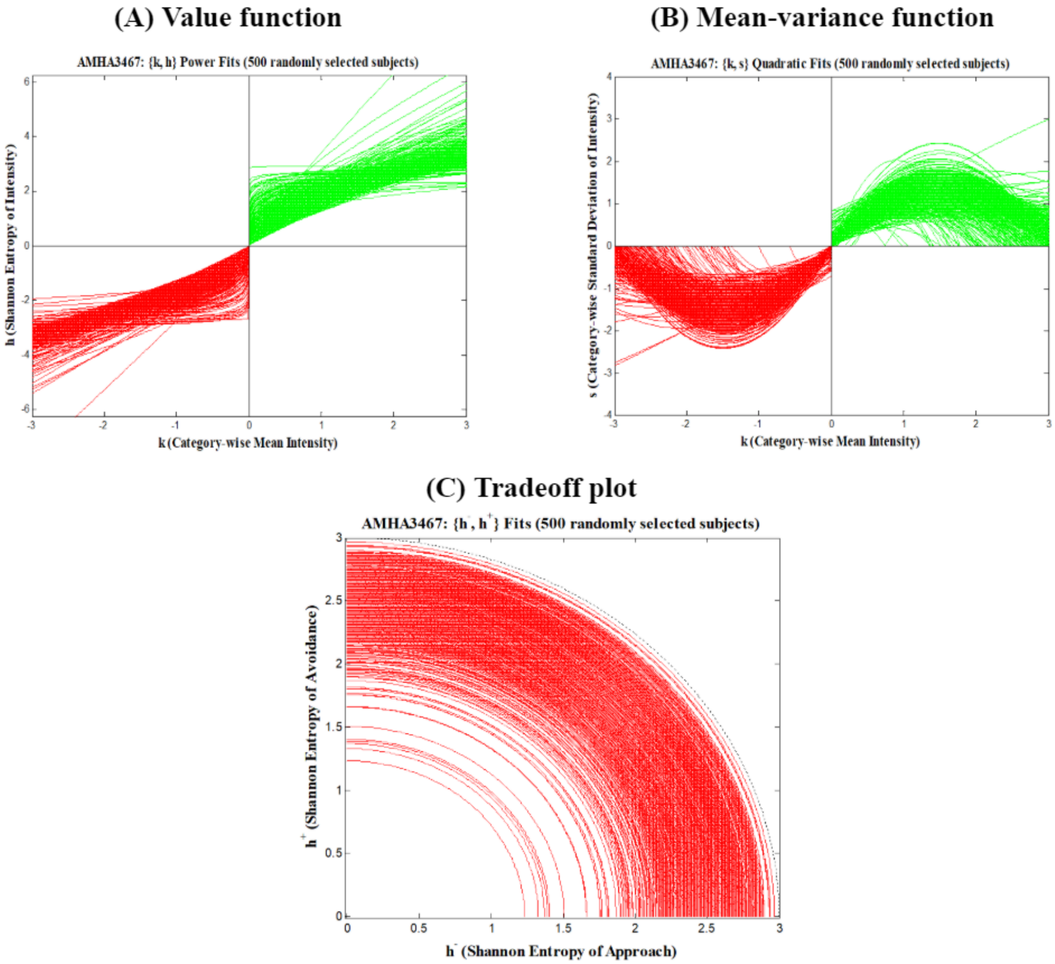
(a) EBS cohort RPT curves.


(A) Value function

(B) Mean-variance function

(C) Tradeoff plot

(b) AMHA-1 cohort RPT curves.


(A) Value function

(B) Mean-variance function

(C) Tradeoff plot

(c) AMHA-2 cohort RPT curves.

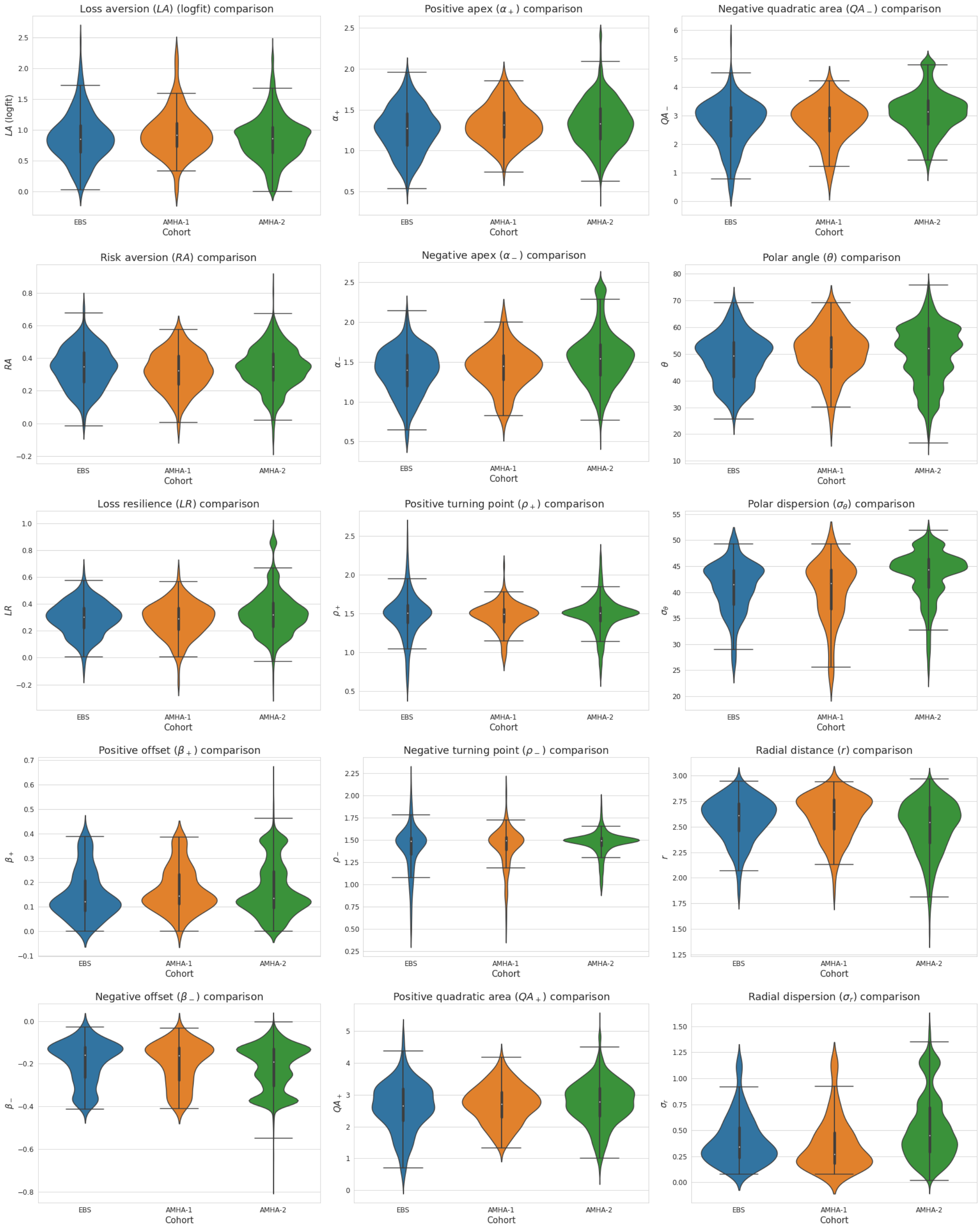


Figure 5.5. Violin plots [**2**] for each of the RPT features are tiled to provide a hybrid visual comparison of the distribution, interquartile range (IQR), and 95% CIs, with respect to the corresponding median, for each RPT feature across all cohorts.

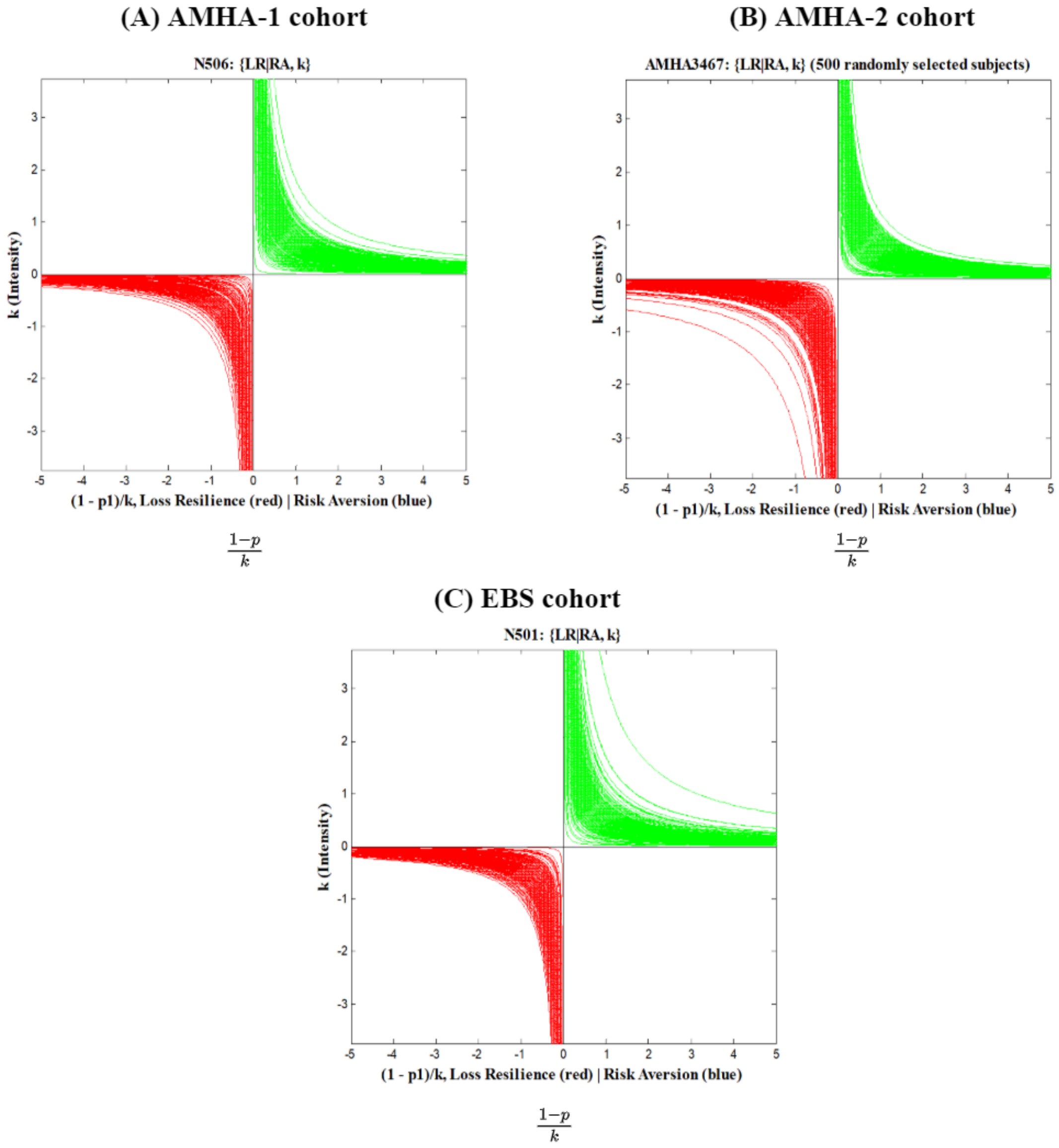


Figure 5.6. Individual risk aversion (RA) and loss resilience (LR) functions for the three cohorts. **(A)** Risk aversion (RA) functions comparing computed RA to mean picture ratings (K) in individual participants are shown for the AMHA-1 cohort. Loss resilience (LR) (the same computation as RA done for avoidance ratings) comparing computed LR to mean picture ratings intensity (K) in individual participants are shown as well. Note the hyperbolic functional forms in green (approach) and red (avoidance) for each curve. **(B)** Risk aversion functions and loss resilience functions shown for the AMHA-2 cohort. Note the hyperbolic functional forms in green (approach) and red (avoidance) for each curve. **(C)** Risk aversion functions and loss resilience functions show for the EBS cohort. Note the hyperbolic functional forms in green (approach) and red (avoidance) for each curve.

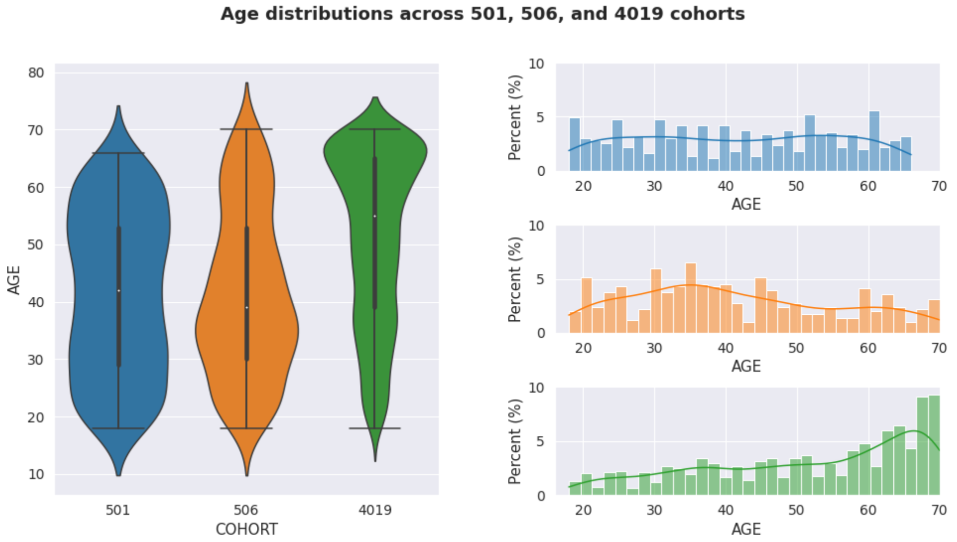


Figure 5.7. Violin plots [**2**] for each of the three cohorts are provided for a visual comparison of the participant age distributions, along with the respective interquartile ranges (IQRs), and 95% CIs, with respect to the corresponding median across all cohorts.

CHAPTER 6

Predicting Demographics using Human Reward Behavior

Abstract

In this work, graph neural networks (GNNs) have primarily been discussed in the context of trimeshes and their interpretation as *homogeneous* graphs (single vertex/edge types). In recent years, GNNs have gone from a niche sub-topic of machine learning, to an emerging success for modeling unstructured data. As they gain popularity, special attention needs to be paid towards modeling heterogeneity of entity and relation types within graphs, particularly interaction graphs. In this chapter, we utilize a special type of GNN architecture known as the heterogeneous graph transformer (HGT) and apply them to tackle the challenge of *semi-supervised* node classification for predicting demographics of human nodes within a heterogeneous graph describing their interaction with picture stimuli from International Affective Picture System (IAPS). Respectively, we achieve an average 0.85/0.79/0.73/0.85 accuracy/F-score/precision/recall scores for predicting gender assigned at birth as the target demographic experiment. This framework requires further experimentation and can easily be adapted to predict other demographics and targets related to human reward behavior, which we describe using the approach-avoidance features

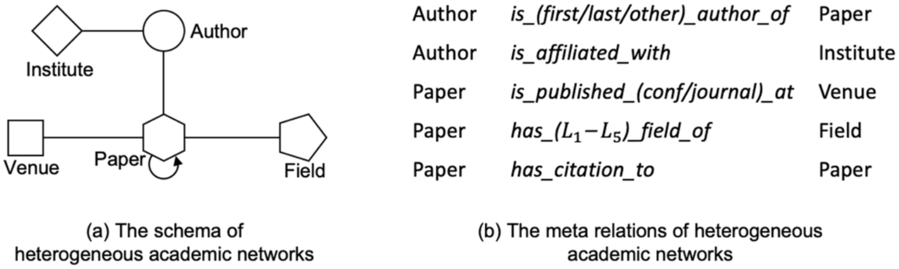
from the previous chapter using the International Affective Picture System (IAPS) picture ratings task.

6.1. Introduction

So far, *homogeneous* graphs have been the centerpiece for the majority of this text, where the entities of graphs are assumed to be of one *type*, as well as the edges. However, many complex systems and networks are naturally *heterogeneous* in nature, where entities can be of multiple types, as well as their relations to one another. Like standard message passing neural networks (MPNNs) on homogeneous graphs, message passing on heterogeneous graphs can be defined using relationships between entities expressed as triplets (or tuples) in the form of: $\{e_i, r_{ij}, e_j\}$, where $e_{i,j}$ represents entities, e, of type i and type j, and their relationship, r_{ij}. As mentioned before, entities can contain vectors of features that correspond to a particular entity type and relationships between entities can also contain feature vectors; the only difference here is that relationship edges can connect entities of different types as well.

Some prevalent examples of applications with heterogeneous graphs include: academic citation graphs where nodes can be of *author*, *paper*, and *journal* types, social media graphs (e.g., Facebook entity graph, LinkedIn economic graphs), and more broadly the Internet of Things (IoT) networks. As an example, the Open Academic Graph (OAG) in Figure 6.1 outlines a heterogeneous graph that contains five types of nodes: papers, authors, institutions, venues (journal, conference, or preprint), and fields, as well as the varying types of relationships amongst them.

Over the past decade, significant research has been explored in the paradigm of mining heterogeneous graphs. Besides classical graph mining methods, like using meta paths (such as PathSim [**236**] and metapath2vec [**237**]), graph neural networks (GNNs) [**38, 32, 238**] have also been adopted to learn with heterogeneous networks [**239, 240, 241**]. More recently, heterogeneous graph transformers (HGTs) [**235**] have been presented for successfully mining Web-scale heterogeneous graphs, specifically by modeling heterogeneity using node- and edge-type dependent learnable parameters to characterize heterogeneous attention over each relation-type based on the triplets previously discussed. Doing so, empowers HGTs to handle learning about heterogeneous relationships by establishing dedicated learned representations for different types of nodes and edges on a graph.



(a) The schema of heterogeneous academic networks

Author	*is_(first/last/other)_author_of*	Paper
Author	*is_affiliated_with*	Institute
Paper	*is_published_(conf/journal)_at*	Venue
Paper	*has_(L_1−L_5)_field_of*	Field
Paper	*has_citation_to*	Paper

(b) The meta relations of heterogeneous academic networks

Figure 6.1. Schema and meta relations of the Open Academic Graph (OAG) described and used in [**235**].

In this work, we look at the IAPS [**214, 215**] picture rating task described in the previous chapter as a heterogeneous graph by abstracting the picture rating interactions between human participant nodes and IAPS stimuli nodes. Doing so enables us to perform *semi-supervised* node classification on the participant nodes to make demographic predictions about participants in heterogeneous interaction graphs based on dynamic interactions with IAPS stimuli and their respective categories. To handle large-scale interaction graph data, we adopt the heterogeneous mini-batch graph sampling algorithm presented by Hu *et al.* [**235**] for efficient and scalable training of our HGTs. Respectively, we achieve an average 0.85/0.79/0.73/0.85 accuracy/F-score/precision/recall scores for predicting gender assigned at birth as the target demographic experiment. With further experimentation, this work can easily be adapted to potentially predict other demographics in relation to human reward behavior, which we describe using approach-avoidance behavior from a simple picture rating task.

6.2. Related Works

In a 2011 report [**242**], the World Economic Forum and Harvard School of Public Health noted that non-communicable diseases pose a greater risk than contagious illnesses in the future. Claiming 63% of all deaths, non-communicable illnesses are currently the world's main killer. Eighty percent of these deaths now occur in low- and middle-income countries, making this all the more prevalent. Half of those who die of chronic non-communicable diseases are typically in the prime of their productive years, and thus, the disability imposed and lives lost to non-communicable illness are also endangering industry competitiveness across borders. Their report [**242**] projected that non-communicable diseases, particularly mental health issues, will be the largest source of costs in global health (more than a third by 2030).

Personality and approach-avoidance behaviors are core concepts in research on mental health issues. Prior work in this area [**243**] has demonstrated that responding to stimuli in ambiguous environments is partially governed by approach-avoidance tendencies. Imbalances in approach-avoidance behaviors towards rewards are implicated in a variety of mental disorders including anxiety disorders, phobias, substance use disorders, and behavioral/societal biases. Approach-avoidance tendencies are constitutionally ingrained to the brain networks implicated in action and reaction to salient stimuli and controlling cognitive and attentional functions, reward sensitivity and emotional expression, since all organisms following a phylogenetic gradient, tend to have highly-conserved mammalian tendencies to approach and avoid certain stimuli. Based on approach-avoidance behavior we may be able to predict demographics such as age. As an example: children tend to exhibit emotional lability, impulsivity, and proclivity to seek rewards, even if these tendencies are maintained in adulthood.

As previously mentioned, imbalances in approach and avoidance behavior can lead to psychopathological disorders such as attention-deficit/hyperactivity disorders [**244**], depression [**245**], substance abuse [**204**], anxiety [**246**], and post-traumatic stress disorders [**247**]. Importantly, many of these conditions can affect humans differently based on demographics such as gender assigned at birth. As a precursor to further research into predicting mental health issues using approach-avoidance behavior (and as a proof of concept), in this work we utilize the broad set of human approach-avoidance features (i.e., relative preference theory (RPT) features [**196, 248, 220**] described in the previous chapter) extracted from a simple picture rating task to predict gender assigned at birth. Given the abundance of compounding factors (e.g., societal biases) that may influence the results of this study, it is important to note that this work is *not claiming* human behavior is deterministic of gender as a whole. In humans, although some actions may derive directly and invariably from these proclivities, ultimate behavior may be self-regulated and subjected to strategic planning, so that individuals can override their initial inclinations and redirect behavior (e.g., putting approach behavior into action to override avoidance tendencies). The simple, computer-based picture rating task employed in this study can easily be adapted and performed on any mobile device with a screen, making it a scalable solution to preference quantitation which could be easily applied to the 83.72% of the world's population that currently owns a smartphone [**10**], or the 85% of Americans with a smartphone (at least 97% own a cellphone of some kind) [**11**].

Today, consumer mobile data, particularly consumer demographics (e.g., gender and age), can play a core role in enabling companies and medical providers to enhance the offers of their services for targeting the right consumers and patients in the right time, manner, and place. Mobile data is increasingly used for humanitarian purposes, as traditional data can be scarce in certain scenarios. In some cases, demographic information can often be absent from mobile phone datasets, limiting the operational impact of the datasets. Prior work on demographic prediction from mobile data has focused on users' names [**249**], social media photos [**250**], and the diversity of writing and speaking styles associated with the demographic attributes. Eckert [**251**] and Holmes [**252**] classified users' gender using spoken language differences including intentional, phonological and conversational cues. More recently, Hu *et al.* [**253**], modeled user web browsing behavior as weighted bipartite graphs and they used support vector machines (SVMs) to classify user gender and regress on user age. SVMs tend to perform great on relatively smaller datasets, but come with a number of caveats in comparison to deep learning (DL) approaches like GNNs (i.e., NNs are “flexible” in that they approximate their own internal representations of input features, rather than having it pre-specified by the kernel function like SVMs).

6.3. Methods

6.3.1. Heterogeneous graphs preliminary

Heterogeneous graphs are abstractions of modeling relational real-world data and complex systems/interactions. As presented by Hu *et al* . [**235**], in this work heterogeneous graphs are defined as:

Definition 6.3.1 (Heterogeneous graph/network)**.** Heterogeneous graphs are defined as directed graphs, $G(\mathcal{V}, \mathcal{E}, \mathcal{A}, \mathcal{R})$, containing the set of nodes/vertices, $v \in \mathcal{V}$, the set of relations/edges, $e \in \mathcal{E}$, along with their type mapping functions, $\tau(v) : V \to \mathcal{A}$, and $\phi(e) : E \to \mathcal{R}$, respectively.

Meta-relation. For a directed edge, $e = (s, t) \in \mathcal{E}$, we can define the meta-relation linking the source node, s, to target node, t, as the triplet $\langle \tau(s), \phi(e), \tau(t) \rangle$. Meta-relations are defined in such a way by Hu *et al.* [**235**] in order to better generalize real-world heterogeneity by assuming that multiple relationships can exist between different types of nodes. In their example using OAG, different types of relationships between *author* and *paper* nodes in an academic citation network can exist by considering authorship order (e.g., first, second, etc.).

6.3.2. Graph neural network generalization

As a generalization from traditional DSP, graph neural networks (GNNs) can be thought of as encoders that use the input graph structure as the computation graph for message passing [**163**], where neighborhood information is aggregated using a set of rules to obtain contextual representations of nodes on the input graph with respect to local neighborhoods and/or the overarching graph topology. Formally message passing is defined by Hu *et al.* [**235**] as:

Definition 6.3.2 (Generalized GNN message passing)**.** Suppose $H^l[i]$ is the node representation of node i at the the l-th GNN layer, the "update" procedure from the $(l-1)$-th layer to the l-th layer for the target node, t, from source node, s, via message passing is defined as:

$$H^l[t] \leftarrow \underset{\forall s \in N(t), \forall e \in E(s,t)}{\mathbf{Aggregate}} \left(\mathbf{Extract} \left(H^{l-1}[s]; H^{l-1}[t], e \right) \right), \tag{6.1}$$

where $N(t)$ denotes all of the source nodes, s, in the neighborhood of node t, and $E(s,t)$ denotes all the edges from node s to t.

Based on Definition 6.3.2, message passing in GNNs is defined by the $\mathbf{Extract}(\cdot)$ and $\mathbf{Aggregate}(\cdot)$ operators. **Extract** is the neighborhood feature extractor that uses the source nodes' representation $H^{l-1}[s]$, along with the target node's representation $H^{l-1}[t]$, and the edge, e, between adjacent nodes as query. As discussed in prior chapters, **Aggregate** gathers the neighborhood information from the target node,

t, with respect to the source nodes, s, using an arbitrary permutation-invariant aggregation operator (e.g., max, min, sum, average). GNN architectures are generally proposed following this framework, with GCNs being one of the earliest examples by Kipf *et al.* [**32**], where they use average **Aggregate** operations with one-hop neighbors for each node in the graph, followed by **Extract** steps in the form of linear projection layers (i.e., feed-forward NNs or MLPs), and non-linear activation functions. This work uses HGTs, which are reminiscent of GATs proposed by Velickovic *et al.* [**238**], where they introduce attention mechanisms into GNNs to allow GATs to assign different "importance" weights to adjacent target nodes, $N(t)$, and edges, e, within the assigned neighborhood of a source node, s.

6.3.3. Heterogeneous graph transformers

Following the conventions of Hu *et al.* [**235**], heterogeneous graph transformers (HGTs) are introduced in this section using the idea of *meta-relations* in heterogeneous graphs in order to parameterize learnable weight matrices for heterogeneous mutual attention, message passing, and propagation steps in the HGT paradigm. The overall architecture for HGT layers is depicted by Figure 6.2, where target node, t, is linked by source nodes, s, via edge, e. The goal of HGTs is to obtain a contextualized representation of the target node, t, by aggregating neighborhood information from source nodes, irrespective of differing node types. This process can be deconstructed into its three core components, which are: (1) *heterogeneous mutual attention*, (2) *heterogeneous message passing*, and (3) *target-specific aggregation*, each outlined within Figure 6.2.

The output of the (l)-th HGT layer is denoted as $H^{(l)}$, which also serves as the input to the $(l+1)$-th layer. Using L HGT layers, the node representations (or embeddings) of the whole graph, $H^{(L)}$, can be obtained and used for end-to-end training or downstream tasks for heterogeneous graphs.

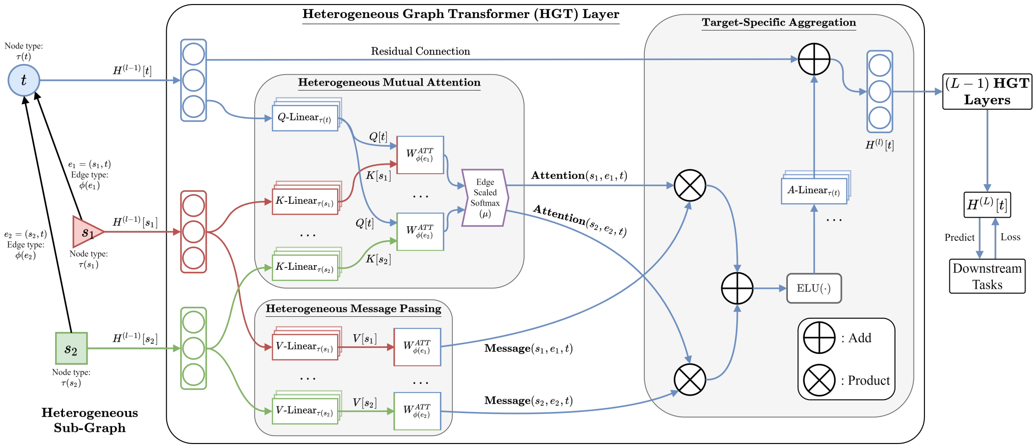


Figure 6.2. **Heterogeneous graph transformer (HGT) computation architecture.** Given the sampled heterogeneous sub-graph of source nodes, $s_{1,2}$, and target node, t, HGTs take edges, $e_1 = (s_1, t)$ and $e_2 = (s_2, t)$, along with their corresponding meta relations, $\langle\tau(s_1), \phi(e_1), \tau(t)\rangle$ and $\langle\tau(s_2), \phi(e_2), \tau(t)\rangle$, as input to learn contextual representations $H^{(L)}$ for each node. Colors in the HGT diagram are used to denote node types. Overall, the HGT model is constructed of three components: (1) meta-relation-aware heterogeneous mutual attention, (2) heterogeneous message passing from source nodes, and (3) target-specific heterogeneous message aggregation as defined by Hu *et al.* [**235**].

6.3.3.1. Heterogeneous mutual attention. The first step in HGTs is to compute the mutual attention between a source node, s, and target node, t. Attention-based GNNs like GATs can be generalized in the form:

$$H^l[t] \leftarrow \underset{\forall s \in N(t), \forall e \in E(s,t)}{\textbf{Aggregate}} \left(\textbf{Attention}(s,t) \cdot \textbf{Message}(s) \right). \tag{6.2}$$

In principle, the **Attention** function estimates the relative "importance" of each source node to the target node; the **Message** function extracts the "message" from the source node; and **Aggregate** fuses the neighborhood message from s to t while also considering relative importance from the attention weights.

Given a target node, t, and its neighbors, $s \in N(t)$, which may belong to varying distributions (since nodes can be of varying types), mutual attention is computed using **meta-relations** (i.e., the $\langle \tau(s), \phi(e), \tau(t) \rangle$ triplets). Motivated by the design of the Transformer [**254**] architecture, target nodes, t, are mapped into Query vectors, $Q[t]$, and source nodes, s, into Key vectors, $K[s]$, and their dot products are calculated as "attention." Unlike "vanilla" Transformers [**254**], where a single set of projections is used for all words, for HGTs we use a distinct set of projection weights for each meta-relation. Specifically Hu *et al.* parameterize weight matrices into source node feature projections, edge feature projections, and target node projections. For each edge, $e(s,t)$, the h-head attention (see Figure 6.2) is computed such that

$$\mathbf{Attention}_{HGT(s,e,t} = \underset{\forall s \in N(t)}{\text{Softmax}} \left(\underset{i \in [1,h]}{\|} \mathit{ATT\text{-}head}^i\left(s,e,t\right) \right) \tag{6.3}$$

$$\mathit{ATT\text{-}head}^i\left(s,e,t\right) = \left(K^i\left(s\right) W^{ATT}_{\phi(e)} Q^i(t)^T\right) \cdot \frac{\mu_{\langle \tau(s),\phi(e),\tau(t)\rangle}}{\sqrt{d}}$$

$$K^i(s) = K\text{-Linear}^i_{\tau(s)}\left(H^{(l-1)}[s]\right)$$

$$Q^i(t) = Q\text{-linear}^i_{\tau(t)}\left(H^{(l-1)}[t]\right)$$

For the i-th attention head $\mathit{ATT\text{-}head}^i(s,e,t)$, the $\tau(s)$-type source node, s, is projected into the i-th Key vector, $K^i(s)$, using a linear projection: $K\text{-Linear}^i_{\tau(s)} : \mathbb{R}^d \rightarrow \mathbb{R}^{\frac{d}{h}}$, for h attention heads and $\frac{d}{h}$ is the vector dimension per head. For each source node type, $\tau(s)$, we use a unique linear projection layer, $K\text{-Linear}^i_{\tau(s)}$, to maximally model the potential varying distributions among node types. The same is done for the target node, t, using the linear projection, $Q\text{-Linear}^i_{\tau(t)}$, for the i-th Query vector.

The next step in computing mutual attention for GNNs is to calculate the similarity between Query vectors, $Q^i(t)$, and Key vectors, $K^i(s)$. A special characteristic of heterogeneous graphs is that multiple varying edge types can exist between node type pairs, e.g., $\tau(s)$ and $\tau(t)$. Unlike "vanilla" Transformers [**254**], which directly calculates the dot product between Query and Key vectors, HGTs keep distinct edge-based matrices, $W^{ATT}_{\phi(e)} \in \mathbb{R}^{\frac{d}{h} \times \frac{d}{h}}$, for each edge type $\phi(e)$, in order to capture the semantics in varying relations between the same node type pairs. Additionally, since relationships can have varying degrees of contributions to the target nodes, a prior tensor, $\mu \in \mathbb{R}^{|\mathcal{A}| \times |\mathcal{R}| \times |\mathcal{A}|}$, is added to indicate the general significance of each meta-relation triplet, $\langle \tau(s), \phi(e), \tau(t) \rangle$, serving as an adaptive scaling to the attention weights.

The final step in heterogeneous mutual attention is to concatenate the h attention heads together in order to obtain an attention vector for each node pair. For each target node, t, all attention vectors are gathered from their respective neighbors, $N(t)$, (i.e., source nodes, s), and a Softmax computation is conducted in order to fulfill $\sum_{\forall \in N(t)} \mathbf{Attention}_{HGT}(s, e, t) = \mathbf{1}_{h \times 1}$.

6.3.3.2. Heterogeneous message passing. Meanwhile computing heterogeneous mutual attention, information is also propagated (follwing Figure 6.2) from source nodes, s, to target nodes, t, via message passing. Meta-relations are also taken into consideration for message passing on heterogeneous graphs using HGTs [**235**]. For the pair of nodes, $e = (s, t)$, the multi-headed **Message** is calculated such that:

$$\mathbf{Message}_{HGT}(s, e, t) = \underset{i \in [1,h]}{\|} \textit{MSG-head}^i(s, e, t) \tag{6.4}$$

$$\textit{MSG-head}(s, e, t) = M\text{-Linear}^i_{\tau(s)} \left(H^{(l-1)}[s] \right) W^{MSG}_{\phi(e)}.$$

To obtain the i-th message head, $\textit{MSG-head}^i(s, e, t)$, the $\tau(s)$-type source node, s, is projected into the i-th message vector using the linear projected layer, $M\text{-Linear}^i_{\tau(s)} : \mathbb{R}^d \rightarrow \mathbb{R}^{\frac{d}{h}}$. To incorporate edge-dependency within message passing, the weight matrix, $W^{MSG}_{\phi(e)} \in \mathbb{R}^{\frac{d}{h} \times \frac{d}{h}}$, is incorporated afterwards. As a final step similar to heterogeneous mutual attention, all h message heads are concatenated, $\underset{i \in [1,h]}{\|}$, to get $\mathbf{Message}_{HGT}(s, e, t)$ for each node pair, $e = (s, t)$.

6.3.3.3. Target-specific Heterogeneous Aggregation. Since the Softmax procedure from Equation 6.3 forces the sum of the attention vectors for each target node, t, to be $\mathbf{1}_{h\times 1}$, we can use the attention vectors to perform weighted averages of the corresponding messages from the source nodes to get the updated vector (using the same notation as Hu *et al.*) $\tilde{H}^{(l)}[t]$ such that:

$$\tilde{H}^{(l)}[t] = \bigoplus_{\forall s \in N(t)} \left(\mathbf{Attention}_{HGT}(s,e,t) \cdot \mathbf{Message}_{HGT}(s,e,t)\right).$$

This step aggregates information (messages) to the target node, t, from all its neighborhood (source nodes), $s \in N(t)$, regardless of varying feature distributions.

As a final step, following the residual connection convention by He *et al.* [**114**], the target node's vector is mapped back to its $\tau(t)$-type specific distribution, by applying another linear projection layer, $A\text{-Linear}_{\tau(t)}$ to the updated vector, $\tilde{H}^{(l)}[t]$, as:

$$H^{(l)}[t] = A\text{-Linear}_{\tau(t)}\left(\sigma\left(\tilde{H}^{(l)}[t]\right)\right) + H^{(l-1)}[t]. \tag{6.5}$$

6.3.4. IAPS picture stimuli rating task

The International Affective Picture System (IAPS) [**214, 215**] stimulus set is a well-validated emotional stimulus set of 48 pictures across six thematic categories (as described in the previous chapter): (1) sports, (2) disasters, (3) cute animals, (4) aggressive animals, (5) nature (beach vs. mountains), and (6) food, with eight pictures spread evenly per category. Pictures had a maximum size of $1,204 \times 768$

pixels. All picture stimuli reported in this chapter are collectively referred to as "IAPS stimuli" throughout the text.

Human participants were prompted to rate IAPS stimuli while completing an online digital survey, which contained questionnaires regarding participant demographic information and research questionnaires for depression symptoms using the Patient Health Questionnaire (PHQ-9) [**216**]; trait anxiety using the Spielberger State-Trait Anxiety Inventory (STAI) [**217**]; a broad array of mental health, neurological, and medical issues using the MGH Phenotype Genotype Project in Addiction and Mood Disorders symptom questionnaire (MGH-SQ); and behavioral health disorders (e.g., internalizing or externalizing psychiatric disorders, substance use disorders, or crime/violence problems) from the GAIN-SS short screen assessment [**218**]. As discussed in the previous chapter, each picture was presented as shown in Figure 5.2, with the ratings, along a 7-point Likert-like scale from -3 to 3, below each stimulus. There was no operant consequence in the rating task, i.e., no change in viewing time, no time limit for assigning ratings to each picture, but participants were requested to rate each picture as quickly as possible without the ability to change their response after selecting a rating.

6.3.5. Picture Rating Interaction Graph

In this section, we adopt a graph formalism to abstract the previously described IAPS picture rating task as a heterogeneous interaction network (graph) between human participant nodes and picture stimuli from the International Affective Picture System (IAPS) [**214, 215**].

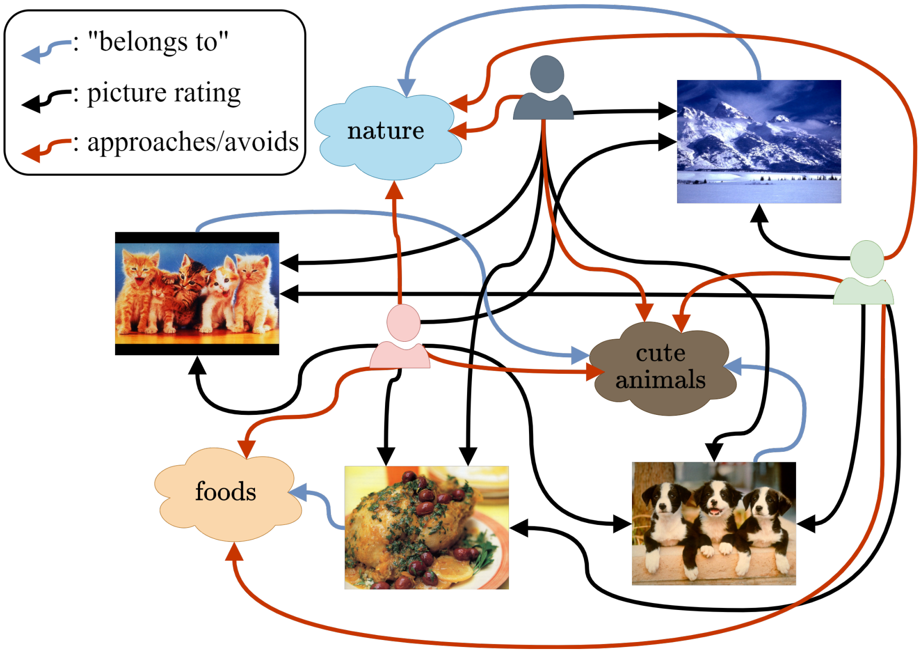


Figure 6.3. Heterogeneous sub-graph (-network) depicting human interaction with IAPS stimuli and abstracted picture categories as interactive objects. Three participants are depicted along with their picture rating relationships with four hand-picked IAPS stimuli belonging to three distinct picture categories (i.e., foods, cute animals, nature (beach vs. mountains)). Relationships (edges) amongst the entities are color-coded based on the three possible relationships: (1) IAPS stimuli “belonging to” a picture category, (2) how an individual rates a particular IAPS stimulus, and (3) how an individual approaches/avoids a specific picture category based on the hand-crafted portfolio of relative preference theory (RPT) features previously described [**248, 220, 196**].

6.3.5.1. Entity (node) features. As depicted in Figure 6.3, the nodes in our heterogeneous graphs were: (1) the human participants who participated in the IAPS picture rating task, (2) the 48 picture stimuli, and (3) the 6 previously described picture categories which are abstracted as interactive objects. The node features at the human entities within our heterogeneous graphs contain demographic features specific to each individual (i.e., age, gender assigned at birth, household income, occupation, etc.). We omit, the target demographic(s), in the event of training a ML model to predict one or more of them. Given the categorical nature of the IAPS picture categories (i.e., 6 possible categories), one-hot encoded vectors (i.e., $\in \mathbb{R}^6$) pertaining to the respective picture categories are assigned to the picture category nodes.

Lastly, given that IAPS stimuli are *images* (i.e., 2D/3D data, matrices of pixel intensities), we choose to simplify the features at the stimulus nodes by extracting meaningful vector representations of IAPS stimuli using *transfer learning* [**255**] with convolutional neural networks (CNNs). Highly-accurate, modern-day CNNs typically have millions of parameters that require a large amount of training data and computational power to train from scratch. The intuition behind transfer learning is that if a ML model is trained on a large and general enough dataset, said model can potentially serve as a generic model of the visual world. With transfer learning, we can take advantage of learned vector representations of images without having to train a large CNN model from scratch on a large dataset. By "freezing" the learnable parameters of our pre-trained model, we can simply add a new trainable classifier on top of the pre-trained model (which will be trained from scratch) so that the feature maps, learned previously from the larger dataset, can be repurposed for a new objective. We refer to this step as "feature extraction", since we start with a pre-trained CNN and only "fine-tune" the parameters of the newly-added final layer weights, from which we derive predictions.

In this work, we used a ResNet-50 [**114**] CNN model, pre-trained on the ImageNet [**256**] database, before fine-tuning on our dataset. CNNs models that are trained on extensive datasets ImageNet, perform well at recognizing objects in images. Therefore, it only makes sense to directly use what they have learned and fine-tune them to our purpose. We did not perform any additional preprocessing to the IAPS stimuli besides the image preprocessing steps already used in the original ResNet-50 [**114**] paper.

Following Figure 6.4, we take a pre-trained ResNet-50 CNN model, "freeze" its convolutional layers' learnable parameters, and remove its final two output layers (i.e., "fully-connected" MLPs). Then, we initialize two new fully-connected layers for classifying IAPS stimuli into one of six possible picture categories instead of the original ImageNet [**256**] classes. Doing this allows us to transfer domain knowledge from the original object recognition task to categorizing IAPS stimuli. Once the newly-added MLP layers are fine-tuned to this task (convolutional layers' parameters are not re-trained), we extract the $\mathbb{R}^{2048}$ vector representations of each IAPS stimulus from the penultimate MLP layer as the node features for each IAPS node in our heterogeneous graphs.

6.3.5.2. Relation (edge) features. As illustrated in Figure 6.3, the edges in our heterogeneous graphs were used to denote: (1) IAPS stimuli "belonging to" a particular picture category, (2) an individual's picture rating towards an IAPS stimulus, and

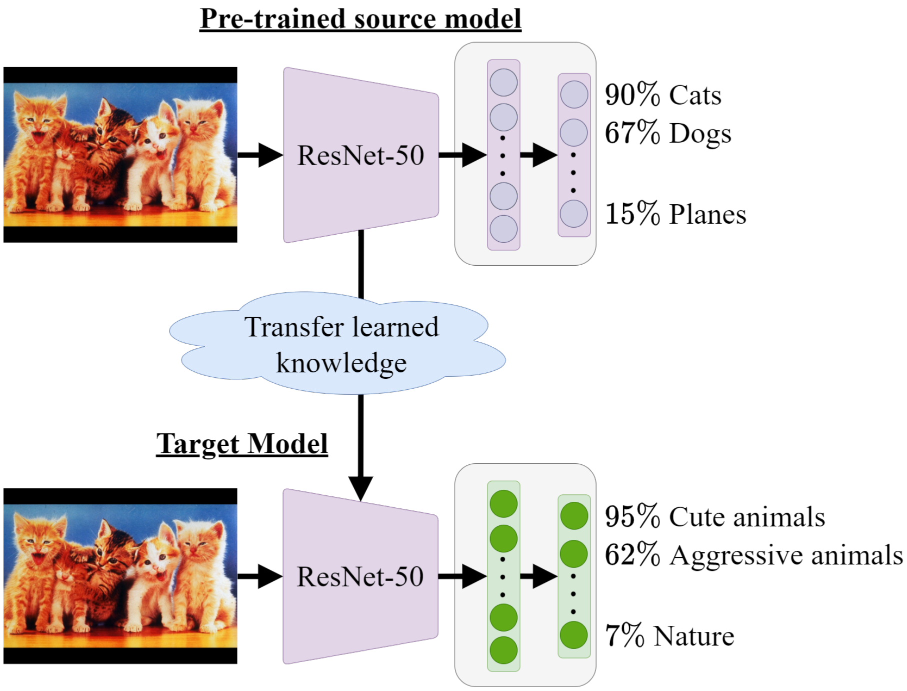


Figure 6.4. Example of fine-tuning a pre-trained ResNet-50 [**114**] CNN for the feature extraction process of IAPS stimuli. Purple layers are NN layers which are pre-trained on ImageNet [**256**], while the green layers are fine-tuned MLP layers which are trained from scratch (in conjunction with the frozen ResNet-50 layers) on categorizing IAPS stimuli into their respective picture categories.

(3) how an individual approaches/avoids a specific picture category. The edges for "belonging to" a picture category remained featureless for the purposes of this study. The edges denoting how an individual rates a picture used one-hot encoded vector representations, $\mathbb{R}^7$, to represent one of 7 possible ratings on the [-3, +3] Likert-like scale used for picture ratings. The edges representing how human participants approach/avoid IAPS picture categories use a concatenated vector representation of the previously discussed 15 features from RPT [**248, 220, 196**], which describe approach and avoidance behavior (e.g., conflict within a category, indifference, etc.).

6.4. Experimental Design

The gender assigned at birth classification task performed in this study uses the cohort of 4,105 human participants from the previous chapter (which discusses the subject populations and picture rating task in greater detail). Since we don't assume the features of node/edge types belong to the same distributions, we are free to use the most appropriate features to represent each type of node/edge. Demographic information for the human participants only included subject age as the feature to "learn upon," meanwhile the target demographic for prediction was gender assigned at birth. In this study we only focused on predicting male vs. female, given that of the 4,105 participants, only 3 indicated "other" in the demographic survey.

The International Affective Picture System (IAPS) stimuli used in this experimental construct are the same as those used in the picture rating task as well (belonging to the same 6 picture categories). By way of transfer learning, we use $\mathbb{R}^{2,048}$ vector representations of IAPS stimuli; which are obtained after fine-tuning the output layers of a pre-trained ResNet-50 [**114**] CNN model to categorize IAPS stimuli into their respective picture categories. Along with the feature extracted vector representations of IAPS stimuli, we also concatenate already-provided normative values of IAPS "norms," which were developed to provide researchers with values pertaining to each stimulus in emotional evocation, specifically for arousal, dominance, and valence.

Between each participant and picture stimulus, the "picture rating" edge features consisted of $\mathbb{R}^7$ one-hot encoded vectors, used to represent an individual's rating for each given stimulus on the 7-point [-3, +3] Likert scale. Between participants and picture category nodes, we used a concatenation of the 15 RPT features discussed in the previous chapter. Lastly, the “belonging to” edges connecting IAPS stimuli to their respective picture categories remained featureless. This did not pose a problem since part of the HGT paradigm involves constructing learnable parameters that are specific to meta-relations (i.e., there isn't a dependence on edge features *per se*).

The dataset in this experiment was divided into training and testing components using a 80-20% split. Specifically, we use a stratified 5-fold cross-validation over the human participant population.

Heterogeneous graphs in this experiment were constructed using the PyTorch Geometric [**257**] library, which contains a myriad of useful functions and implementations for constructing and designing graph neural networks (GNNs) as well. Heterogeneous graph transformer (HGT) layers, directly from PyTorch Geometric, were also utilized in this experiment. Specifically, 4 HGT layers (i.e., receptive field) were implemented, with 64 as the hidden dimension throughout the GNN. For multi-headed attention, we set the head number as 4. At the output-end of our NN architecture, we add a MLP which takes in the node embeddings of each human participant node (GNN outputs node embeddings for all nodes in the input graph) to perform the *semi-supervised* node classification task of predicting gender assigned at birth.

To handle large-scale graph data with our large subject population, we use the heterogeneous graph sampling (HGSampling), algorithm proposed by Hu *et al.* [**235**].

This allows us to sample neighborhoods (sub-graphs) of the overall heterogeneous interactome, as inputs to our GNN throughout training. Our approach optimizes a standard binary cross-entropy loss function via the AdamW optimizer [**258**], using a Cosine Annealing Learning Rate Scheduler [**259**]. We train our GNNs for 200 epochs and select the one with the lowest validation loss as the reported model. We use the default parameters used in the GNN literature and do not tune hyper-parameters.

6.5. Results & Discussion

We report the accuracy, F-score, precision, and recall for each fold of our 5-fold cross-validation in Table 6.1. The highest performing fold in our cross-validation resulted in a 86.24% accuracy, 0.8082 F-score, 0.7615 precision score, and 0.8609 recall (4th fold); along with the average of those scores across all 5 folds: 85.15%, 0.7908, 0.7388, 0.8510 respectively. Like the work of Hu *et al.*, where HGTs are introduced, we observe a decent performance on the node classification task by modeling heterogeneous relations according to their meta-relation schema, thus providing a

Table 6.1. Classification results of gender assigned at birth task

Fold	**Accuracy**	**F-score**	**Precision**	**Recall**
1	0.8418	0.7731	0.7104	0.8479
2	0.8550	0.7959	0.7423	0.8577
3	0.8463	0.7862	0.7423	0.8355
4	0.8624	0.8082	0.7615	0.8609
5	0.8519	0.7909	0.7374	0.8527
Average	0.8515	0.7908	0.7388	0.8510

better generalization for learning complex interactions with varying entity and relation types. This outcome, solely based on using human behavior, falls in line with multiple studies which utilize braining imaging on the same gender prediction task [**260, 261, 262**]. The results of this experiment provide justification for further experimentation into studying and leveraging *relational inductive bias* with GNNs for analyzing human interaction.

This dissertation has looked at multiple applications involving graph neural networks (GNNs) and graph data. The first study looked into applying GNNs towards Alzheimer's classification using neuroimaging. The following study improved upon that work by providing a mesh-specific GNN construct for the purposes of using a single GNN feature extractor for discriminative or generative tasks using trimeshes. Before the final study, we provide a detailed survey on a simple picture rating task that is then abstracted using a graph formalism to make predictions about human demographics in this final chapter. In the end, the benefits of this work can carry on past the confines of this dissertation, and hopefully provide a solid foundation for future works on graph-based machine learning in neuroimaging and human reward behavior going forward.

Bibliography

[1] I. S. Dhillon, Y. Guan, and B. Kulis, "Weighted graph cuts without eigenvectors a multilevel approach," *IEEE transactions on pattern analysis and machine intelligence*, vol. 29, no. 11, pp. 1944–1957, 2007.

[2] J. L. Hintze and R. D. Nelson, "Violin plots: A box plot-density trace synergism," *The American Statistician*, vol. 52, p. 181, 5 1998.

[3] M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," in *Proceedings of the 30th International Conference on Neural Information Processing Systems*, 2016, pp. 3844–3852.

[4] A. Ranjan, T. Bolkart, S. Sanyal, and M. J. Black, "Generating 3d faces using convolutional mesh autoencoders," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 704–720.

[5] M. Garland and P. S. Heckbert, "Surface simplification using quadric error metrics," in *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, 1997, pp. 209–216.

[6] T. R. Insel, P. Y. Collins, and S. E. Hyman, "Darkness invisible: The hidden global costs of mental illness," *Foreign Affairs*, vol. 94, no. 1, pp. 127–135, 2015.

[7] A. J. Blood, D. V. Iosifescu, N. Makris, R. H. Perlis, D. N. Kennedy, D. D. Dougherty, B. W. Kim, M. J. Lee, S. Wu, S. Lee *et al.*, "Microstructural abnormalities in subcortical reward circuitry of subjects with major depressive disorder," *PloS one*, vol. 5, no. 11, p. e13945, 2010.

[8] D. A. Pizzagalli, A. J. Holmes, D. G. Dillon, E. L. Goetz, J. L. Birk, R. Bogdan, D. D. Dougherty, D. V. Iosifescu, S. L. Rauch, and M. Fava, "Reduced caudate and nucleus accumbens response to rewards in unmedicated individuals with

major depressive disorder," *American Journal of Psychiatry*, vol. 166, no. 6, pp. 702–710, 2009.

[9] W.-N. Zhang, S.-H. Chang, L.-Y. Guo, K.-L. Zhang, and J. Wang, "The neural correlates of reward-related processing in major depressive disorder: a meta-analysis of functional magnetic resonance imaging studies," *Journal of affective disorders*, vol. 151, no. 2, pp. 531–539, 2013.

[10] S. O'Dea, "Number of smartphone subscriptions worldwide from 2016 to 2027," 9 2021. [Online]. Available: https://www.statista.com/statistics/330695/number-of-smartphone-users-worldwide/

[11] Pew Research Center, "Mobile fact sheet," 4 2021. [Online]. Available: https://www.pewresearch.org/internet/fact-sheet/mobile/

[12] A. V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing*, 3rd ed. USA: Prentice Hall Press, 2009.

[13] W. M. Siebert, *Circuits, signals, and systems.* MIT press, 1986.

[14] S. K. Mitra and Y. Kuo, *Digital signal processing: a computer-based approach.* McGraw-Hill New York, 2006, vol. 2.

[15] A. Sandryhaila and J. M. Moura, "Discrete signal processing on graphs," *IEEE transactions on signal processing*, vol. 61, no. 7, pp. 1644–1656, 2013.

[16] A. Ortega, P. Frossard, J. Kovačević, J. M. Moura, and P. Vandergheynst, "Graph signal processing: Overview, challenges, and applications," *Proceedings of the IEEE*, vol. 106, no. 5, pp. 808–828, 2018.

[17] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE signal processing magazine*, vol. 30, no. 3, pp. 83–98, 2013.

[18] A. Sandryhaila and J. M. Moura, "Big data analysis with signal processing on graphs: Representation and processing of massive data sets with irregular structure," *IEEE Signal Processing Magazine*, vol. 31, no. 5, pp. 80–90, 2014.

[19] B. Girault, P. Gonçalves, and É. Fleury, "Translation on graphs: An isometric shift operator," *IEEE Signal Processing Letters*, vol. 22, no. 12, pp. 2416–2420, 2015.

[20] A. Gavili and X.-P. Zhang, "On the shift operator, graph frequency, and optimal filtering in graph signal processing," *IEEE Transactions on Signal Processing*, vol. 65, no. 23, pp. 6303–6318, 2017.

[21] A. A. Shvets, A. Rakhlin, A. A. Kalinin, and V. I. Iglovikov, "Automatic instrument segmentation in robot-assisted surgery using deep learning," in *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA).* IEEE, 2018, pp. 624–628.

[22] Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger, "3d u-net: learning dense volumetric segmentation from sparse annotation," in *International conference on medical image computing and computer-assisted intervention.* Springer, 2016, pp. 424–432.

[23] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-net: Fully convolutional neural networks for volumetric medical image segmentation," in *2016 fourth international conference on 3D vision (3DV).* IEEE, 2016, pp. 565–571.

[24] W. E. Lorensen and H. E. Cline, "Marching cubes: A high resolution 3d surface construction algorithm," *ACM siggraph computer graphics*, vol. 21, no. 4, pp. 163–169, 1987.

[25] O. Sorkine, "Laplacian mesh processing," *Eurographics (STARs)*, vol. 29, 2005.

[26] A. Nealen, T. Igarashi, O. Sorkine, and M. Alexa, "Laplacian mesh optimization," in *Proceedings of the 4th international conference on Computer graphics and interactive techniques in Australasia and Southeast Asia*, 2006, pp. 381–389.

[27] M. Wardetzky, S. Mathur, F. Kälberer, and E. Grinspun, "Discrete laplace operators: no free lunch," in *Symposium on Geometry processing.* Aire-la-Ville, Switzerland, 2007, pp. 33–37.

[28] J. Bruna, W. Zaremba, A. Szlam, and Y. Lecun, "Spectral networks and locally connected networks on graphs," in *International Conference on Learning Representations (ICLR2014), CBLS, April 2014*, 2014.

[29] M. Henaff, J. Bruna, and Y. LeCun, "Deep convolutional networks on graph-structured data," *arXiv preprint arXiv:1506.05163*, 2015.

[30] D. K. Duvenaud, D. Maclaurin, J. Iparraguirre, R. Bombarell, T. Hirzel, A. Aspuru-Guzik, and R. P. Adams, "Convolutional networks on graphs for learning molecular fingerprints," in *Advances in Neural Information Processing Systems*, C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, Eds., vol. 28. Curran Associates, Inc., 2015.

[31] Y. Li, D. Tarlow, M. Brockschmidt, and R. Zemel, "Gated graph sequence neural networks," *arXiv preprint arXiv:1511.05493*, 2015.

[32] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *International Conference on Learning Representations (ICLR)*, 2017.

[33] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[34] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[35] R. Ying, R. He, K. Chen, P. Eksombatchai, W. L. Hamilton, and J. Leskovec, "Graph convolutional neural networks for web-scale recommender systems," in *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, 2018, pp. 974–983.

[36] P. D. Hoff, A. E. Raftery, and M. S. Handcock, "Latent space approaches to social network analysis," *Journal of the american Statistical association*, vol. 97, no. 460, pp. 1090–1098, 2002.

[37] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," *Advances in neural information processing systems*, vol. 26, 2013.

[38] W. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," *Advances in neural information processing systems*, vol. 30, 2017.

[39] J. Viš ˇnovskỳ, O. Kaššák, M. Kompan, and M. Bieliková, "The cold-start problem: Minimal users' activity estimation," in *Proceedings of the Workshop*

on Recommender Systems for Television and Online Video in conjuction with RecSyS, 2014.

[40] Y. Zhu, J. Lin, S. He, B. Wang, Z. Guan, H. Liu, and D. Cai, "Addressing the item cold-start problem by attribute-driven active learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 32, no. 4, pp. 631–644, 2019.

[41] F. M. Bianchi, D. Grattarola, L. Livi, and C. Alippi, "Hierarchical representation learning in graph neural networks with node decimation pooling," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–13, 2020.

[42] J. Lee, I. Lee, and J. Kang, "Self-attention graph pooling," in *International Conference on Machine Learning.* PMLR, 2019, pp. 3734–3743.

[43] R. Ying, J. You, C. Morris, X. Ren, W. L. Hamilton, and J. Leskovec, "Hierarchical graph representation learning with differentiable pooling," *arXiv preprint arXiv:1806.08804*, 2018.

[44] F. Milano, A. Loquercio, A. Rosinol, D. Scaramuzza, and L. Carlone, "Primal-dual mesh convolutional neural networks," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, Eds., vol. 33. Curran Associates, Inc., 2020, pp. 952–963.

[45] G. M. McKhann *et al.*, "The diagnosis of dementia due to alzheimer's disease: recommendations from the national institute on aging-alzheimer's association workgroups on diagnostic guidelines for alzheimer's disease," *Alzheimer's & dementia*, vol. 7, no. 3, pp. 263–269, 2011.

[46] J. Xu, K. D. Kochanek, S. L. Murphy, B. Tejada-Vera *et al.*, "National vital statistics reports," *National vital statistics reports*, vol. 58, no. 19, pp. 1–136, 2010.

[47] M. Heron and B. Smith, "Deaths: Leading causes for 2003. national vital statistics reports," *National Center for Health Statistics, Natl Vital Stat Rep 2007*, vol. 55, 2013.

[48] B. Tejada-Vera, "Mortality from alzheimer's disease in the united states: Data for 2000 and 2010. nchs data brief, no. 116," *National Center for Health Statistics: Hyattsville, MD*, 2013.

[49] D. J. Selkoe, “Alzheimer’s disease is a synaptic failure,” *Science*, vol. 298, no. 5594, pp. 789–791, 2002.

[50] C. R. Jack Jr, D. S. Knopman, W. J. Jagust, L. M. Shaw, P. S. Aisen, M. W. Weiner, R. C. Petersen, and J. Q. Trojanowski, “Hypothetical model of dynamic biomarkers of the alzheimer’s pathological cascade,” *The Lancet Neurology*, vol. 9, no. 1, pp. 119–128, 2010.

[51] S. W. Scheff, D. A. Price, F. A. Schmitt, and E. J. Mufson, “Hippocampal synaptic loss in early alzheimer’s disease and mild cognitive impairment,” *Neurobiology of aging*, vol. 27, no. 10, pp. 1372–1384, 2006.

[52] L. Rajendran and R. C. Paolicelli, “Microglia-mediated synapse loss in alzheimer’s disease,” *Journal of Neuroscience*, vol. 38, no. 12, pp. 2911–2919, 2018.

[53] P. Coupé, J. V. Manjón, E. Lanuza, and G. Catheline, “Lifespan changes of the human brain in alzheimer’s disease,” *Scientific reports*, vol. 9, no. 1, pp. 1–12, 2019.

[54] M. I. Miller, L. Younes, J. T. Ratnanather, T. Brown, H. Trinh, E. Postell, D. S. Lee, M.-C. Wang, S. Mori, R. O’Brien *et al.*, “The diffeomorphometry of temporal lobe structures in preclinical alzheimer’s disease,” *NeuroImage: Clinical*, vol. 3, pp. 352–360, 2013.

[55] C. Bernard, C. Helmer, B. Dilharreguy, H. Amieva, S. Auriacombe, J.-F. Dartigues, M. Allard, and G. Catheline, “Time course of brain volume changes in the preclinical phase of alzheimer’s disease,” *Alzheimer’s & Dementia*, vol. 10, no. 2, pp. 143–151, 2014.

[56] P. Coupé, V. S. Fonov, C. Bernard, A. Zandifar, S. F. Eskildsen, C. Helmer, J. V. Manjón, H. Amieva, J.-F. Dartigues, M. Allard *et al.*, “Detection of alzheimer’s disease signature in mr images seven years before conversion to dementia: Toward an early individual prognosis,” *Human brain mapping*, vol. 36, no. 12, pp. 4758–4770, 2015.

[57] T. den Heijer, F. van der Lijn, P. J. Koudstaal, A. Hofman, A. van der Lugt, G. P. Krestin, W. J. Niessen, and M. M. Breteler, “A 10-year follow-up of hippocampal volume on magnetic resonance imaging in early dementia and cognitive decline,” *Brain*, vol. 133, no. 4, pp. 1163–1172, 2010.

[58] F. Kumfor, L.-A. Sapey-Triomphe, C. E. Leyton, J. R. Burrell, J. R. Hodges, and O. Piguet, “Degradation of emotion processing ability in corticobasal syndrome and alzheimer’s disease,” *Brain*, vol. 137, no. 11, pp. 3061–3072, 2014.

[59] M. De Leon, A. George, L. Stylopoulos, G. Smith, and D. Miller, “Early marker for alzheimer’s disease: the atrophic hippocampus.” *Lancet (London, England)*, vol. 2, no. 8664, pp. 672–673, 1989.

[60] B. Hyman, G. Van Hoesen, A. Damasio, and C. Barnes, “Alzheimer’s disease: cell-specific pathology isolates the hippocampal formation,” *Science*, vol. 225, no. 4667, pp. 1168–1170, 1984.

[61] R. D. Terry, E. Masliah, D. P. Salmon, N. Butters, R. DeTeresa, R. Hill, L. A. Hansen, and R. Katzman, “Physical basis of cognitive alterations in alzheimer’s disease: Synapse loss is the major correlate of cognitive impairment,” *Annals of Neurology*, vol. 30, no. 4, pp. 572–580, 1991.

[62] M. Tondelli, G. K. Wilcock, P. Nichelli, C. A. De Jager, M. Jenkinson, and G. Zamboni, “Structural mri changes detectable up to ten years before clinical alzheimer’s disease,” *Neurobiology of Aging*, vol. 33, no. 4, pp. 825.e25–825.e36, 2012.

[63] S. C. Cramer, M. Sur, B. H. Dobkin, C. O’Brien, T. D. Sanger, J. Q. Trojanowski, J. M. Rumsey, R. Hicks, J. Cameron, D. Chen, W. G. Chen, L. G. Cohen, C. deCharms, C. J. Duffy, G. F. Eden, E. E. Fetz, R. Filart, M. Freund, S. J. Grant, S. Haber, P. W. Kalivas, B. Kolb, A. F. Kramer, M. Lynch, H. S. Mayberg, P. S. McQuillen, R. Nitkin, A. Pascual-Leone, P. Reuter-Lorenz, N. Schiff, A. Sharma, L. Shekim, M. Stryker, E. V. Sullivan, and S. Vinogradov, “Harnessing neuroplasticity for clinical applications,” *Brain*, vol. 134, no. 6, pp. 1591–1609, 04 2011.

[64] N. L. Hill, A. M. Kolanowski, and D. J. Gill, “Plasticity in early alzheimer’s disease: an opportunity for intervention,” *Topics in geriatric rehabilitation*, vol. 27, no. 4, p. 257, 2011.

[65] H. L. Rutenberg, H. Schwartz, and L. A. Soloff, “Norepinephrine-and heparin-induced changes in plasma free fatty acids: a comparison between patients with ischemic heart disease and normal young adults,” *American heart journal*, vol. 76, no. 2, pp. 183–192, 1968.

[66] A. Akaike, "Preclinical evidence of neuroprotection by cholinesterase inhibitors," *Alzheimer Disease & Associated Disorders*, vol. 20, pp. S8–S11, 2006.

[67] A. Akaike, Y. Takada-Takatori, T. Kume, and Y. Izumi, "Mechanisms of neuroprotective effects of nicotine and acetylcholinesterase inhibitors: role of $\alpha 4$ and $\alpha 7$ receptors in neuroprotection," *Journal of Molecular Neuroscience*, vol. 40, no. 1, pp. 211–216, 2010.

[68] P. Anand and B. Singh, "A review on cholinesterase inhibitors for alzheimer's disease," *Archives of pharmacal research*, vol. 36, no. 4, pp. 375–399, 2013.

[69] A. Kumar, A. Singh, and Ekavali, "A review on alzheimer's disease pathophysiology and its management: an update," *Pharmacological Reports*, vol. 67, no. 2, pp. 195–203, 2015.

[70] P. Golland, W. E. L. Grimson, M. E. Shenton, and R. Kikinis, "Detection and analysis of statistical differences in anatomical shape," *Medical image analysis*, vol. 9, no. 1, pp. 69–86, 2005.

[71] E. Gerardin, G. Chételat, M. Chupin, R. Cuingnet, B. Desgranges, H.-S. Kim, M. Niethammer, B. Dubois, S. Lehéricy, L. Garnero *et al.*, "Multidimensional classification of hippocampal shape features discriminates alzheimer's disease and mild cognitive impairment from normal aging," *Neuroimage*, vol. 47, no. 4, pp. 1476–1486, 2009.

[72] K.-k. Shen, J. Fripp, F. Mériaudeau, G. Chételat, O. Salvado, P. Bourgeat, A. D. N. Initiative *et al.*, "Detecting global and local hippocampal shape changes in alzheimer's disease using statistical shape models," *Neuroimage*, vol. 59, no. 3, pp. 2155–2166, 2012.

[73] G. Gerig, M. Styner, D. Jones, D. Weinberger, and J. Lieberman, "Shape analysis of brain ventricles using spharm," in *Proceedings IEEE Workshop on Mathematical Methods in Biomedical Image Analysis (MMBIA 2001)*. IEEE, 2001, pp. 171–178.

[74] G. Gerig, M. Styner, M. E. Shenton, and J. A. Lieberman, "Shape versus size: Improved understanding of the morphology of brain structures," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2001, pp. 24–32.

[75] M. Styner, J. A. Lieberman, R. K. McClure, D. R. Weinberger, D. W. Jones, and G. Gerig, "Morphometric analysis of lateral ventricles in schizophrenia and healthy controls regarding genetic and disease-specific factors," *Proceedings of the National Academy of Sciences*, vol. 102, no. 13, pp. 4872–4877, 2005.

[76] T. B. Terriberry, S. C. Joshi, and G. Gerig, "Hypothesis testing with nonlinear shape models," in *Biennial International Conference on Information Processing in Medical Imaging.* Springer, 2005, pp. 15–26.

[77] E. Luders, P. M. Thompson, K. Narr, A. W. Toga, L. Jancke, and C. Gaser, "A curvature-based approach to estimate local gyrification on the cortical surface," *Neuroimage*, vol. 29, no. 4, pp. 1224–1230, 2006.

[78] C. Wachinger, P. Golland, W. Kremen, B. Fischl, and M. Reuter, "Brainprint: A discriminative characterization of brain morphology," *NeuroImage*, vol. 109, pp. 232–248, 2015.

[79] M. Reuter, F.-E. Wolter, and N. Peinecke, "Laplace–beltrami spectra as 'shape-dna'of surfaces and solids," *Computer-Aided Design*, vol. 38, no. 4, pp. 342–366, 2006.

[80] D. S. Marcus, T. H. Wang, J. Parker, J. G. Csernansky, J. C. Morris, and R. L. Buckner, "Open access series of imaging studies (oasis): cross-sectional mri data in young, middle aged, nondemented, and demented older adults," *Journal of cognitive neuroscience*, vol. 19, no. 9, pp. 1498–1507, 2007.

[81] D. S. Marcus, A. F. Fotenos, J. G. Csernansky, J. C. Morris, and R. L. Buckner, "Open access series of imaging studies: longitudinal mri data in nondemented and demented older adults," *Journal of cognitive neuroscience*, vol. 22, no. 12, pp. 2677–2684, 2010.

[82] P. LaMontagne, T. Benzinger, J. Morris, S. Keefe, R. Hornbeck, C. Xiong, and D. Marcus, "Oasis-3: Longitudinal neuroimaging," *Clinical, and Cognitive Dataset for Normal Aging and Alzheimer Disease. medRxiv*, vol. 2013, 2019.

[83] R. C. Petersen, P. Aisen, L. A. Beckett, M. Donohue, A. Gamst, D. J. Harvey, C. Jack, W. Jagust, L. Shaw, A. Toga *et al.*, "Alzheimer's disease neuroimaging initiative (adni): clinical characterization," *Neurology*, vol. 74, no. 3, pp. 201–209, 2010.

[84] C. Gaser, K. Franke, S. Klöppel, N. Koutsouleris, H. Sauer, A. D. N. Initiative *et al.*, "Brainage in mild cognitive impaired patients: predicting the conversion to alzheimer's disease," *PloS one*, vol. 8, no. 6, p. e67346, 2013.

[85] C. Wachinger, K. Batmanghelich, P. Golland, and M. Reuter, "Brainprint in the computer-aided diagnosis of alzheimer's disease," in *Proceedings MICCAI workshop challenge on computer-aided diagnosis of dementia based on structural MRI data, Boston, MA, USA*. Citeseer, 2014.

[86] C. Wachinger, P. Golland, and M. Reuter, "Brainprint: Identifying subjects by their brain," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2014, pp. 41–48.

[87] B. Ng, M. Toews, S. Durrleman, and Y. Shi, "Shape analysis for brain structures," *Shape Analysis in Medical Image Analysis*, pp. 3–49, 2014.

[88] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. Van Der Laak, B. Van Ginneken, and C. I. Sánchez, "A survey on deep learning in medical image analysis," *Medical image analysis*, vol. 42, pp. 60–88, 2017.

[89] B. Gutiérrez-Becker, I. Sarasua, and C. Wachinger, "Discriminative and generative models for anatomical shape analysis on point clouds with deep neural networks," *Medical Image Analysis*, vol. 67, p. 101852, 2021.

[90] A. Bessadok and I. Rekik, "Intact connectional morphometricity learning using multi-view morphological brain networks with application to autism spectrum disorder," in *International Workshop on Connectomics in Neuroimaging*. Springer, 2018, pp. 38–46.

[91] A. Fornito, A. Zalesky, and M. Breakspear, "The connectomics of brain disorders," *Nature Reviews Neuroscience*, vol. 16, no. 3, pp. 159–172, 2015.

[92] A. S. Göktaş, A. Bessadok, and I. Rekik, "Residual embedding similarity-based network selection for predicting brain network evolution trajectory from a single observation," in *International Workshop on PRedictive Intelligence In MEdicine*. Springer, 2020, pp. 12–23.

[93] M. B. Gurbuz and I. Rekik, "Deep graph normalizer: A geometric deep learning approach for estimating connectional brain templates," in *International*

Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, 2020, pp. 155–165.

[94] A. Nebli and I. Rekik, "Gender differences in cortical morphological networks," *Brain imaging and behavior*, vol. 14, no. 5, pp. 1831–1839, 2020.

[95] J. Yang, Q. Zhu, R. Zhang, J. Huang, and D. Zhang, "Unified brain network with functional and structural data," in *International Conference on Medical Image Computing and Computer-Assisted Intervention.* Springer, 2020, pp. 114–123.

[96] E. A. Azcona, P. Besson, Y. Wu, A. Punjabi, A. Martersteck, A. Dravid, T. B. Parrish, S. K. Bandt, and A. K. Katsaggelos, "Interpretation of brain morphology in association to alzheimer's disease dementia classification using graph convolutional networks on triangulated meshes," in *Shape in Medical Imaging*, M. Reuter, C. Wachinger, H. Lombaert, B. Paniagua, O. Goksel, and I. Rekik, Eds. Cham: Springer International Publishing, 2020, pp. 95–107.

[97] M. Ono, S. Kubik, and C. D. Abernathey, *Atlas of the cerebral sulci.* Thieme Medical Publishers, 1990.

[98] A. M. Kälin *et al.*, "Subcortical shape changes, hippocampal atrophy and cortical thinning in future Alzheimer's disease patients," *Frontiers in Aging Neuroscience*, 2017.

[99] T. Liu *et al.*, "Cortical gyrification and sulcal spans in early stage Alzheimer's disease," *PLoS ONE*, 2012.

[100] J. Pacheco *et al.*, "Greater cortical thinning in normal older adults predicts later cognitive impairment," *Neurobiology of Aging*, vol. 36, no. 2, pp. 903–908, 2015.

[101] L. W. De Jong *et al.*, "Strongly reduced volumes of putamen and thalamus in Alzheimer's disease: An MRI study," *Brain*, vol. 131, no. 12, pp. 3277–3285, 2008.

[102] S. Derflinger *et al.*, "Grey-matter atrophy in Alzheimer's disease is asymmetric but not lateralized," *Journal of Alzheimer's Disease*, vol. 25, no. 2, pp. 347–357, 2011.

[103] D. Zhang and D. Shen, "Multi-modal multi-task learning for joint prediction of multiple regression and classification variables in alzheimer's disease," *NeuroImage*, vol. 59, no. 2, pp. 895–907, 2012.

[104] F. Liu, C.-Y. Wee, H. Chen, and D. Shen, "Inter-modality relationship constrained multi-modality multi-task feature selection for alzheimer's disease and mild cognitive impairment identification," *NeuroImage*, vol. 84, pp. 466–475, 2014.

[105] I. Beheshti *et al.*, "Classification of alzheimer's disease and prediction of mild cognitive impairment-to-alzheimer's conversion from structural magnetic resource imaging using feature ranking and a genetic algorithm," *Computers in biology and medicine*, vol. 83, pp. 109–119, 2017.

[106] R. Li, W. Zhang, H.-I. Suk, L. Wang, J. Li, D. Shen, and S. Ji, "Deep learning based imaging data completion for improved brain disease diagnosis," in *International Conference on Medical Image Computing and Computer-Assisted Intervention.* Springer, 2014, pp. 305–312.

[107] A. Punjabi *et al.*, "Neuroimaging modality fusion in Alzheimer's classification using convolutional neural networks," *PLoS ONE*, vol. 14, no. 12, pp. 1–14, 2019.

[108] J. Masci, D. Boscaini, M. M. Bronstein, and P. Vandergheynst, "Geodesic Convolutional Neural Networks on Riemannian Manifolds," in *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2015-Febru, 2015, pp. 832–840.

[109] S. Parisot, S. I. Ktena, E. Ferrante, M. Lee, R. Guerrero, B. Glocker, and D. Rueckert, "Disease prediction using graph convolutional networks: application to autism spectrum disorder and alzheimer's disease," *Medical image analysis*, vol. 48, pp. 117–130, 2018.

[110] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and S. Y. Philip, "A comprehensive survey on graph neural networks," *IEEE transactions on neural networks and learning systems*, 2020.

[111] G. B. Arfken, H. J. Weber, and F. E. Harris, *Mathematical Methods for Physicists*, 3rd ed. Academic Press, 2013.

[112] B. Fischl, "FreeSurfer." *NeuroImage*, vol. 62, no. 2, pp. 774–81, 8 2012.

[113] P. Besson, R. Lopes, X. Leclerc, P. Derambure, and L. Tyvaert, "Intra-subject reliability of the high-resolution whole-brain structural connectome," *NeuroImage*, vol. 102, pp. 283–293, 2014.

[114] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[115] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2818–2826.

[116] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International conference on machine learning.* PMLR, 2015, pp. 448–456.

[117] T. N. Bui and C. Jones, "Finding good approximate vertex and edge partitions is np-hard," *Information Processing Letters*, vol. 42, no. 3, pp. 153–159, 1992.

[118] G. Karypis and V. Kumar, "A fast and high quality multilevel scheme for partitioning irregular graphs," *SIAM Journal on scientific Computing*, vol. 20, no. 1, pp. 359–392, 1998.

[119] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 22, no. 8, pp. 888–905, 2000.

[120] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 618–626.

[121] C. R. Jack Jr, M. A. Bernstein, N. C. Fox, P. Thompson, G. Alexander, D. Harvey, B. Borowski, P. J. Britson, J. L. Whitwell, C. Ward *et al.*, "The alzheimer's disease neuroimaging initiative (adni): Mri methods," *Journal of Magnetic Resonance Imaging: An Official Journal of the International Society for Magnetic Resonance in Medicine*, vol. 27, no. 4, pp. 685–691, 2008.

[122] Y. R. Fung, Z. Guan, R. Kumar, J. Y. Wu, and M. Fiterau, "Alzheimer's disease brain mri classification: Challenges and insights," *arXiv preprint arXiv:1906.04231*, 2019.

[123] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015.

[124] E. Westman, J.-S. Muehlboeck, and A. Simmons, "Combining mri and csf measures for classification of alzheimer's disease and prediction of mild cognitive impairment conversion," *Neuroimage*, vol. 62, no. 1, pp. 229–238, 2012.

[125] K. Hu, Y. Wang, K. Chen, L. Hou, and X. Zhang, "Multi-scale features extraction from baseline structure mri for mci patient classification and ad early diagnosis," *Neurocomputing*, vol. 175, pp. 132–145, 2016.

[126] I. Beheshti, H. Demirel, and H. Matsuda, "Classification of alzheimer's disease and prediction of mild cognitive impairment-to-alzheimer's conversion from structural magnetic resource imaging using feature ranking and a genetic algorithm," *Computers in Biology and Medicine*, vol. 83, pp. 109–119, 2017.

[127] E. Hosseini-Asl, R. Keynton, and A. El-Baz, "Alzheimer's disease diagnostics by adaptation of 3d convolutional network," in *2016 IEEE International Conference on Image Processing (ICIP)*, 2016, pp. 126–130.

[128] A. Gupta, M. Ayhan, and A. Maida, "Natural image bases to represent neuroimaging data," in *International conference on machine learning.* PMLR, 2013, pp. 987–994.

[129] B. Fischl *et al.*, "High-resolution intersubject averaging and a coordinate system for the cortical surface," *Human brain mapping*, vol. 8, no. 4, pp. 272–284, 1999.

[130] B. C. Dickerson, A. Bakkour, D. H. Salat, E. Feczko, J. Pacheco, D. N. Greve, F. Grodstein, C. I. Wright, D. Blacker, H. D. Rosas *et al.*, "The cortical signature of alzheimer's disease: regionally specific cortical thinning relates to symptom severity in very mild to mild ad dementia and is detectable in asymptomatic amyloid-positive individuals," *Cerebral cortex*, vol. 19, no. 3, pp. 497–510, 2009.

[131] P. Thompson, J. Moussai, S. Zohoori, A. Goldkorn, A. Khan, M. Mega, G. Small, J. Cummings, and A. Toga, "Cortical variability and asymmetry in normal aging and alzheimer's disease." *Cerebral Cortex (New York, NY: 1991)*, vol. 8, no. 6, pp. 492–509, 1998.

[132] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems*, Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Q. Weinberger, Eds., vol. 27. Curran Associates, Inc., 2014. [Online]. Available: https://proceedings.neurips.cc/paper/2014/file/5ca3e9b122f61f8f06494c97b1afccf3-Paper.pdf

[133] C. J. Brignell, I. L. Dryden, S. A. Gattone, B. Park, S. Leask, W. J. Browne, and S. Flynn, "Surface shape analysis with an application to brain surface asymmetry in schizophrenia," *Biostatistics*, vol. 11, no. 4, pp. 609–630, 2010.

[134] H. Kim, T. Mansi, and N. Bernasconi, "Disentangling hippocampal shape anomalies in epilepsy," *Frontiers in neurology*, vol. 4, p. 131, 2013.

[135] M. Shakeri, H. Lombaert, A. N. Datta, N. Oser, L. Létourneau-Guillon, L. V. Lapointe, F. Martin, D. Malfait, A. Tucholka, S. Lippé *et al.*, "Statistical shape analysis of subcortical structures using spectral matching," *Computerized Medical Imaging and Graphics*, vol. 52, pp. 58–71, 2016.

[136] M. Tondelli, G. K. Wilcock, P. Nichelli, C. A. De Jager, M. Jenkinson, and G. Zamboni, "Structural mri changes detectable up to ten years before clinical alzheimer's disease," *Neurobiology of aging*, vol. 33, no. 4, pp. 825–e25, 2012.

[137] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 652–660.

[138] M. M. Bronstein, J. Bruna, Y. LeCun, A. Szlam, and P. Vandergheynst, "Geometric deep learning: going beyond euclidean data," *IEEE Signal Processing Magazine*, vol. 34, no. 4, pp. 18–42, 2017.

[139] J. Wu, C. Zhaong, T. Xue, W. T. Freeman, and J. B. Tenenbaum, "Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling," in *Proceedings of the 30th International Conference on Neural Information Processing Systems*, ser. NIPS'16. Red Hook, NY, USA: Curran Associates Inc., 2016, p. 82–90.

[140] P. Achlioptas, O. Diamanti, I. Mitliagkas, and L. Guibas, "Learning representations and generative models for 3D point clouds," in *Proceedings of the 35th International Conference on Machine Learning*, ser. Proceedings of Machine

Learning Research, J. Dy and A. Krause, Eds., vol. 80. PMLR, 10–15 Jul 2018, pp. 40–49.

[141] G. Bouritsas, S. Bokhnyak, S. Ploumpis, M. Bronstein, and S. Zafeiriou, "Neural 3d morphable models: Spiral convolutional networks for 3d shape representation learning and generation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 7213–7222.

[142] N. Kolotouros, G. Pavlakos, and K. Daniilidis, "Convolutional mesh regression for single-image human shape reconstruction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4501–4510.

[143] O. Litany, A. Bronstein, M. Bronstein, and A. Makadia, "Deformable shape completion with graph convolutional autoencoders," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 1886–1895.

[144] N. Wang, Y. Zhang, Z. Li, Y. Fu, W. Liu, and Y.-G. Jiang, "Pixel2mesh: Generating 3d mesh models from single rgb images," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 52–67.

[145] U. Wickramasinghe, E. Remelli, G. Knott, and P. Fua, "Voxel2mesh: 3d mesh model generation from volumetric data," in *International Conference on Medical Image Computing and Computer-Assisted Intervention.* Springer, 2020, pp. 299–308.

[146] S. Gong, L. Chen, M. Bronstein, and S. Zafeiriou, "Spiralnet++: A fast and highly efficient mesh convolution operator," in *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, 2019, pp. 0–0.

[147] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++ deep hierarchical feature learning on point sets in a metric space," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 2017, pp. 5105–5114.

[148] K. Sohn, H. Lee, and X. Yan, "Learning structured output representation using deep conditional generative models," *Advances in neural information processing systems*, vol. 28, pp. 3483–3491, 2015.

[149] G. B. Frisoni, N. C. Fox, C. R. Jack, P. Scheltens, and P. M. Thompson, "The clinical use of structural mri in alzheimer disease," *Nature Reviews Neurology*, vol. 6, no. 2, pp. 67–77, 2010.

[150] S. Klöppel, C. M. Stonnington, C. Chu, B. Draganski, R. I. Scahill, J. D. Rohrer, N. C. Fox, C. R. Jack Jr, J. Ashburner, and R. S. Frackowiak, "Automatic classification of mr scans in alzheimer's disease," *Brain*, vol. 131, no. 3, pp. 681–689, 2008.

[151] Z. C. Marton, R. B. Rusu, and M. Beetz, "On fast surface reconstruction methods for large and noisy point clouds," in *2009 IEEE international conference on robotics and automation*. IEEE, 2009, pp. 3218–3223.

[152] G. TaubinÝ, "Geometric signal processing on polygonal meshes," *Proceedings of EUROGRAPHICS 2000: state of the art report*, 2000.

[153] A. Bessadok, M. A. Mahjoub, and I. Rekik, "Hierarchical adversarial connectomic domain alignment for target brain graph prediction and classification from a source graph," in *International Workshop on PRedictive Intelligence In MEdicine*. Springer, 2019, pp. 105–114.

[154] ——, "Brain graph synthesis by dual adversarial domain alignment and target graph prediction from a source graph," *Medical Image Analysis*, vol. 68, p. 101902, 2021.

[155] A. Sserwadda and I. Rekik, "Topology-guided cyclic brain connectivity generation using geometric deep learning," *Journal of Neuroscience Methods*, vol. 353, p. 108988, 2021.

[156] L. Zhang, L. Wang, and D. Zhu, "Recovering brain structural connectivity from functional connectivity via multi-gcn based generative adversarial network," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2020, pp. 53–61.

[157] H. Choi, H. Kang, D. S. Lee, A. D. N. Initiative *et al.*, "Predicting aging of brain metabolic topography using variational autoencoder," *Frontiers in aging neuroscience*, vol. 10, p. 212, 2018.

[158] D. P. Kingma and M. Welling, "Auto-Encoding Variational Bayes," in *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.

[159] J. M. Joyce, "Kullback-leibler divergence," *International Encyclopedia of Statistical Science*, pp. 720–722, 2011.

[160] I. Lim, A. Dielen, M. Campen, and L. Kobbelt, "A simple approach to intrinsic correspondence learning on unstructured 3d meshes," in *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, 2018, pp. 0–0.

[161] G. Corso, L. Cavalleri, D. Beaini, P. Liò, and P. Veličković, "Principal neighbourhood aggregation for graph nets," *Advances in Neural Information Processing Systems*, vol. 33, 2020.

[162] Y. Xie, S. Li, C. Yang, R. C.-W. Wong, and J. Han, "When do gnns work: Understanding and improving neighborhood aggregation," in *IJCAI'20: Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence,{IJCAI} 2020*, vol. 2020, no. 1, 2020.

[163] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl, "Neural message passing for quantum chemistry," in *International Conference on Machine Learning.* PMLR, 2017, pp. 1263–1272.

[164] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," *International Conference on Learning Representations (ICLR)*, 2016.

[165] L. Lu, Y. Shin, Y. Su, and G. E. Karniadakis, "Dying relu and initialization: Theory and numerical examples," *Communications in Computational Physics*, vol. 28, no. 5, pp. 1671–1706, 2020.

[166] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and accurate deep network learning by exponential linear units (elus)," *International Conference on Learning Representations (ICLR)*, 2015.

[167] Dawson-Haggerty *et al.*, "trimesh," 2019. [Online]. Available: https://trimsh.org/

[168] G. Van Rossum and F. L. Drake Jr, *Python tutorial.* Centrum voor Wiskunde en Informatica Amsterdam, The Netherlands, 1995.

[169] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," *arXiv preprint arXiv:1711.05101*, 2017.

[170] M. C. Corballis, "Left brain, right brain: facts and fantasies," *PLoS Biol*, vol. 12, no. 1, p. e1001767, 2014.

[171] P. T. Nelson, E. Head, F. A. Schmitt, P. R. Davis, J. H. Neltner, G. A. Jicha, E. L. Abner, C. D. Smith, L. J. Van Eldik, R. J. Kryscio *et al.*, "Alzheimer's disease is not "brain aging": neuropathological, genetic, and epidemiological human studies," *Acta neuropathologica*, vol. 121, no. 5, pp. 571–587, 2011.

[172] B. C. Dickerson, I. Goncharova, M. Sullivan, C. Forchetti, R. Wilson, D. Bennett, L. A. Beckett, and L. deToledo Morrell, "Mri-derived entorhinal and hippocampal atrophy in incipient and very mild alzheimer's disease," *Neurobiology of aging*, vol. 22, no. 5, pp. 747–754, 2001.

[173] H. Braak and E. Braak, "Neuropathological stageing of alzheimer-related changes," *Acta neuropathologica*, vol. 82, no. 4, pp. 239–259, 1991.

[174] B. Dubois, H. H. Feldman, C. Jacova, S. T. DeKosky, P. Barberger-Gateau, J. Cummings, A. Delacourte, D. Galasko, S. Gauthier, G. Jicha *et al.*, "Research criteria for the diagnosis of alzheimer's disease: revising the nincds–adrda criteria," *The Lancet Neurology*, vol. 6, no. 8, pp. 734–746, 2007.

[175] Y. Klein-Koerkamp, R. A Heckemann, K. T Ramdeen, O. Moreaud, S. Keignart, A. Krainik, A. Hammers, M. Baciu, P. Hot, A. disease Neuroimaging Initiative *et al.*, "Amygdalar atrophy in early alzheimer's disease," *Current Alzheimer Research*, vol. 11, no. 3, pp. 239–252, 2014.

[176] C. Ledig, A. Schuh, R. Guerrero, R. A. Heckemann, and D. Rueckert, "Structural brain imaging in alzheimer's disease and mild cognitive impairment: biomarker analysis and shared morphometry database," *Scientific reports*, vol. 8, no. 1, pp. 1–16, 2018.

[177] P. Besson, T. Parrish, A. K. Katsaggelos, and S. K. Bandt, "Geometric deep learning on brain shape predicts sex and age," *Computerized Medical Imaging and Graphics*, vol. 91, p. 101939, 2021.

[178] X. Long, L. Zhang, W. Liao, C. Jiang, B. Qiu, and A. D. N. Initiative, "Distinct laterality alterations distinguish mild cognitive impairment and alzheimer's disease from healthy aging: Statistical parametric mapping with high resolution mri," *Human brain mapping*, vol. 34, no. 12, pp. 3400–3410, 2013.

[179] P. M. Thompson, K. M. Hayashi, R. A. Dutton, M.-C. Chiang, A. D. Leow, E. R. Sowell, G. De Zubicaray, J. T. Becker, O. L. Lopez, H. J. Aizenstein *et al.*, "Tracking alzheimer's disease," *Annals of the New York Academy of Sciences*, vol. 1097, p. 183, 2007.

[180] L. Frings, S. Hellwig, T. S. Spehl, T. Bormann, R. Buchert, W. Vach, L. Minkova, B. Heimbach, S. Klöppel, and P. T. Meyer, "Asymmetries of amyloid-β burden and neuronal dysfunction are positively correlated in alzheimer's disease," *Brain*, vol. 138, no. 10, pp. 3089–3099, 2015.

[181] R. Barber, I. McKeith, C. Ballard, and J. O'Brien, "Volumetric mri study of the caudate nucleus in patients with dementia with lewy bodies, alzheimer's disease, and vascular dementia," *Journal of Neurology, Neurosurgery & Psychiatry*, vol. 72, no. 3, pp. 406–407, 2002.

[182] L. Ferrarini, W. M. Palm, H. Olofsen, M. A. van Buchem, J. H. Reiber, and F. Admiraal-Behloul, "Shape differences of the brain ventricles in alzheimer's disease," *Neuroimage*, vol. 32, no. 3, pp. 1060–1069, 2006.

[183] J. G. Keilp, G. E. Alexander, Y. Stern, and I. Prohovnik, "Inferior parietal perfusion, lateralization, and neuropsychological dysfunction in alzheimer's disease," *Brain and cognition*, vol. 32, no. 3, pp. 365–383, 1996.

[184] P. E. McKight and J. Najab, "Kruskal-wallis test," *The Corsini Encyclopedia of Psychology*, 2010.

[185] D. Mann, "The topographic distribution of brain atrophy in alzheimer's disease," *Acta neuropathologica*, vol. 83, no. 1, pp. 81–86, 1991.

[186] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[187] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 31, no. 1, 2017.

[188] J. M. Roe, D. Vidal-Piñeiro, Ø. Sørensen, A. M. Brandmaier, S. Düzel, H. A. Gonzalez, R. A. Kievit, E. Knights, S. Kühn, U. Lindenberger *et al.*, "Asymmetric thinning of the cerebral cortex across the adult lifespan is accelerated in alzheimer's disease," *Nature communications*, vol. 12, no. 1, pp. 1–11, 2021.

[189] C. Wachinger, D. H. Salat, M. Weiner, M. Reuter, and for the Alzheimer's Disease Neuroimaging Initiative, "Whole-brain analysis reveals increased neuroanatomical asymmetries in dementia for hippocampus and amygdala,"

Brain, vol. 139, no. 12, pp. 3253–3266, 10 2016. [Online]. Available: https://doi.org/10.1093/brain/aww243

[190] S. J. Crutch, J. M. Schott, G. D. Rabinovici, B. F. Boeve, S. F. Cappa, B. C. Dickerson, B. Dubois, N. R. Graff-Radford, P. Krolak-Salmon, M. Lehmann *et al.*, "Shining a light on posterior cortical atrophy," *Alzheimer's & Dementia*, vol. 9, no. 4, pp. 463–465, 2013.

[191] K. Lewin, "A dynamic theory of personality," *Development, Factor Analysis, and Validation*, 1935.

[192] T. C. Schneirla, "An evolutionary and developmental theory of biphasic processes underlying approach and withdrawal." *Nebraska symposium on motivation, 1959.*, pp. 1–42, 1959.

[193] ——, "Aspects of stimulation and organization in approach/withdrawal processes underlying vertebrate behavioral development," *Advances in the Study of Behavior*, vol. 1, pp. 1–74, 1 1965.

[194] W. M. Baum, "On two types of deviation from the matching law: bias and undermatching 1," *Journal of the experimental analysis of behavior*, vol. 22, no. 1, pp. 231–242, 1974.

[195] R. J. Herrnstein, "Secondary reinforcement and rate of primary reinforcement 1," *journal of the Experimental Analysis of Behavior*, vol. 7, no. 1, pp. 27–36, 1964.

[196] B. W. Kim, D. N. Kennedy, J. Lehár, M. J. Lee, A. J. Blood, S. Lee, R. H. Perlis, J. W. Smoller, R. Morris, M. Fava, and H. C. Breiter, "Recurrent, robust and scalable patterns underlie human approach and avoidance," *PLOS ONE*, vol. 5, p. e10613, 2010. [Online]. Available: https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0010613

[197] S. Lee, M. J. Lee, B. W. Kim, J. M. Gilman, J. K. Kuster, A. J. Blood, C. M. Kuhnen, and H. C. Breiter, "The commonality of loss aversion across procedures and stimuli," *PLOS ONE*, vol. 10, p. e0135216, 9 2015. [Online]. Available: https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0135216

[198] S. L. Livengood, J. P. Sheppard, B. W. Kim, E. C. Malthouse, J. E. Bourne, A. E. Barlow, M. J. Lee, V. Marin, K. P. O'Connor, J. G. Csernansky, M. P.

Block, A. J. Blood, and H. C. Breiter, “Keypress-based musical preference is both individual and lawful,” *Frontiers in Neuroscience*, vol. 11, p. 136, 2017.

[199] R. P. Feynman, *The Character of Physical Law.* British Broadcasting Corporation, 1965.

[200] I. Aharon, N. Etcoff, D. Ariely, C. F. Chabris, E. O’Connor, and H. C. Breiter, “Beautiful faces have variable reward value: fmri and behavioral evidence,” *Neuron*, vol. 32, pp. 537–551, 11 2001.

[201] V. Viswanathan, S. Lee, J. M. Gilman, B. W. Kim, N. Lee, L. Chamberlain, S. L. Livengood, K. Raman, M. J. Lee, J. Kuster, D. B. Stern, B. Calder, F. J. Mulhern, A. J. Blood, and H. C. Breiter, “Age-related striatal bold changes without changes in behavioral loss aversion,” *Frontiers in Human Neuroscience*, vol. 9, pp. 1–12, 4 2015.

[202] G. P. Gasic, J. W. Smoller, R. H. Perlis, M. Sun, S. Lee, B. W. Kim, M. J. Lee, D. J. Holt, A. J. Blood, N. Makris, D. K. Kennedy, R. D. Hoge, J. Calhoun, M. Fava, J. F. Gusella, and H. C. Breiter, “Bdnf, relative preference, and reward circuitry responses to emotional communication,” *American Journal of Medical Genetics, Part B: Neuropsychiatric Genetics*, vol. 150, pp. 762–781, 9 2009.

[203] R. H. Perlis, D. J. Holt, J. W. Smoller, A. J. Blood, S. Lee, B. W. Kim, M. J. Lee, M. Sun, N. Makris, D. K. Kennedy, K. Rooney, D. D. Dougherty, R. Hoge, J. F. Rosenbaum, M. Fava, J. Gusella, G. P. Gasic, and H. C. Breiter, “Association of a polymorphism near creb1 with differential aversion processing in the insula of healthy participants,” *Archives of General Psychiatry*, vol. 65, pp. 882–892, 8 2008. [Online]. Available: https://jamanetwork.com/journals/jamapsychiatry/fullarticle/210118

[204] N. Makris, G. P. Gasic, D. N. Kennedy, S. M. Hodge, J. R. Kaiser, M. J. Lee, B. W. Kim, A. J. Blood, A. E. Evins, L. J. Seidman, D. V. Iosifescu, S. Lee, C. Baxter, R. H. Perlis, J. W. Smoller, M. Fava, and H. C. Breiter, “Cortical thickness abnormalities in cocaine addiction – a reflection of both drug use and a pre-existing disposition to drug abuse?” *Neuron*, vol. 60, p. 174, 10 2008. [Online]. Available: /pmc/articles/PMC3772717//pmc/articles/PMC3772717/?report=abstracthttps://www.ncbi.nlm.nih.gov/pmc/articles/PMC3772717/

[205] I. Elman, D. Ariely, N. Mazar, I. Aharon, N. B. Lasko, M. L. Macklin, S. P. Orr, S. E. Lukas, and R. K. Pitman, “Probing reward function in post-traumatic

stress disorder with beautiful facial images," *Psychiatry Research*, vol. 135, pp. 179–183, 6 2005.

[206] B. Levy, D. Ariely, N. Mazar, W. Chi, S. Lukas, and I. Elman, "Gender differences in the motivational processing of facial beauty," *Learning and Motivation*, vol. 39, pp. 136–145, 5 2008.

[207] M. M. Strauss, N. Makris, I. Aharon, M. G. Vangel, J. Goodman, D. N. Kennedy, G. P. Gasic, and H. C. Breiter, "fmri of sensitization to angry faces," *NeuroImage*, vol. 26, pp. 389–413, 6 2005.

[208] V. Viswanathan, J. P. Sheppard, B. W. Kim, C. L. Plantz, H. Ying, M. J. Lee, K. Raman, F. J. Mulhern, M. P. Block, B. Calder, S. Lee, D. T. Mortensen, A. J. Blood, and H. C. Breiter, "A quantitative relationship between signal detection in attention and approach/avoidance behavior," *Frontiers in Psychology*, vol. 8, p. 122, 2 2017.

[209] R. Yamamoto, D. Ariely, W. Chi, D. D. Langleben, and I. Elman, "Gender differences in the motivational processing of babies are determined by their facial attractiveness," *PLOS ONE*, vol. 4, p. e6042, 6 2009. [Online]. Available: https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0006042

[210] K. C. Berridge and T. E. Robinson, "The mind of an addicted brain: Neural sensitization of wanting versus liking:," *https://doi.org/10.1111/1467-8721.ep10772316*, vol. 4, pp. 71–75, 6 2016. [Online]. Available: https://journals.sagepub.com/doi/10.1111/1467-8721.ep10772316

[211] D. Kahneman and A. Tversky, "Prospect theory: An analysis of decision under risk," *Econometrica*, vol. 47, p. 263, 3 1979.

[212] H. Markowitz, "Portfolio selection," *The Journal of Finance*, vol. 7, pp. 77–91, 3 1952.

[213] A. Tversky and D. Kahneman, "Advances in prospect theory: Cumulative representation of uncertainty," *Journal of Risk and Uncertainty*, vol. 5, pp. 297–323, 10 1992.

[214] P. J. Lang, M. M. Bradley, B. N. Cuthbert *et al.*, "International affective picture system (iaps): Technical manual and affective ratings," *NIMH Center for the Study of Emotion and Attention*, vol. 1, no. 39-58, p. 3, 1997.

[215] P. J. Lang, M. M. Bradley, and B. N. Cuthbert, "International affective picture system (iaps): Affective ratings of pictures and instruction manual. technical report a-8," 2008.

[216] K. Kroenke, R. L. Spitzer, and J. B. Williams, "The phq-9: validity of a brief depression severity measure," *Journal of general internal medicine*, vol. 16, pp. 606–613, 2001. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/11556941/

[217] C. D. Spielberger, L. Gorsuch, L. Laux, P. Glanzmann, and P. Schaffner, "Das state-trait-angstinventar: Stai," *Beltz Test*, 2001.

[218] M. L. Dennis, Y. F. Chan, and R. R. Funk, "Development and validation of the gain short screener (gss) for internalizing, externalizing and substance use disorders and crime/violence problems among adolescents and adults," *The American Journal on Addictions*, vol. 15, pp. s80–s91, 2006. [Online]. Available: https://onlinelibrary.wiley.com/doi/full/10.1080/10550490601006055https://onlinelibrary.wiley.com/doi/abs/10.1080/10550490601006055https://onlinelibrary.wiley.com/doi/10.1080/10550490601006055

[219] C. E. Shannon and W. Weaver, *The Mathematical Theory of Communication.* University of Illinois Press, 1949, vol. 1.

[220] H. C. Breiter and B. W. Kim, "Recurrent and robust patterns underlying human relative preference and associations with brain circuitry plus genetics," *Design Principles in Biology (University of Minnesota, Institute of Mathematics and its Applications)*, vol. 4, pp. 21–25, 2008.

[221] H. T. Banks and H. T. Tran, *Mathematical and experimental modeling of physical and biological processes.* CRC Press, 2009.

[222] R. Zhang, T. J. Brennan, and A. W. Lo, "The origin of risk aversion," *Proceedings of the National Academy of Sciences*, vol. 111, pp. 17 777–17 782, 12 2014.

[223] K. Kendall and M. George, "Kruskal-wallis test," *The Concise Encyclopedia of Statistics*, pp. 288–290, 2 2008. [Online]. Available: https://link.springer.com/referenceworkentry/10.1007/978-0-387-32833-1_216

[224] A. Dinno, "Nonparametric pairwise multiple comparisons in independent groups using dunn's test:," *https://doi.org/10.1177/1536867X1501500117*,

vol. 15, pp. 292–300, 4 2015. [Online]. Available: https://journals.sagepub.com/doi/10.1177/1536867X1501500117

[225] K. Kendall and M. George, “Kolmogorov–smirnov test,” *The Concise Encyclopedia of Statistics*, pp. 283–287, 2 2008. [Online]. Available: https://link.springer.com/referenceworkentry/10.1007/978-0-387-32833-1_214

[226] S. M. Tom, C. R. Fox, C. Trepel, and R. A. Poldrack, “The neural basis of loss aversion in decision-making under risk,” *Science*, vol. 315, pp. 515–518, 1 2007.

[227] M. Buhrmester, T. Kwang, and S. D. Gosling, “Amazon’s mechanical turk: A new source of inexpensive, yet high-quality data?” *Methodological issues and strategies in clinical research (4th ed.).*, pp. 133–139, 12 2015. [Online]. Available: /record/2015-32022-009

[228] K. Casler, L. Bickel, and E. Hackett, “Separate but equal? a comparison of participants and data gathered via amazon’s mturk, social media, and face-to-face behavioral testing,” *Computers in Human Behavior*, vol. 29, pp. 2156–2160, 11 2013.

[229] D. J. Hauser and N. Schwarz, “Attentive turkers: Mturk participants perform better on online attention checks than do subject pool participants,” *Behavior Research Methods*, vol. 48, pp. 400–407, 3 2016. [Online]. Available: https://link.springer.com/article/10.3758/s13428-015-0578-z

[230] W. Mason and S. Suri, “Conducting behavioral research on amazon’s mechanical turk,” *Behavior research methods*, vol. 44, pp. 1–23, 3 2012. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/21717266/

[231] G. Paolacci, J. Chandler, and P. G. Ipeirotis, “Running experiments on amazon mechanical turk,” *Judgement and Decision Making*, vol. 5, pp. 411–419, 8 2010.

[232] J. Chandler, C. Rosenzweig, A. J. Moss, J. Robinson, and L. Litman, “Online panels in social science research: Expanding sampling methods beyond mechanical turk,” *Behavior Research Methods*, vol. 51, pp. 2022–2038, 10 2019. [Online]. Available: /record/2019-63223-006

[233] J. H. Cheung, D. K. Burns, R. R. Sinclair, and M. Sliter, “Amazon mechanical turk in organizational psychology: An evaluation and practical recommendations,” *Journal of Business and Psychology*, vol. 32, pp. 347–361, 8 2017. [Online]. Available: /record/2016-32638-001

[234] M. J. Crump, J. V. McDonnell, and T. M. Gureckis, "Evaluating amazon's mechanical turk as a tool for experimental behavioral research," *PLOS ONE*, vol. 8, p. e57410, 3 2013. [Online]. Available: https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0057410

[235] Z. Hu, Y. Dong, K. Wang, and Y. Sun, "Heterogeneous graph transformer," in *Proceedings of The Web Conference 2020*, 2020, pp. 2704–2710.

[236] Y. Sun, J. Han, X. Yan, P. S. Yu, and T. Wu, "Pathsim: Meta path-based top-k similarity search in heterogeneous information networks," *Proceedings of the VLDB Endowment*, vol. 4, no. 11, pp. 992–1003, 2011.

[237] Y. Dong, N. V. Chawla, and A. Swami, "metapath2vec: Scalable representation learning for heterogeneous networks," in *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, 2017, pp. 135–144.

[238] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," *stat*, vol. 1050, p. 20, 2017.

[239] M. Schlichtkrull, T. N. Kipf, P. Bloem, R. v. d. Berg, I. Titov, and M. Welling, "Modeling relational data with graph convolutional networks," in *European semantic web conference.* Springer, 2018, pp. 593–607.

[240] S. Yun, M. Jeong, R. Kim, J. Kang, and H. J. Kim, "Graph transformer networks," *Advances in neural information processing systems*, vol. 32, 2019.

[241] C. Zhang, D. Song, C. Huang, A. Swami, and N. V. Chawla, "Heterogeneous graph neural network," in *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, 2019, pp. 793–803.

[242] D. E. Bloom, E. Cafiero, E. Jané-Llopis, S. Abrahams-Gessel, L. R. Bloom, S. Fathima, A. B. Feigl, T. Gaziano, A. Hamandi, M. Mowafi *et al.*, "The global economic burden of noncommunicable diseases," Program on the Global Demography of Aging, Tech. Rep., 2012.

[243] K. Fricke and S. Vogel, "How interindividual differences shape approach-avoidance behavior: Relating self-report and diagnostic measures of interindividual differences to behavioral measurements of approach and avoidance," *Neuroscience & Biobehavioral Reviews*, vol. 111, pp. 30–56, 2020.

[244] J. T. Nigg and B. Casey, “An integrative theory of attention-deficit/hyperactivity disorder based on the cognitive and affective neurosciences,” *Development and psychopathology*, vol. 17, no. 3, pp. 785–806, 2005.

[245] S. Radke, F. Güths, J. A. André, B. W. Müller, and E. R. de Bruijn, “In action or inaction? social approach–avoidance tendencies in major depression,” *Psychiatry research*, vol. 219, no. 3, pp. 513–517, 2014.

[246] P. P. Martin *et al.*, “Neural systems underlying approach and avoidance in anxiety disorders,” *Dialogues in clinical neuroscience*, 2022.

[247] M. Dempsey, O. Stacy, and B. Moely, ““approach” and “avoidance” coping and ptsd symptoms in innercity youth,” *Current Psychology*, vol. 19, no. 1, pp. 28–45, 2000.

[248] H. C. Breiter, R. L. Gollub, R. M. Weisskoff, D. N. Kennedy, N. Makris, J. D. Berke, J. M. Goodman, H. L. Kantor, D. R. Gastfriend, J. P. Riorden, R. T. Mathew, B. R. Rosen, and S. E. Hyman, “Acute effects of cocaine on human brain activity and emotion,” *Neuron*, vol. 19, pp. 591–611, 1997. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/9331351/

[249] A. Mislove, S. Lehmann, Y.-Y. Ahn, J.-P. Onnela, and J. Rosenquist, “Understanding the demographics of twitter users,” in *Proceedings of the International AAAI Conference on Web and Social Media*, vol. 5, no. 1, 2011, pp. 554–557.

[250] M. Merler, L. Cao, and J. R. Smith, “You are what you tweet… pic! gender prediction based on semantic analysis of social media images,” in *2015 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 2015, pp. 1–6.

[251] P. Eckert, “Gender and sociolinguistic variation 64–75,” *Reading in language and gender. Oxford: Blackwell*, 1997.

[252] J. Holmes, “Women’s talk: the question of sociolinguistic universals.-earlier version of this paper presented to the australian communication association. national conference (1993: Victoria university of technology)-,” *Australian journal of communication*, vol. 20, no. 3, pp. 125–149, 1993.

[253] J. Hu, H.-J. Zeng, H. Li, C. Niu, and Z. Chen, “Demographic prediction based on user’s browsing behavior,” in *Proceedings of the 16th international conference on World Wide Web*, 2007, pp. 151–160.

[254] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[255] K. Weiss, T. M. Khoshgoftaar, and D. Wang, "A survey of transfer learning," *Journal of Big data*, vol. 3, no. 1, pp. 1–40, 2016.

[256] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition.* Ieee, 2009, pp. 248–255.

[257] M. Fey and J. E. Lenssen, "Fast graph representation learning with PyTorch Geometric," in *ICLR Workshop on Representation Learning on Graphs and Manifolds*, 2019.

[258] L. Ilya, H. Frank *et al.*, "Decoupled weight decay regularization," *Proceedings of ICLR*, 2019.

[259] I. Loshchilov and F. Hutter, "Sgdr: Stochastic gradient descent with warm restarts," *arXiv preprint arXiv:1608.03983*, 2016.

[260] C. Zhang, C. C. Dougherty, S. A. Baum, T. White, and A. M. Michael, "Functional connectivity predicts gender: Evidence for gender differences in resting brain connectivity," *Human brain mapping*, vol. 39, no. 4, pp. 1765–1776, 2018.

[261] G. Zhao, G. Hwang, C. J. Cook, F. Liu, M. E. Meyerand, and R. M. Birn, "Deep learning and bayesian deep learning based gender prediction in multi-scale brain functional connectivity," *arXiv preprint arXiv:2005.08431*, 2020.

[262] H. Y. Teke, Ö. Ünlütürk, E. Günaydin, S. Duran, and S. Özsoy, "Determining gender by taking measurements from magnetic resonance images of the patella," *Journal of Forensic and Legal Medicine*, vol. 58, pp. 87–92, 2018.

[263] F. Gantmacher, "The theory of matrices," *New York*, 1964.

[264] P. Lancaster, "M. tismenetsky the theory of matrices," *Computer science and applied mathematics, Academic Press,*, 1985.

[265] K. Lenc and A. Vedaldi, "Understanding image representations by measuring their equivariance and equivalence," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 991–999.

[266] M. D. Zeiler and R. Fergus, “Visualizing and understanding convolutional networks,” in *European conference on computer vision.* Springer, 2014, pp. 818–833.

[267] H. Gholamalinezhad and H. Khosravi, “Pooling methods in deep neural networks, a review,” *arXiv preprint arXiv:2009.07485*, 2020.

APPENDIX A

Supporting Graph Signal Processing Proofs

A.1. LSI Graph Filters Given by Polynomials in Graph Shift

Theorem 1. *Allowing* $\mathbf{A}$ *to be the adjacency matrix of a graph and assuming that its characteristic and minimal polynomials are equal:* $p_{\mathbf{A}}(x) = m_{\mathbf{A}}(x)$*, a graph filter* $\mathbf{H}$ *is LSI iff* $\mathbf{H}$ *is a* ***polynomial*** *in the graph shift* $\mathbf{A}$*, i.e., iff there exists a polynomial*

$$h(x) = h_0 + h_1 x + h_2 x^2 + \cdots + h_L x^L,$$

with complex coefficients $h_i \in \mathbb{C}$*, such that:*

$$\mathbf{H} = (\mathbf{A}) = h_0 \mathbf{I} + h_1 \mathbf{A} + \cdots + h_L \mathbf{A}^L.$$

Proof. Since the shift invariance property in Equation 1.15 holds for all signals $\mathbf{s}$, the matrices $\mathbf{A}$ and $\mathbf{H}$ commute: $\mathbf{AH} = \mathbf{HA}$. Given that $p_{\mathbf{A}}(x) = m_{\mathbf{A}}(x)$, each eigenvalue of $\mathbf{A}$ has a unique eigenvector associated to it [**263, 264**]. Therefore, $\mathbf{H}$ commutes with $\mathbf{A}$ iff it is a polynomial in $\mathbf{A}$ given Proposition 12.4.1 in [**264**]. □

APPENDIX B

Traditional Convolutional Neural Networks (CNNs)

B.1. Localized Convolutional Filter (Kernel)

To understand convolutional neural networks (CNNs), it is easiest to follow using 2D images as an example. A naive approach to applying machine learning (ML) for a learning task on 2D image data would be to flatten the 2D image, treat each individual pixel of the image as an independent feature for the corresponding sample, and train a multilayer perceptron (MLP) on the corresponding task, as shown by Figure B.1. However, a major flaw in this approach is mostly in its scalability. Larger images would require larger MLP layers with more learnable parameters, and if the problem is reduced to working with multiple MLPs for different patches: the spatial correlations provided within the image are lost and left up to the MLPs to pick up on by chance. *Convolution* solves this problem.

The etymology of the word convolution can be traced to the Latin *convolvre*, or “to roll together” [**265, 266**]. As a preface, in computer science, particularly machine learning (ML), convolution in the context of CNNs is often interchangeably used with *cross-correlation*. Given that the weights of a convolutional kernel are randomly initialized and *learned*, flipping *and* shifting the kernel across the input would be redundant; therefore only requiring a shifting kernel. With 2D convolution,

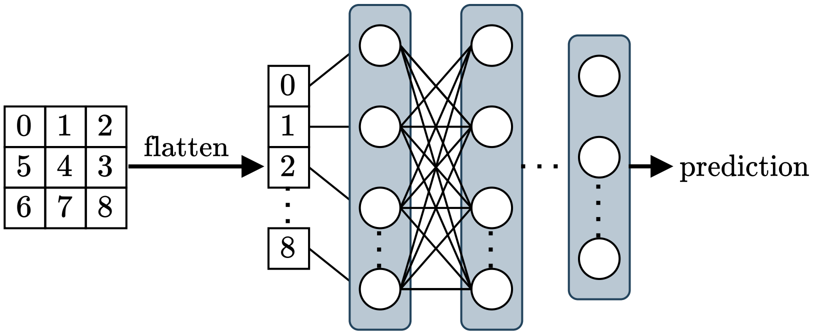
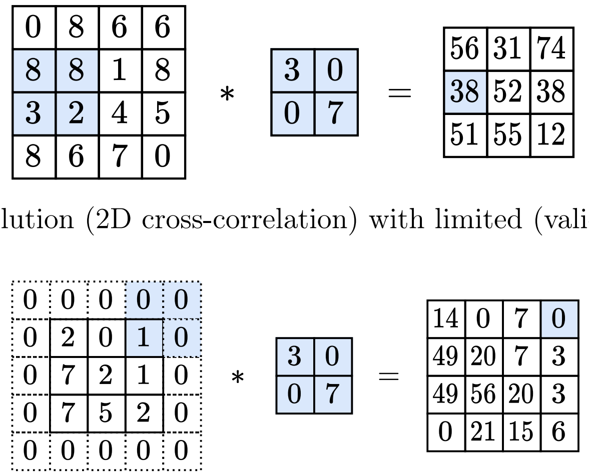


Figure B.1. Flattening 2D image to 1D feature vector and using the flattened sample as input to the MLP for prediction.

we start with a small 2D filter/kernel, or small matrix of weights. This 2D kernel "slides" along the 2D input image data, performing an element-wise multiplication (Hadamard product) of the overlapping matrices and then summing up the elements of the product into a single output pixel for the corresponding output feature map, as depicted in Figure B.2. This process is repeated for every location the kernel slides over on the input image, converting one 2D array of features into another 2D array of features. A convolutional layer of a CNN is made up of multiple convolutional kernels whose weights are randomly initialized (illustrated in Figure B.3) and optimized with gradient-based learning techniques (i.e. gradient descent) that require fully differentiable operations, i.e. convolution.

Convolution still allows us to perform a linear transform, using far less parameters when compared to the naive MLP approach. Rather than "looking at" each individual pixel as a feature requiring more parameters in a MLP, convolutional filters work by getting a "look" at input features that are roughly in the same area on the 2D grid. Since a 2D Euclidean space is assumed, a 2D kernel can exploit this assumption



$$\begin{bmatrix} 0 & 8 & 6 & 6 \\ 8 & 8 & 1 & 8 \\ 3 & 2 & 4 & 5 \\ 8 & 6 & 7 & 0 \end{bmatrix} * \begin{bmatrix} 3 & 0 \\ 0 & 7 \end{bmatrix} = \begin{bmatrix} 56 & 31 & 74 \\ 38 & 52 & 38 \\ 51 & 55 & 12 \end{bmatrix}$$

(a) 2D convolution (2D cross-correlation) with limited (valid-)“padding”.

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 1 & 0 \\ 0 & 7 & 2 & 1 & 0 \\ 0 & 7 & 5 & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} * \begin{bmatrix} 3 & 0 \\ 0 & 7 \end{bmatrix} = \begin{bmatrix} 14 & 0 & 7 & 0 \\ 49 & 20 & 7 & 3 \\ 49 & 56 & 20 & 3 \\ 0 & 21 & 15 & 6 \end{bmatrix}$$

(b) 2D convolution (full 2D cross-correlation) with a zero-padding.

Figure B.2. 2D convolution with different forms of padding. In the case of valid-padding, output values are only considered where zero-padding is not required for complete overlay of the 2D kernel.

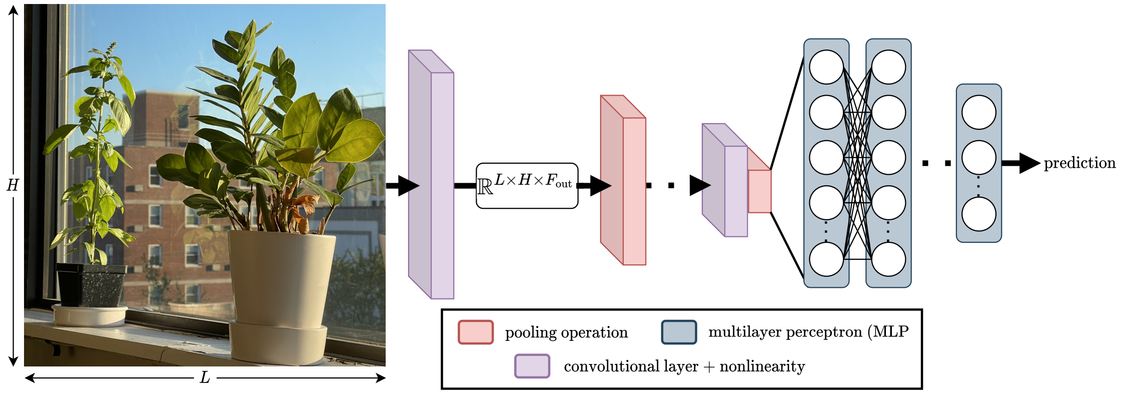

H
L
$\mathbb{R}^{L\times H\times F_{\text{out}}}$
prediction
pooling operation
multilayer perceptron (MLP
convolutional layer + nonlinearity

Figure B.3. Typical CNN architecture using an alternating sequence of convolutional layers followed by element-wise nonlinear activation function and pooling operations. A MLP can follow after to tie together latent features for varying prediction tasks.

and use convolution to its advantage by translating a parameterized 2D kernel, as illustrated in Figure B.2, to efficiently learn on spatial correlations in the data (i.e., NN trained for vertical edge detection).

Padding is one of the most popular tools for controlling the size of the output for a convolution operation, and therefore a convolutional layer. Generally, a filter's output dimensionality is determined by the size of the input and the size of the kernel. Signal padding in its one of many forms, as illustrated in Figure B.2, provides a quick solution to resolving boundary issues that come up as a result of convolution (i.e., zero-padding a 3×3 image before convolving with a 2×2 kernel to preserve a 3×3 boundary for the corresponding output feature map).

Under the assumption of some form of padding, and a given input, $\mathbf{X} \in \mathbb{R}^{L \times H \times F_{\text{in}}}$ ($L \times H$ image with F_{in} channels/features per pixel), a convolutional layer is defined as a set of F_{out} convolutional filters which are optimized to map $\mathbf{X} \in \mathbb{R}^{L \times H \times F_{\text{in}}} \mapsto \mathbf{Y} \in \mathbb{R}^{L \times H \times F_{\text{out}}}$ for a given task by computing the corresponding output feature map $\mathbf{Y} \in \mathbb{R}^{L \times H \times F_{\text{out}}}$. As a generalization, regardless of the n-dimensional Euclidean space, CNNs can be defined as linear mapping tools that learn to map an input feature map $\mathbf{X} \in \mathbb{R}^{\cdots \times F_{\text{in}}}$ to an output feature map $\mathbf{Y} \in \mathbb{R}^{\cdots \times F_{\text{out}}}$ by using a smaller filter in the same n-dimensional space and exploiting the spatial correlations in the data that exists because of lattice-structure the elements are laid out.

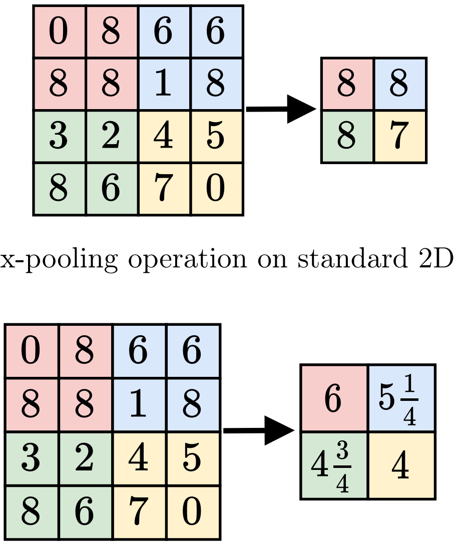
B.2. Pooling

Pooling layers are designed to reduce the dimensions of input data in order to reduce the computational power required to process data in a CNN. Pooling within a NN is useful for extracting dominant features which are rotational- and spatial-invariant.

Several forms of pooling exist in the literature [**267**], however for simplicity: pooling is explained via *max*- and *average*-pooling, which are two of the most common techniques. Max-pooling is best explained in synonymy to cross-correlation with valid-padding by applying a kernel, of an arbitrary predetermined size, and returning the maximum value in the overlapping portion at valid locations, as illustrated by Figure B.4a.

In a crude sense, max-pooling acts as an indirect noise suppressant after a non-linear activation function is applied (Figure B.3), discarding noisy activations and returning the dominant activation within a patch of the input data at each valid location. *Average*-pooling applies a similar idea, by returning the mean of the values within an overlapping patch, instead of the dominant value (see Figure B.4b).

Each form of pooling comes with its own set of trade-offs which may vary by application. In general however, pooling provides an efficient method to reduce the computational complexity required to train large NNs and view input feature maps at multiple multiple resolutions.



0	8	6	6
8	8	1	8
3	2	4	5
8	6	7	0

8	8
8	7

(a) Max-pooling operation on standard 2D image.

0	8	6	6
8	8	1	8
3	2	4	5
8	6	7	0

6	$5\frac{1}{4}$
$4\frac{3}{4}$	4

(b) Average-pooling operation on standard 2D image.

Figure B.4. Max- (top) and average-pooling (bottom) examples using a 3 × 3 input image and a 2 × 2 pooling kernel.