
# Learning Interpretable Prototype Trajectories for Patients with Alzheimer's Disease

Master's Thesis in Computer Science

Sarah Moy de Vitry

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CHALMERS UNIVERSITY OF TECHNOLOGY
GOTHENBURG UNIVERSITY
Gothenburg, Sweden 2021
www.chalmers.se

Master's thesis 2021

# Learning Interpretable Prototype Trajectories for Patients with Alzheimer's Disease

Sarah Moy de Vitry

UNIVERSITY OF GOTHENBURG

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Computer Science and Engineering
Chalmers University of Technology
University of Gothenburg
Gothenburg, Sweden 2021

Learning Interpretable Prototype Trajectories for Patients with Alzheimer's Disease
Sarah Moy de Vitry



Supervisors: Fredrik Johansson (Chalmers), Gideon Dresdner (ETH Zurich)
Examiner: Peter Damaschke, Data Science and AI Division

Master's Thesis 2021
Department of Computer Science and Technology
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000
Eidgenössische Technische Hochschule Zürich
Rämistrasse 101, 8092 Zürich

Typeset in LaTeX
Printed by Chalmers Reproservice
Gothenburg, Sweden 2021

Learning Interpretable Prototype Trajectories
for Patients with Alzheimer's Disease
Department of Computer Science and Technology
Sarah Moy de Vitry
Chalmers University of Technology
Gothenburg University
ETH Zurich

# Abstract

Alzheimer's Disease (AD) patients are known to have subtypes with distinctive progression traits. But progressions differ with distinctive patterns emerging even before patients transition to AD. Understanding these differences could help physicians understand what normal vs abnormal progressions look like. One way of characterizing differences in trends is by identifying patients that represent prototypical progressions of the defining features. We trained models to learn low-dimensional representations of all patient sequences and then, among the patients who transition to AD, used unsupervised learning to identify prototypical sequences that resembled and represented subsets of the representations. We examined trends in these subsets, searching for distinctive feature progression, and compared the prototypes to the subsets they described. We found that there are unique trends for subgroups of patients. In particular, we found a cohort that was predominantly female (79%) with distinctly high verbal memory retention relative to other cohorts. This was significant because verbal memory is not a significant predictive factor, but seemed to be an axis of variation when distinguishing among subgroups. Additionally, we looked at how prototypes learned during training improve model performance compared to prototypes selected after training.

Keywords: Alzheimer's Disease, prototypes, interpretability, deep clustering, machine learning

# Acknowledgements

My deepest thanks go to my supervisor at Chalmers, Fredrik Johansson, for his constant practical and moral support throughout this project. I would also like to thank my supervisors at ETH Zurich, Gideon Dresdner and Gunnar Raetsch, for their interest and support of the project, even from afar. Finally, I thank my mother, Joan de Vitry Moy, for her help patiently reading and editing this thesis report.

Sarah Moy de Vitry, Gothenburg, June 2021

# Contents

**List of Figures** **xi**

**List of Tables** **xiii**

**1 Introduction** **1**
1.1 Motivation . . . 1
1.2 Goals . . . 2
1.3 Related Work . . . 3
1.4 Limitations . . . 4
1.5 Ethics . . . 4

**2 Background** **7**
2.1 Alzheimer's Disease . . . 7
2.2 Clustering . . . 8
2.3 Sequence Learning . . . 8
2.4 Prototype Learning . . . 9

**3 Data** **11**
3.1 Alzheimer's Disease Neuroimaging Initiative (ADNI) . . . 11
3.2 Missingness . . . 11
3.3 Feature Selection . . . 14
3.3.1 Cognitive Scores . . . 14
3.3.2 Biomarkers . . . 14

**4 Methods** **17**
4.1 Dimensionality Reduction . . . 17
4.1.1 Classifier . . . 18
4.1.2 Sequence-to-Sequence Autoencoder . . . 19
4.2 Prototype Learning through Deep Clustering . . . 20
4.3 Joint Dimensionality Reduction and Prototype Learning . . . 21
4.4 Preprocessing . . . 23
4.5 Sample Splitting . . . 23
4.6 Hyperparameter ranges . . . 23

**5 Results** **25**
5.1 Model Performance . . . 25
5.2 Cluster Inspection . . . 27

5.3 Prototypes . . . 33
5.3.1 Clustering Prototypes . . . 33
5.3.2 Jointly Learned Prototypes . . . 35

**6 Discussion** **37**
6.1 Dimensionality Reduction . . . 37
6.2 Cluster Analysis . . . 38
6.3 Prototype Analysis . . . 39
6.3.1 Cluster Prototypes . . . 39
6.3.2 Jointly Learned Prototypes . . . 39
6.4 Future Work . . . 40
6.4.1 Quantitative Prototype Analysis . . . 40
6.4.2 Qualitative Prototype Analysis . . . 40
6.4.3 Feature Selection . . . 41

**7 Conclusion** **43**

**Bibliography** **45**

# List of Figures


3.1 A graph of the number of patients that registered a visit with respect to the number of months after the baseline measurement (month 0). This graph does not indicate how many measurements are taken at followup visit. . . . . 13

4.1 A high-level diagram of the classifier model. The hidden state $h_t$ is the embedding for sequence $x_0 - x_t$ . . . . . 18
4.2 A detailed diagram of the encoder. See Equations 4.1-4.4. . . . . 18
4.3 A high-level diagram of the sequence-to-sequence autoencoder model. The data for the entire sequence is encoded and the decoder predicts back the input sequence in reverse order. . . . . 20
4.4 A high-level diagram of the deep clustering model architecture for the classifier encoder. The encoder is pretrained with diagnosis and continuous features as target values. K-means clustering is performed on the extracted embeddings and the cluster centroids are taken as prototypes. The similarity to the prototypes (Equation 4.15) are used to predict target value $\hat{x}_{t+1}$ . . . . . 21
4.5 Autoencoder model with joint dimensionality reduction and prototype learning. The model has an architecture similar to a simple autoencoder, however, instead of taking the last embedding from the encoder as input, the decoder takes a vector of similarities between the embedding and the prototypes. The prototypes are learned in the same latent space as the embeddings and are updated during back propagation. . . . . 22

5.1 Loss over epochs for the two dimensionality reduction models. Loss for the classifier was measured by the cross entropy (CE) loss and for the autoencoder by the cross entropy and mean absolute error (MAE) score . . . . . 25
5.2 Balanced accuracy score for the classifier model trained on cluster centroids as prototypes with respect to the number of prototypes used. 26
5.3 Average loss for all features predicted by the autoencoder trained on similarity to prototypes with respect to the number of prototypes used. The loss is calculated using cross entropy on discrete variables and mean squared error on continuous variables. The dashed line indicates the score for the baseline autoencoder (without prototypes). 27

5.4 Visualization of the two principal components for all datapoints encoded by the **classifier**. The color indicates the change in diagnosis at time $t$ from time $t-1$. A label of CN, MCI, or AD indicate no change has occurred since the previous measurement. . . . . 28
5.5 Visualization of the two principal components for all datapoints encoded by the **autoencoder**. The color indicates the change in diagnosis at time $t$ from time $t-1$. A label of CN, MCI, or AD indicate no change has occurred since the previous measurement. . . . . 29
5.6 Visualization of the two principal components of latent spaces for patients who have transitioned from MCI to AD. The colors indicate cluster assignments and the larger circles indicate cluster centroids. . 31
5.7 Visualization of the two principal components for all datapoints in the latent space learned by the **prototype based model** with 9 prototypes. The figures represent the embeddings learned for two different folds of the data. The color indicates the change in diagnosis at time $t$ from time $t-1$. . . . . 36

# List of Tables

3.1 This table shows the mean and availability of features for all time points where they are registered in ADNI. Certain features, such as beta-amyloid and phosphorylated tau, are measured at a lower occurance than features such as ADAS scores. This imbalance comes from varying collection sites and from relative difficulties of biospecimen collection [27]. . . . . 12

5.1 Gender and age distributions for the clusters in the two latent spaces. 30
5.2 Average ADAS13 scores for clusters at transition time $t$, change with respect to that measure at 12 months before and 12 months after recorded transition time $t$. Higher scores indicate worse cognitive function. . . . . 32
5.3 Average RAVLT learning score for clusters 12 months before, at, and 12 months after recorded transition time $t$. Lower scores indicate a worse cognitive performance. . . . . 32
5.4 Average whole brain size for clusters 12 months before, at, and 12 months after recorded transition time $t$ (scaled by e-06). Smaller brain sizes are associated with worse cognitive decay in Alzheimer's patients. . . . . 32
5.5 Average tau for clusters 12 months before, at, and 12 months after recorded transition time $t$. Higher levels of tau are correlated with more severe cases of AD. . . . . 33
5.6 Average A-beta for clusters 12 months before, at, and 12 months after recorded transition time $t$. . . . . 33

# 1

# Introduction

Alzheimer's Disease (AD) is a form of dementia that affects more than 30 million people, a number that is expected to triple by 2050 [15]. It is characterized by symptoms of cognitive decline so severe that patients forget their names, their loved ones, and how to perform simple day-to-day tasks such as dressing themselves or brushing their teeth. Although there is no cure for the disease, early diagnosis, preferably before the onset of clinical symptoms, can help mitigate the negative effects through psychological preparation for both the patients and their caregivers [28].

Given the high stakes of being able to anticipate an AD diagnosis, predicting which patients are at risk of being diagnosed has been at the forefront of research on AD. Models harnessing the power of recurrent neural networks or long short-term memory modules have already been successful at increasing prediction accuracy [27]. However, in this thesis we focused on understanding not which patients will progress, but how patients progress from mild cognitive impairment to AD. We sought to group patients who transition based on their similarities with respect to important features and characterized these groups by a single prototypical trajectory. Similar to the way physicians would classify an abnormally high or low level of some biomarker with respect to patients they have seen in the past, we identified salient features in trajectories that would distinguish patients from each other.

This thesis could be of interest to researchers studying prototype-based sequence learning as an example of a new area to which the method has been applied. Furthermore, while we use a single, limited dataset to train our model, the conclusions could be useful to medical researchers studying subtypes of AD.

## 1.1 Motivation

With the rise of digitized medicine, we have access to data that is collected over time. Using this data, we have the potential to analyse a patient's state not only at a single examination visit, but over many visits, and using deep learning techniques, we have the capacity to compress and evaluate this volume of data. This amalgamation of data and computing power has the potential to reveal underlying patterns in data that were not previously accessible. Identifying such patterns in key features could

reveal differences among patients that divide them into distinct cohorts.

Cohort identification could lead to an effective way of differentiating subgroups of patients based on their disease phenotype. Further describing the sequential characteristics of patient trajectories tractably and presenting interpretable results could provide new clinical insights. This is true especially for diseases that have a large variance in characteristics over time but where patients converge to a similar final disease state. Important examples are chronic diseases such as rheumatoid arthritis or Alzheimer's Disease, where patients suffer for up to half of their life with varying treatments and symptoms along the way [9, 1**?** ].

Treating and using voluminous data requires more consideration than naive application of traditional machine learning methods. For example, an intuitive solution to the problem of grouping data points would be clustering. Clustering refers to unsupervised machine learning methods that assign group labels to data points based on their proximity to one another. However, for high-dimensional data this is problematic: how do we define proximity for sequences with many different features, over time, and with potentially different lengths ?

The first task is therefore to identify only the salient features and progressions. Since the capacity to store almost infinite data encourages collection of features even if they are redundant, models risk identifying only superficial patterns. Additionally, the sequential order of digital patient records contains valuable information, but to capture that information, models must be adapted to consider the order as a feature. The second task is to group patients based on differences in the features identified as relevant.

Altogether, the problem reduces to two parts: learning low-dimensional representations of data such that only important features are captured and selecting subsets of the representations that are characterized by differing progressions.

## 1.2 Goals

The goal of this project was to formulate the distinct patterns of variability in the context of Alzheimer's disease with a specific focus on interpretability. We aimed to develop and evaluate a machine learning method which identifies and describes prototypical clinical progressions leading up to a given observed event. In particular, we wanted to identify the different patient trajectories leading up to the transition from Mild Cognitive Impairment (MCI) to Alzheimer's Disease (AD). In order to reach that goal, we had to answer a series of questions that became our intermediate goals.

- *How should feature progression be captured ? What features are relevant for distinguishing among subgroups ?*
- *Are there identifiable and distinctive traits for the subgroups of patients ? How can subgroup quality be measured ?*

- *How can prototypes be forced to represent salient features ? What is the best way to present a prototype such that it is useful and understandable ?*
- *How can prototype quality be measured ?*

## 1.3 Related Work

Alzheimer's Disease is the most prevalent form of dementia and early detection can improve quality of life for patients, making its prediction an important area of research [24]. Early detection is critical enough that it is one of the main goals of the Alzheimer's Disease Neuroimaging Initiative (ADNI) study, a longitudinal study that has collected a database of clinical, imaging, genetic, and biochemical biomarkers from over 2000 patients since 2004 [1]. The ADNI database has been used in over 1800 studies and is an important resource both in time-series based machine learning and for healthcare in general [24]. One example of how ADNI data has been used for disease progression prediction is the The Alzheimer's Disease Prediction Of Longitudinal Evolution (TADPOLE) Challenge [19], where participants were asked to predict the progression of three biomarkers for up to 5 years into the future. One example of a TADPOLE Challenge submission is work by Nguyen et al. [27] that uses a simple recurrent neural network (RNN) layer for prediction.

While prediction accuracy is important, adding an extra constraint of model interpretability is critical for high-stakes decisions such as those related to healthcare [32]. Interpretability can be added to a model by adding constraints based on domain knowledge and is inherent to algorithms such as linear regression and decision trees. In the context of data from ADNI, one challenge was selecting an interpretable model that would integrate the temporal aspect of the data. One method for obtaining interpretable predictions for time series data is through deep clustering: the categorization of data points into clusters based on their similarity in a low-dimensional latent space [21]. The goal is that the latent space captures the most important features of the data and that performing clustering in this space highlights groups of patients with relevant similarities. Then, given a history shared by other patients, a single patient is predicted to have outcomes similar to those of similar patients. Lee et al. [16] proposed a method for performing supervised clustering in a low-dimensional space by optimizing the loss of prediction based on both the latent representation and the cluster assignment. Their main goal was to identify clusters that shared future outcomes and to then assign new patients to clusters based on similar past trajectories. Another method for interpretable prediction is to compare a patient trajectory to a limited set of exemplar trajectories (prototypes) also in a latent space as done by Ming et al. [22].

Prediction is necessary for healthcare because it helps identify patients who are more at risk of progressing to AD. However, it is also relevant to examine patients retrospectively to understand how their progressions differ and to effectively identify phenotypical subtypes (cohorts of patients who share feature progression such as unusually rapid decline in cognitive capacity). Like prediction, one way of identifying

patient subtypes is to perform clustering in a latent space. An example of this is shown by Baytas et al. [4] who train an auto-encoder to learn a low-dimensional representation of the data and then perform k-means clustering in the latent space. While the representation they learn is optimal for reconstruction, it is not necessarily optimal for clustering, which is an issue we try to address in our model.

While the work done in this thesis is similar to the previous methods in some aspects, we focus specifically on performing unsupervised clustering that is optimized for reconstruction loss. We are interested in examining the prototypes learned by correctly predicting sequences, and only use the predicted sequences as a measure of how viable the model is. This project is perhaps most similar to the work done by Lee et al. [22] applied to the ADNI dataset and without the focus on prediction.

## 1.4 Limitations

The main limitation of this work is the single dataset we use. While ADNI is the most extensive Alzheimer's Disease studies publicly available, it is limited to a little over 2000 patients who are geographically restricted to the US and Canada. Without data from varied populations to compare it with, we are constrained to this single cohort. Further, we have no way of telling whether the subset of patients who enroll in the study are representative of the entire population of AD patients. Certain measurements are also missing at regular intervals and could lead to biased results for points where the missing data has been imputed.

In future work, this could be counteracted by including data from other AD datasets such as the The Australian Imaging, Biomarker and Lifestyle Flagship Study of Ageing (AIBL) [33].

## 1.5 Ethics

One aim of this thesis is to identify a subset of patients that describes typical progression towards a common outcome. Moreover, we want to identify certain features which distinguish the trajectories from themselves. For example, if all prototype patients share similar rates of cognitive degradation, cognitive degradation will not be highlighted as a defining feature. Studies suggest that there are subtypes of Alzheimer's disease [25] and finding key features could be important in identifying these subtypes. Ideally this would allow doctors to provide treatment more targeted to the Alzheimer's subtype exhibited.

One asterisk to keep in mind, however, is that our model is not intended for use as a predictive tool. The intent is to inform doctors that a patient's trajectory most resembles this subset of other patients, with this similarity. We do not intend the prototypes to be used as indicators of future progression; this question has already been studied in [22].

From the perspective of regulations on data handling, we worked with de-identified

subjects who consented to their data being used for the advancement of Alzheimer's research. We have not attempted to re-identify any subject in the context of our research, in accordance with Human Subject Protection rules [5]. Moreover we will not share the raw data we use, further ensuring the protection of all study participants.

# 2

# Background

We were interested in detecting subtypes of patients who transition from Mild Cognitive Impairment (MCI) to Alzheimer's Disease (AD) based on differences in their trajectories. While this problem seems clear, it needs to be reduced to concise definitions that allow a mathematical representation. Patient trajectories can be defined as sequences where every set of measurements at a given time is represented as a vector of values, and where a set of these value vectors is the sequence. Subtypes of AD can be thought of as clusters of patients who, by some distance metric, are more similar among themselves than to other patients. In this section, we review some of the background knowledge that will be useful in defining this problem mathematically.

## 2.1 Alzheimer's Disease

Dementia is a blanket term which encompasses multiple disorders associated with aging, ranging from disorientation to memory loss. Approximately 50 million people (5-8% of the global human population over 60 years of age) suffer from some form of dementia. This number is expected to triple to 150 million people by 2050 due to the fast-growing elderly population [30]. Although there are different forms of dementia, AD is the most prevalent (estimated at upwards of 60% of cases) [11].

Alzheimer's disease is defined by the death of nerve cells in specific areas of the brain. In particular, it is identified by shrinkage of the temporal lobe and hippocampus, both areas of the brain responsible for storing and retrieving new information [29]. Because of this significant impact on memory and because early detection can help manage the disease, studies of biomarkers, cognitive indicators, and clinical function measurements have been done in an attempt to understand their role in AD progression [1].

While there is currently no cure for AD, early detection of the disease can help delay its progression and plan symptom management [15]. Ideally, an AD diagnosis comes before the onset of severe cognitive symptoms. This has made prediction based on physical indicators such as plasma biomarkers and brain structure an important area of research [13].

A widespread model of AD is that it is a family of multiple diseases subtypes, rather than a single disease [25]. These subtypes have been characterized by features such as age at onset, spatial expansion of diseased regions of the brain and rapidity of cognitive decline [36]. In the interest of going beyond prediction and understanding the pathology of AD, examining this type of variation among patients is also important. Identifying prototypical trajectories towards AD can help characterize such subtypes [16].

## 2.2 Clustering

Subtypes of patients can be understood as groups of patients who are more similar among themselves than they are to patients from other subgroups, while still sharing at least one common trait, the same basic medical diagnosis, with other subgroups. Clustering is an unsupervised machine learning technique used to group datapoints by their similarities. The goal is to identify subgroups of points that share key features [2]. Clustering is used for statistical analysis and as a preprocessing technique to help discover subpopulations in data.

One of the most common clustering algorithms is k-means, where each cluster is represented by a single mean vector or centroid. The initial centroids are chosen from the data points uniformly at random. Each datapoint is then assigned the class of its nearest centroid. The clusters are updated such as to minimize the variance among cluster points. While k-means is quick to converge to a local optimum, it is very sensitive to initialization and generally performs better when used with initialization techniques such as k-means++. While k-means is an effective technique for grouping datapoints, it cannot be naively scaled to high-dimensional data. However, in combination with other techniques such as dimensionality reduction, k-means can be used with high-dimensional, high-noise data [38].

One form of such clustering is deep clustering, where k-means is run on data that has been encoded to a latent space. The idea is that the latent space represents only the most important features of the data and that therefore, the clusters will identify subgroups based on these features. This is useful when clustering on the raw data is intractable, such as with high-dimensional data.

## 2.3 Sequence Learning

In machine learning, sequences refer to variable-length data that have a sequential order and where that order contains information about the data. This sequential dimension can be logical, for example with language data, or it can be temporal, as for time series data [7]. Sequence learning refers to learning underlying patterns and characteristics of sequential data.

The challenge of sequence learning is representing relevant information from past data along with information from new data: how much importance should be given

to past measurements at the present time ? Moreover, how can patterns be recognized over time ? Another challenge of sequence learning is the variable input length. Classic neural networks used for pattern detection are designed to take fixed length vector inputs and to output fixed length vectors. The fact that this does not work for sequences is particularly visible for sequence learning in language: sentences do not have a fixed length, nor should they be forced to. However this variability requires a way standardizing the data.

A solution to these problems is reducing the dimensionality of sequences by using models to learn salient features and compressing the data to be represented as a single vector. Sequence-to-sequence autoencoders are models that learn low-dimensional representations of sequences by optimizing for reconstruction loss during training. The learned latent space can then be used as input for models that require fixed-length input [6].

The two main components of a sequence-to-sequence autoencoder are the encoder and the decoder. The encoder is a recurrent neural network that takes as input raw data from timepoint $t$ and the embedded vector from timepoint $t-1$, and outputs the embedded vector for timepoint $t$. This new embedded vector $h_t$ represents all the data embedded up to time $t$. The decoder takes $h_t$ as input and outputs an estimate $\hat{x}_t$ and the next hidden state $\hat{h}_{t-1}$. The decoder thus predicts sequentially backwards from time $t$ (its last prediction will be of time 0) [34].

## 2.4 Prototype Learning

Prototypes are single datapoints that represent subgroups of data [23]. They are selected based on how well they summarize their subgroup. For example, the prototypes for a group of points could be the cluster centroids. Cluster centroids could be a good way of summarizing subgroups because all points in that group are close to the centroid. Prototypes can be used for data inspection, but are mainly used to create interpretable machine learning models [17].

One method for learning prototypes in the context of deep learning, and the one closest to what we implement in this project, integrates prototype selection into the model training process. The architecture consists of four key elements: an autoencoder which embeds data in a lower-dimensional space, a prototype layer which learns a set of prototypes, or points in the latent space that are similar to embedded real datapoints, a linear layer which performs classification based on similarity to prototypes, and a projection step, by which the prototypes in the latent space are reflected back to raw data [22]. The prototypes are forced to be similar to data in the latent space by regularization terms. For further interpretability, the learned prototypes are projected onto the closest real datapoint in the latent space every couple of epochs.

Prototype learning is a response to critique of the lack of interpretability in the domain of deep neural networks [23]. Prediction based on prototypes is considered

interpretable because the prototypes can be inspected by decoding to the original n-dimensional space. Since the prediction is based on similarity to this set of prototypes, the motivation for decisions is reflected by specific traits of the prototypes [22]. Moreover, because the set of prototypes is learned during training, they are the explanation for what the model has learned.

# 3

# Data

## 3.1 Alzheimer's Disease Neuroimaging Initiative (ADNI)

This project usesd data from ADNI [1], a study that collects data with the goal of encouraging the study of Alzheimer's Disease to help discover treatments and prevention strategies, or to identify patients at risk of progressing to the disease.

ADNI is ongoing since 2004 and collects data from 63 sites in the US and Canada. It contains measurements from 2155 patients and collects biospecimens for features that have an observed correlation with AD progression. The population of participants are individuals between the ages of 55 and 90 years old. Individuals are either classified as cognitively normal (CN) or diagnosed with mild cognitive impairment (MCI) or Alzheimer's Disease (AD). CN individuals are considered the baseline cohort.

A series of baseline measurements are taken from the participants upon entering the study and are tested at followup sessions with variable frequencies of 3, 6, 12, or 24 months. The measurements consist of cognitive tests, biomarker samples, and MRI or PET scans, depending on which phase of the study ADNI was in at the time (ADNI has gone through four phases since it was initiated, and certain biomarkers are only measured in more recent phases) [1].

## 3.2 Missingness

In data science, missingness is used to indicate the absence of parts of the data that was expected to be observed [8]. In general it is signaled by a missingness indicator such "NaN" or a negative value for a feature that can only be positive. Missingness can be the complete absence of a data point or the absence of a feature value within a data point. In the context of sequence learning, for example, data point absence could be a missing data point at time t where other sequences do have data.

In health care, missingness can have several causes. The absence of an entire data point can be caused by patients not showing up to a given appointment and therefore

| | Mean (± std) | % timepoints with meas. |
|---|---|---|
| Clinical Dementia Rating (SB) | $2.17 \pm 2.81 \times 10^0$ | 70.36% |
| ADAS-Cog11 | $1.13 \pm 0.86 \times 10^1$ | 69.95% |
| ADAS-Cog13 | $1.75 \pm 1.16 \times 101$ | 69.27% |
| Mini-Mental State Exam (MMSE) | $2.65 \pm 0.39 \times 10^1$ | 70.12% |
| RAVLT immediate | $3.44 \pm 1.36 \times 10^1$ | 69.33% |
| RAVLT learning | $4.02 \pm 2.81 \times 10^0$ | 69.33% |
| RAVLT forgetting | $4.23 \pm 2.52 \times 10^0$ | 69.12% |
| RAVLT forgetting percent | $5.97 \pm 3.83 \times 10^1$ | 68.57% |
| Functional Act. Quest.(FAQ) | $5.59 \pm 7.92 \times 10^0$ | 70.60% |
| Montreal Cog. Assess. (MOCA) | $2.30 \pm 0.47 \times 10^1$ | 38.99% |
| Ventricles | $4.21 \pm 2.32 \times 10^4$ | 58.44% |
| Hippocampus | $6.68 \pm 1.24 \times 10^3$ | 53.39% |
| Whole brain volume | $1.01 \pm 0.11 \times 10^6$ | 60.35% |
| Entorhinal cortical vol | $3.44 \pm 0.81 \times 10^3$ | 50.78% |
| Fusiform cortical vol | $1.71 \pm 0.28 \times 10^4$ | 50.78% |
| Middle temporal cortical vol | $1.92 \pm 0.31 \times 10^4$ | 50.78% |
| Intracranial volume | $1.53 \pm 0.16 \times 10^6$ | 62.43% |
| Florbetapir (18F-AV-45) - PET | $1.19 \pm 0.22 \times 10^0$ | 16.62% |
| Fluorodeoxyglucose (FDG) - PET | $1.20 \pm 0.16 \times 10^0$ | 26.31% |
| Beta-amyloid (CSF) | $1.02 \pm 0.59 \times 10^3$ | 18.60% |
| Total tau | $2.93 \pm 1.30 \times 10^2$ | 18.55 % |
| Phosphorylated tau | $4.80 \pm 1.44 \times 10^1$ | 18.62% |
| Diagnosis | | 69.89% |

**Table 3.1:** This table shows the mean and availability of features for all time points where they are registered in ADNI. Certain features, such as beta-amyloid and phosphorylated tau, are measured at a lower occurance than features such as ADAS scores. This imbalance comes from varying collection sites and from relative difficulties of biospecimen collection [27].

not getting data collected at that time. For example, Figure 3.1 shows that compared to the number of patients who show up at the baseline visit at 0 months, very few patients register a visit for the 3 month followup session. Feature value absence can be caused by inaccessability of certain measurement tools or by the difficulty of sampling. In ADNI in particular, while a set of features is defined for what should be sampled, there are many different collection sites, all with varying capacities for feature collection. Further, ADNI's longitudinal nature results in participants dropping out over time creating a nonuniform sequence length.

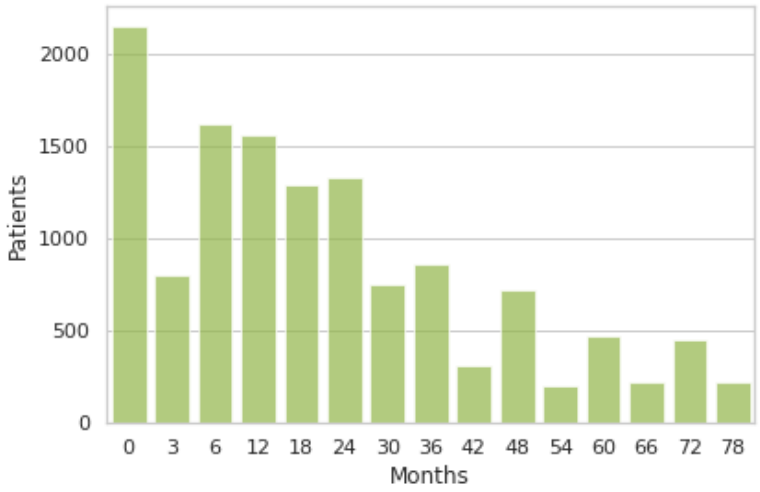



**Figure 3.1:** A graph of the number of patients that registered a visit with respect to the number of months after the baseline measurement (month 0). This graph does not indicate how many measurements are taken at followup visit.

Missingness is problematic in sequence learning because most models are not well designed to handle input with varying time steps between data points [10]. So how do we deal with sequences where data points at time t are present in some sequences but not others? One option would be to drop all sequences that are missing data at time point $t$. This is possible either if the number of sequences with missing data is negligible or if the dataset is large enough that removing those sequences does not affect the data distributions. ADNI, while a large study in the domain of healthcare, is not objectively big and contains a high proportion of missingness. Consequently, the missing datapoints could not be dropped as this would cause a huge loss of information.

Another solution is to perform imputation for the missing values [18]. There are a number of ways data can be imputed. Forward filling, for example, imputes the data by copying values from the previous time point. Linear filling estimates the data by inserting values calculated from previous and following time points. Values that have been imputed should be marked as such and excluded from model performance estimates such as loss functions since they could bias the optimization.

## 3.3 Feature Selection

In order to limit complexity and to achieve better interpretability, we chose to use only a small subset of the biomarkers available in ADNI. We did not use the genetic data nor the MRI and PET imaging data. Since we were interested in selecting features that had known association with AD progression, we used the same subset of features as those used in the TADPOLE challenge [19, 27]. Some of the features most correlated to AD diagnosis are ADAS-Cog, FDG, amyloid-beta, and tau.

### 3.3.1 Cognitive Scores

The cognitive function of patients was measured by a variety of cognitive tests.

Mini-Mental State Examination (MMSE) and Alzheimer's Disease Assessment Scale-Cognitive Subscale (ADAS-Cog) are two cognitive tests used to measure mental decline for AD patients. MMSE is a widely-used assessment of cognitive function for elderly patients. It consists of 30 questions that relate to orientation, memory, and recall and is scored on a scale between 0 and 30 where a lower score indicates worse cognitive function.

ADAS-Cog is considered a more precise indicator of cognitive function for MCI patients than MMSE [3]. ADAS-Cog includes tasks such as recalling words, following commands, and temporal orientation, and measures general comprehension [14]. ADAS-Cog scores are given on a scale between 0 and 85 where a higher score indicates worse cognitive impairment. It is used in pharmaceutical trials as an indication of AD progression but it is still imprecise for patients with MCI [14].

Multiple cognitive tests were included in the input, in spite of redundancy, because the trade off between quality and coverage is unclear. For example, the Montreal Cognitive Assessment (MOCA) score is considered more indicative of MCI and AD compared to the MMSE score [31]. However, the MMSE score is available for 70.12% of measured time points in ADNI, while the MOCA score is only available for 38.99% of measured time points.

### 3.3.2 Biomarkers

**A-beta** The beta-amyloid peptide is a component of the amyloid plaques present in the brains of AD patients. In cognitively normal individuals, the peptide is evacuated from the brain during sleep, but its build up is a strong indicator of AD and while its pathology is not exactly defined, it is attributed to play a significant role in AD disease progression [37]. A-beta peptides are deposited in the brain early in the disease, long before the onset of clinical symptoms and do not seem to evolve much during the later stages of AD.

**Tau** The tau protein is associated with insoluble filaments that form in the brain prior to manifestation of AD symptoms. Tau is present in healthy individuals and

helps maintain stability of microtubules. However, in patients with AD, the production of tau is deregulated, causing the microtubules to disassemble [20].

# 4

# Methods

Our main goal was to identify interpretable trajectories that could distinguish differences among patients who transition to AD and thereby recognize patterns that align with humanly-understandable knowledge. This is a task that is challenging to quantify since interpretability is inherently subjective. However, it can be broken down into two elements: isolating features that contribute to a meaningful distinction and identifying patients who differ with respect to these features.

The same way physicians would recognize a high or low level of some biomarker as an indicator of normal or abnormal progression based on patients they have seen, the model needed to be able to identify salient features in trajectories that would distinguish them from each other. A model built to detect patterns on the raw data would be susceptible to noise and be more likely to identify superficial patterns that do not capture underlying subtleties in the data. One solution to this is dimensionality reduction, a way for a model to learn to ignore noise in the data and conserve only relevant features. Because of this, our first step was to implement models from which we could extract the learned latent spaces.

To group patients according to shared feature progressions, we took two approaches. The first was to perform clustering on latent spaces of reduced dimension. The cluster centroids were taken as prototypes and they, along with their assigned clusters, were examined for prominent feature differences. The second approach combined dimensionality reduction and prototype learning into a single pipeline and optimized the latent space for learning prototypes.

## 4.1 Dimensionality Reduction

To achieve dimensionality reduction for time series, we trained two models with different target values and extracted their embeddings[1]. The first was a classifier and the second was a sequence-to-sequence autoencoder. In this section, we discuss the details of both models.

[1]The entire implementation of the models discussed in these sections is available at https://github.com/sarah-mdv/LIPTraAD.

### 4.1.1 Classifier

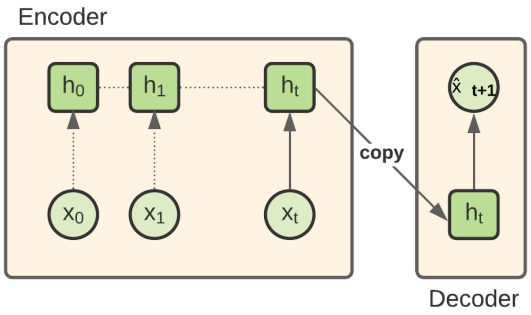



**Figure 4.1:** A high-level diagram of the classifier model. The hidden state $h_t$ is the embedding for sequence $x_0 - x_t$

We used the architecture from a previously implemented and tested recurrent neural network (RNN) classifier that was used to predict progression for Alzheimer's patients on ADNI data [27] as a baseline implementation[2]. We chose an RNN encoder over a more complicated long short-term memory (LSTM) encoder because there were fewer hyperparameters to adjust for and we preferred a simpler and more easily understood architecture. As shown in Figure 4.2, at every time point, the encoder

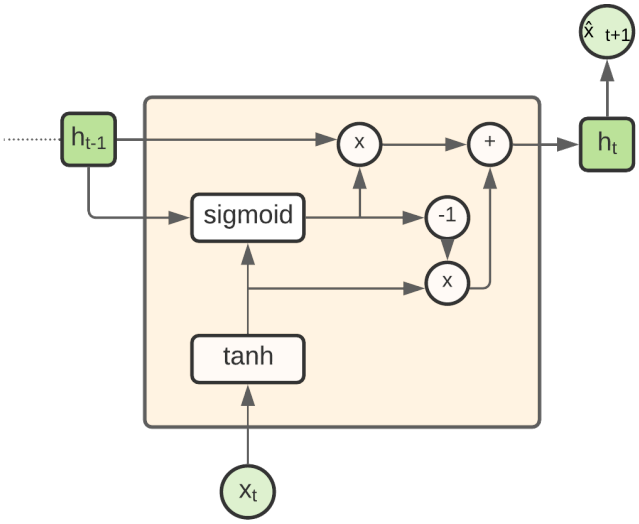



**Figure 4.2:** A detailed diagram of the encoder. See Equations 4.1-4.4.

takes the discrete diagnosis variable $c_t$ and continuous variables $k_t$ as combined input $x_t$.

$$x_t = [c_t, k_t] \tag{4.1}$$

[2]The original implementation is available at https://github.com/ThomasYeoLab/Standalone_Nguyen2020_RNNAD.

The input $x_t$ is scaled by a weight matrix $W_x$ and mapped by an activation function tanh.

$$u_t = \tanh(W_x x_t) \tag{4.2}$$

The hidden state $h_t$ is a combination of the hidden states of previous time points and the transformed current input $x_t$. Both the previous hidden state and the current input are weighted by $f_t$, a weight function that calculates how much value should be given to previous and current inputs. Both the input $h_t$ and $u_t$ are scaled by weight matrices $U_h$ and $W_u$.

$$f_t = \sigma(U_h h_t + W_u u_t) \tag{4.3}$$

$$h_t = f_t \circledast h_{t-1} + (f_t - 1) \circledast u_t \tag{4.4}$$

The target values were the feature values for the next time step.

$$\hat{c}_{t+1} = \text{softmax}(W_s h_t) \tag{4.5}$$

$$\hat{k}_{t+1} = W_g h_t + k_t \tag{4.6}$$

$$\hat{x}_{t+1} = [\hat{c}_{t+1}, \hat{k}_{t+1}] \tag{4.7}$$

The loss for the classifier was the sum of cross-entropy (CE) loss over the diagnosis and mean absolute error (MAE) over the continuous variables:

$$L = \sum(\text{CrossEntropy}(c_t, \hat{c}_t) + \text{MAE}(k_t, \hat{k}_t)) \tag{4.8}$$

$$\text{CrossEntropy}(c_t, \hat{c}_t) = -\sum_{j=1}^{3} c_j^t \log \hat{c}_j^t \tag{4.9}$$

$$\text{MAE}(k_t, \hat{k}_t) = \frac{1}{n} \sum_{j=1}^{n} |k_t^j - \hat{k}_t^j| \tag{4.10}$$

Where $n$ is the number of continuous variables used as input.

### 4.1.2 Sequence-to-Sequence Autoencoder

Since we were most interested in capturing subtleties in patient progression until transition, the second model from which we extracted the embedding layer was an autoencoder. Autoencoders are optimized for this type of task because, by design, they capture features that are relevant to reproducing the data (in contrast, classifiers capture features useful for predicting future data).

We used the same encoder architecture and the same loss functions as for the classifier (Equations 4.1-4.4) but trained the model for different target values. The sequence-to-sequence autoencoder was trained to reproduce the input of the patient diagnosis and feature value progressions (Equations 4.11-4.14).

$$\hat{h}_{t-1} = \sigma(W_h \hat{h}_t + c_h) \tag{4.11}$$

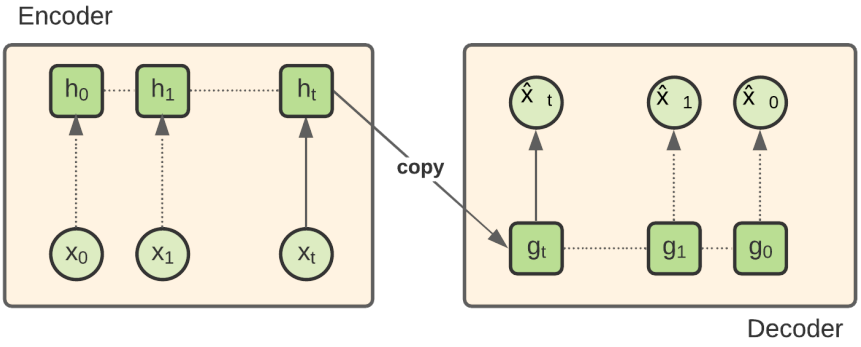



**Figure 4.3:** A high-level diagram of the sequence-to-sequence autoencoder model. The data for the entire sequence is encoded and the decoder predicts back the input sequence in reverse order.

$$\hat{c}_{t-1} = \text{softmax}(W_s \hat{h}_{t-1}) \tag{4.12}$$

$$\hat{k}_{t-1} = W_g \hat{h}_{t-1} + q_g \tag{4.13}$$

$$\hat{x}_{t-1} = [\hat{c}_{t-1}, \hat{k}_{t-1}] \tag{4.14}$$

## 4.2 Prototype Learning through Deep Clustering

For the second part of our task, we wanted to identify patients that were similar with respect to features that had catalyzing effects on their trajectories. Since we had the embeddings from the classifier and the autoencoder capturing these features, we could group patients within these latent spaces. We performed k-means clustering on patients in the two latent spaces and considered the cluster centroids as the prototype $p_k$ for each cluster $k$. We compared the resulting prototypes and their clusters to identify which features were captured differently by the classifier and the autoencoder.

To get an empirical measure for how well the clusters described the dataset, we trained models to perform tasks based only on similarity to the prototypes. The encoder and the clustering were fixed at training time and only the decoder was trained. Figure 4.4 shows how embeddings were used as input for the clustering algorithm and for similarity measures. For every datapoint, we computed the similarity vector $s_i$ where the $k$th element is the similarity to prototype $p_k$:

$$s_i^k = e^{-\|p_k - h_i\|_2^2} \tag{4.15}$$

The similarity vectors were used as input to a linear layer to predict the target values. We did this for the encoders trained for classification and for input reproduction (the autoencoder). We therefore had the target values of $\hat{x}_{t+1}$ and sequence $(\hat{x}_t, \hat{x}_{t-1}, ..., \hat{x}_0)$ respectively.

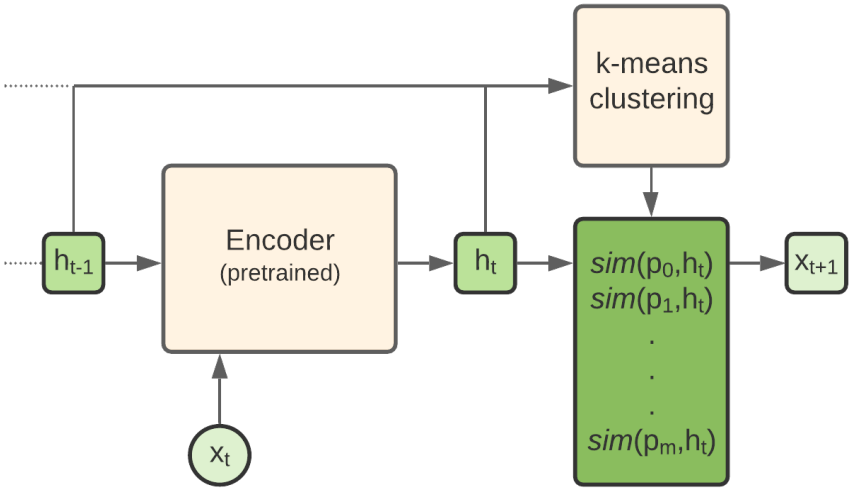



**Figure 4.4:** A high-level diagram of the deep clustering model architecture for the classifier encoder. The encoder is pretrained with diagnosis and continuous features as target values. K-means clustering is performed on the extracted embeddings and the cluster centroids are taken as prototypes. The similarity to the prototypes (Equation 4.15) are used to predict target value $\hat{x}_{t+1}$

## 4.3 Joint Dimensionality Reduction and Prototype Learning

We were also interested in learning prototypes in a latent space that was optimized specifically for that task. The final model we implemented performed joint dimensionality reduction and prototype learning.

This model combined the elements from the previous model into a single pipeline. We used the same encoder and decoder as for the autoencoder, however the input for the decoder was not the hidden state from time $t$, but the similarity between that hidden state and the set of prototypes $p_k$. The similarity measure we used was:

$$sim(p_i, h_t) = e^{-\|h_t - p_i\|_2^2}$$

During back propagation, both the prototypes and the latent space are updated.

### Prototype Projection

In order to achieve the best interpretability for the prototypes, we did not decode the prototypes from the latent space but instead we projected the prototypes onto the closest real datapoint in the latent space ($r(X)$).

$$\boldsymbol{p}_i = \arg\min_{e \in r(X)} \|e - \boldsymbol{p}_i\| \tag{4.16}$$

We then retrieved the real patient progression since it is mapped to the latent space. This ensured that prototypes are only real patient trajectories.

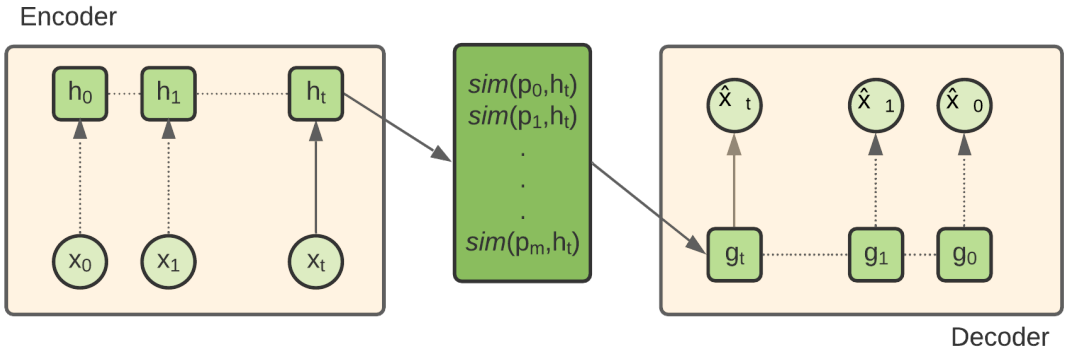



**Figure 4.5:** Autoencoder model with joint dimensionality reduction and prototype learning. The model has an architecture similar to a simple autoencoder, however, instead of taking the last embedding from the encoder as input, the decoder takes a vector of similarities between the embedding and the prototypes. The prototypes are learned in the same latent space as the embeddings and are updated during back propagation.

### Loss

In addition to the reconstruction loss, we add regularization terms to the prototypes to constrain them in the latent space. Diversity is important because it is not useful to have multiple prototypes that describe the same cohort of patients if there are no relevant differences between them. Additionally, it keeps the prototypes from collapsing into a single point. First, we force diversity by encouraging prototypes to be as distinct as possible, penalizing prototypes that are closest.

$$R_d = \sum_{i=0}^{k} \sum_{j=i+1}^{k} max(0, d_{min} - \|p_i - p_j\|_2)^2 \tag{4.17}$$

With $d_{min}$ being the threshold for which we consider two prototypes as similar or not. Second, to increase interpretability as suggested by Li and al. [17], we add clustering regularization and evidence regularization:

$$R_c = \sum_{(x^t)_{t=1}^T} (min_{i=1}^k \|r((x^t)_{t=1}^T) - p_i\|_2^2) \tag{4.18}$$

$$R_e = \sum_{i=i}^{k} (min_{(x^t)_{t=1}^T} \|p_i - r((x^t)_{t=1}^T)\|_2^2) \tag{4.19}$$

The full objective function is:

$$L = \lambda_{ce} CE + \lambda_{mae} MAE + \lambda_c R_c + \lambda_e R_e + \lambda_d R_d \tag{4.20}$$

## 4.4 Preprocessing

We first normalized the continuous variables by mean and standard deviation. The granularity of time points was set to a step size of 6 months. We imputed the missing data copying data from the previous time point (forward filling). Since the imputed values were not ground truth, we excluded them when computing loss during training and validation. For variables that were missing initial measurements or for sequences that were missing certain values in any measurements, we filled them using default values.

## 4.5 Sample Splitting

We performed k-fold cross-validation with $k = 10$ folds, using $k - 1$ sets for training and 1 set for testing. The model was trained independently over $k$ folds rotating the testing set and the results of prediction on the testing set were averaged over the folds. This ensured that our testing set was balanced over the entire dataset and that the model was evaluated based on its performance when seeing new data.

We also treated sub-sequences of trajectories as input for the model. This allowed us to extract trajectories that ended in transitions from MCI to AD and also provided us with more training points.

## 4.6 Hyperparameter ranges

We trained the models with learning rates [0.01, 0.005, 0.001, 0.0001] and weight decay [0.01, 0.005, 0.001, 0.0001]. We tested batch sizes in the range of [1, 2, 4, 8, 16, 32, 64, 128]. The hyperparameter most critical to our project was the number of prototypes or clusters selected for the models. We trained the model with the number of prototypes ranging between 1 and 15 prototypes.

For all the models, small batch sizes produced the best results. The models trained on similarity to prototypes were very sensitive to the hyper parameters. In particular, hyperparameters would vary with respect to the number of clusters or prototypes that were learned.

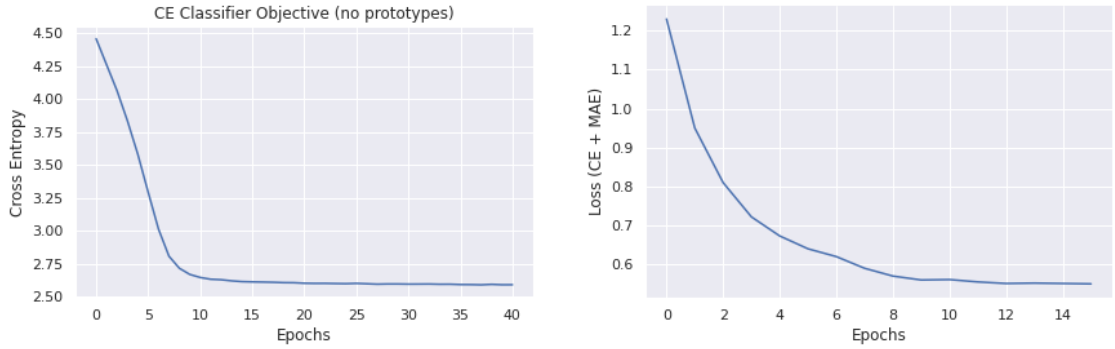
# 5

# Results

Our methodological goal was divided into two parts: dimensionality reduction and prototype learning in the reduced dimension. We can divide the results into two parts as well. We estimated the quality of dimensionality reduction by measuring model performance and analysed the prototype learning by examining how well clusters captured trends in the data.

## 5.1 Model Performance

When training the models used for dimensionality reduction, we used cross entropy loss and mean absolute error as metrics for performance. These metrics gave empirical assessments of how well the models were able to ignore noise and accurately predict or reproduce input. Since our assumption was that training the models would produce latent spaces that captured key features, then model performance was a reasonable proxy measure for how well those features were being represented.




**(a)** Classifier.

**(b)** Autoencoder.

**Figure 5.1:** Loss over epochs for the two dimensionality reduction models. Loss for the classifier was measured by the cross entropy (CE) loss and for the autoencoder by the cross entropy and mean absolute error (MAE) score

Since the classifier was the replication of a previously tested classifier for AD progression on ADNI data (Minimal RNN [27]), we could compare the results of our classifier to ensure that it was producing the expected results. The best results produced by the Minimal RNN classifier in the original paper with forward filling was a

balanced-accuracy score of $0.867 \pm 0.023$. We achieved a balanced accuracy score of $0.863 \pm 0.041$ which was close enough to show that we had a valid implementation.

The performance of the autoencoder was measured by averaging the mean absolute error over all the continuous variables and one minus the balanced accuracy score of the discrete variable (diagnosis). The autoencoder recorded a loss of $0.404 \pm 0.036$. The autoencoder achieved a balanced-accuracy score of $0.923 \pm 0.036$. Note that this score is not comparable to the balanced-accuracy score of the classifier since the two models are estimating different quantities.

We measured how well clustering in the classifier's latent space represented the data by using similarity to cluster centroids for prediction. This gave a metric for the quality of the different clusters. We compared how well this model performed compared to the simple RNN (MinimalRNN) model we chose as a baseline. Figure 5.2 shows performance when varying the numbers of clusters, there was a sharp increase in accuracy until the number of clusters equaled the number of class labels (3). With 4 prototypes, the model achieved a BAC score of $0.834 \pm 0.021$, and increased slightly with the number of clusters.

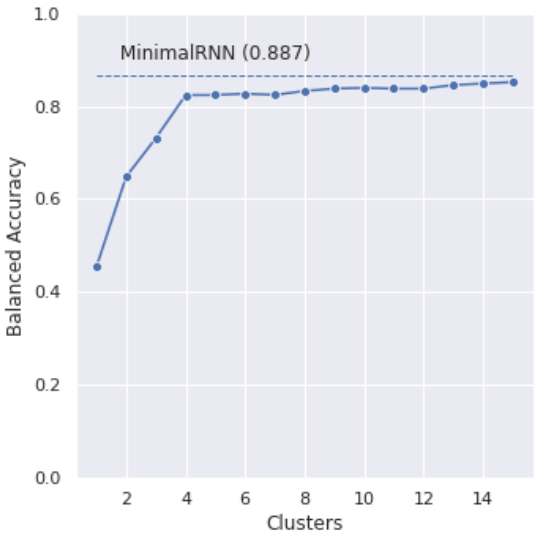



**Figure 5.2:** Balanced accuracy score for the classifier model trained on cluster centroids as prototypes with respect to the number of prototypes used.

We also compared the performance of the autoencoder when reconstruction was based on similarity to prototypes that were cluster centroids versus jointly learned (Figure 5.3). The autoencoder that jointly learned prototypes performed consistently better than the autoencoder based on similarity to cluster centroid prototypes.

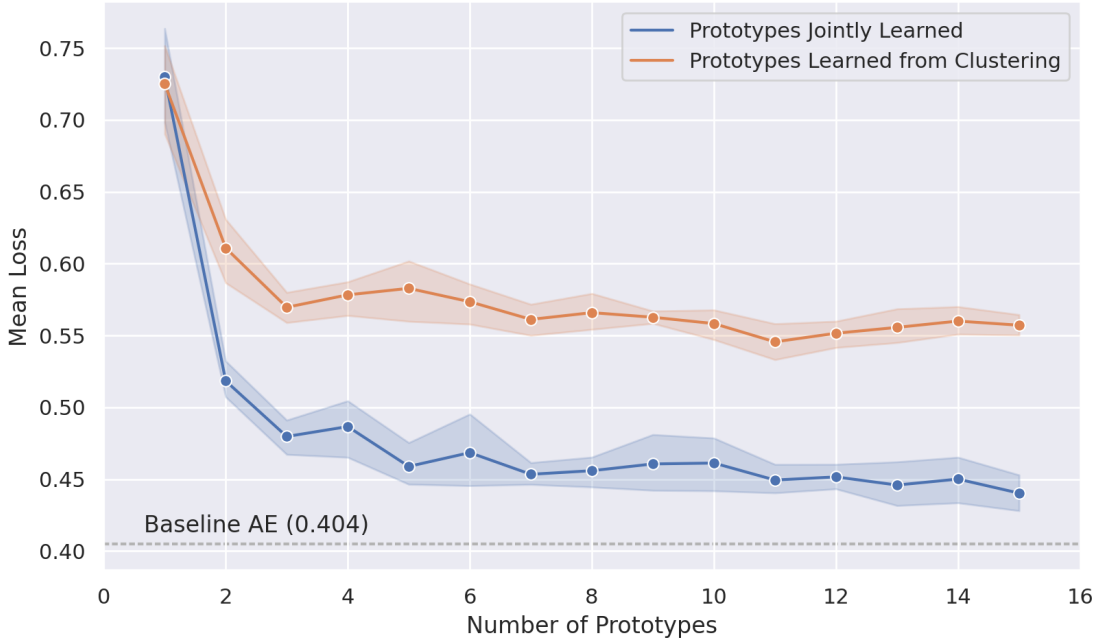



**Figure 5.3:** Average loss for all features predicted by the autoencoder trained on similarity to prototypes with respect to the number of prototypes used. The loss is calculated using cross entropy on discrete variables and mean squared error on continuous variables. The dashed line indicates the score for the baseline autoencoder (without prototypes).

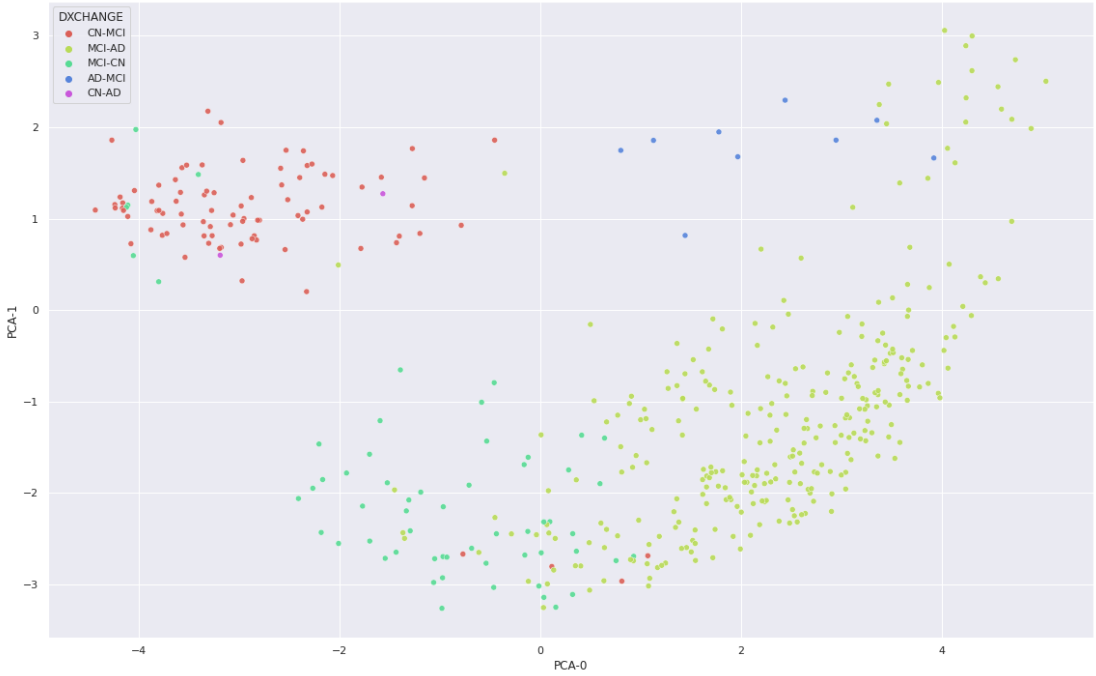
## 5.2 Cluster Inspection

To understand how the prototypes were distributed in the latent space, we made visualizations of the data and its labels. We performed Principal Component Analysis (PCA) on the embedded dataset to extract the salient dimensions. Figures 5.4a and 5.5a show the first two principal components on the axes and the color shows diagnosis.

### General Trends

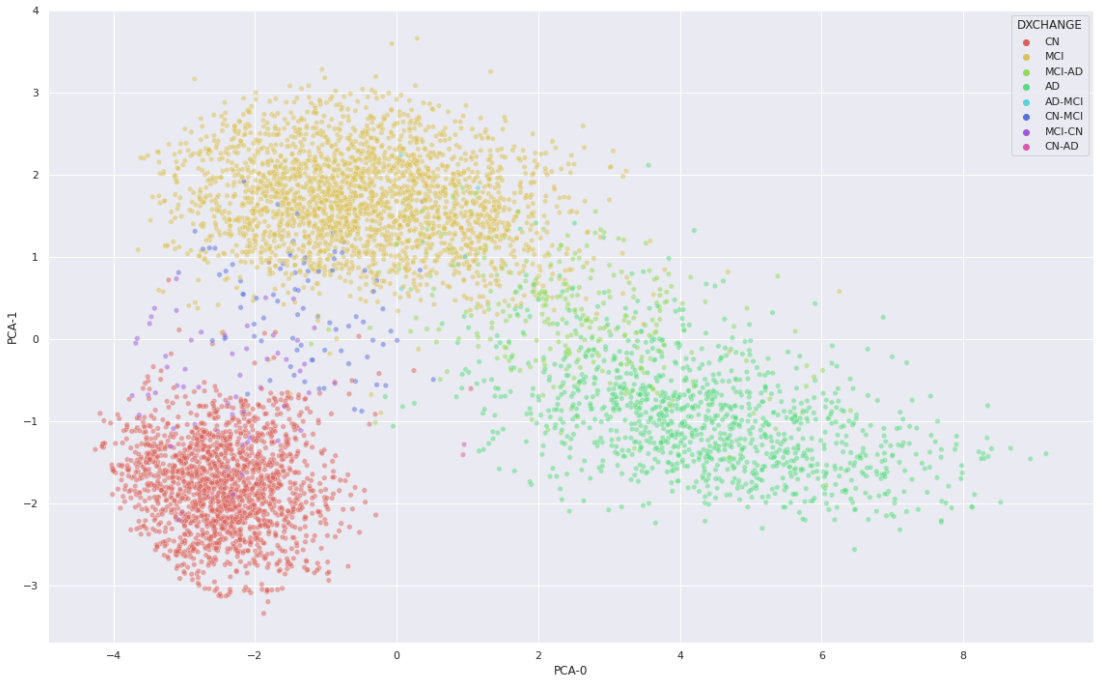
The latent spaces from both the classifier and autoencoder show clear groupings of points with respect to their diagnosis (Figures 5.4a and 5.5a). However for the classifier embedding, transition points are considered most similar to the groups of their diagnosis previous to transition (i.e. MCI to AD patients are indistinguishable from MCI patients (Figure 5.4b)). The autoencoder embedding shows a more clear distinction of transitioners from non-transitioners but still shows overlap (Figure 5.5b).


**(a)** All patient datapoints.

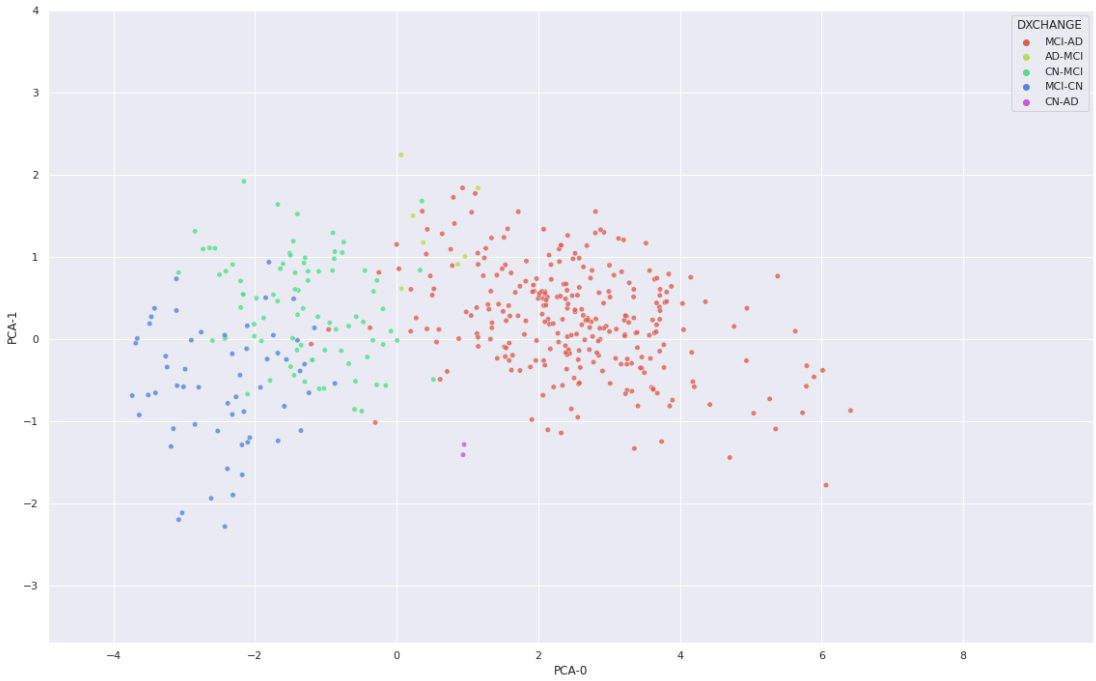



**(b)** Only datapoints from patients who have recorded a transition in diagnosis at time $t$.

**Figure 5.4:** Visualization of the two principal components for all datapoints encoded by the **classifier**. The color indicates the change in diagnosis at time $t$ from time $t-1$. A label of CN, MCI, or AD indicate no change has occurred since the previous measurement.




**(a)** All patient datapoints.




**(b)** Only datapoints from patients who have transitioned at time $t$ are shown.

**Figure 5.5:** Visualization of the two principal components for all datapoints encoded by the **autoencoder**. The color indicates the change in diagnosis at time $t$ from time $t-1$. A label of CN, MCI, or AD indicate no change has occurred since the previous measurement.

Even though transition points are more associated with their previous diagnosis, they show variation among themselves. For example MCI to AD patients and MCI to CN patients are clustered with MCI patients but are closer to AD and CN patients respectively. The autoencoder embedding shows a more clear separation of the transition points from their label prior to transition.

## Transitioner Trends

We performed clustering on the embeddings at transition time for patients transitioning from MCI to AD and took the centroids of the clusters as their respective prototypes(Figure 5.6). Visualizations of clusters in these two latent spaces show differences in trajectories' distributions. Below, we discuss the feature patterns observed in the clusters.

We refer to the clusters from the classifier embedding clusters and the autoencoder embedding clusters as $i_{class}$ and $i_{ae}$ respectively, where $i$ corresponds to the cluster number and the label corresponds to the embedding. For the given clusterings, the clusters $0_{class}$ and $2_{ae}$ are most similar to datapoints with MCI labels while clusters $1_{class}$ and $3_{ae}$ are closest to datapoints with AD labels.

**Age and Gender:** The patients across the clusters in both latent spaces have an even distribution with respect to their age. This was more surprising than not, since the rate of cognitive decline is sometimes associated with age at the time of onset. The clusters in the classifier latent space show a relatively even distribution of gender composition while the clusters in the autoencoder latent space are much more irregular. The result that stands out the most is cluster $0_{ae}$, of which nearly 80% of the patients were women.

| Cluster | % Women | Age(mean) |
|---|---|---|
| 0 | 40% | 73.2 ± 7.1 |
| 1 | 59% | 72.1 ± 7.4 |
| 2 | 35% | 73.1 ± 6.3 |
| 3 | 43% | 75.1 ± 7.3 |

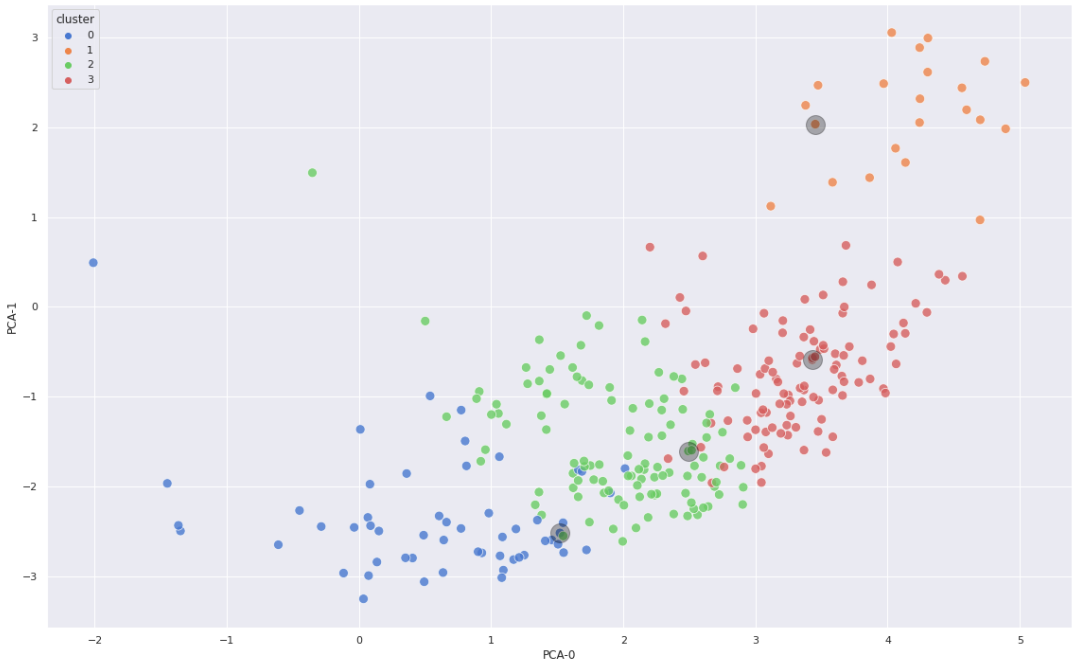
**(a)** Classifier

| Cluster | % Women | Age(mean) |
|---|---|---|
| 0 | 79% | 73.5± 7.7 |
| 1 | 19% | 74.6 ± 6.1 |
| 2 | 22% | 73.7 ± 7.6 |
| 3 | 53% | 74.5 ± 7.0 |

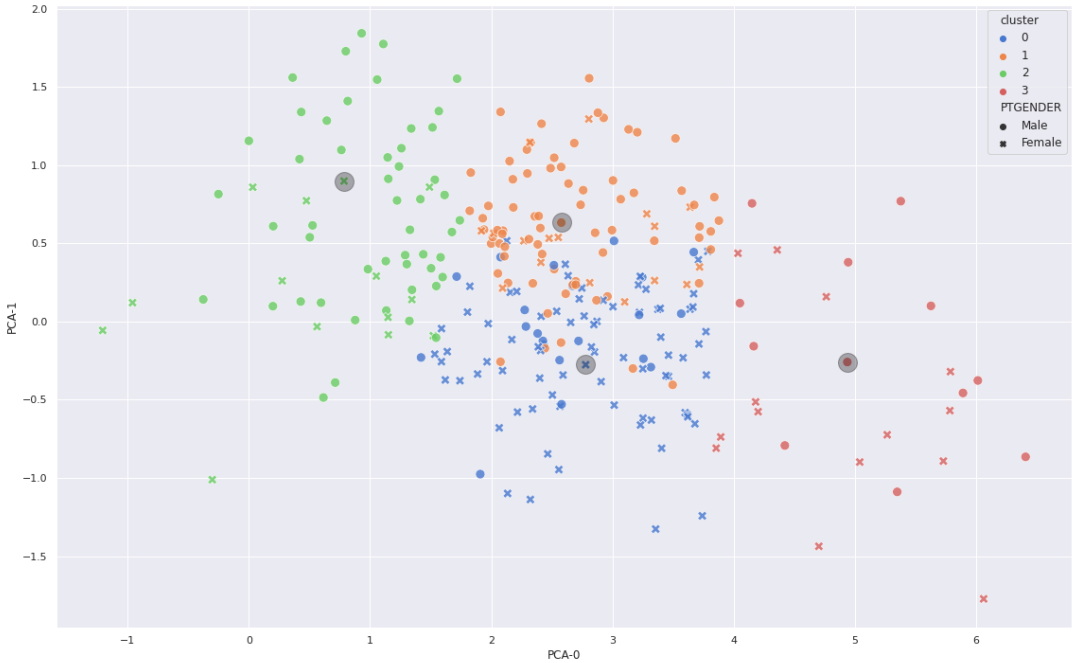
**(b)** Autoencoder

**Table 5.1:** Gender and age distributions for the clusters in the two latent spaces.

**Cognitive Function:** The rate of cognitive decline seems to be a defining feature of the clusters. Clusters $0_{class}$ and $2_{ae}$ show on average a lower ADAS13 score at the time of transition, while clusters $1_{class}$ and $3_{ae}$ show on average a higher score along with worse deterioration both before and after diagnosis (see Table 5.2). Clusters $1_{class}$ and $3_{ae}$ also show greater average decrease in performance 12 months after transition. Clusters $2_{class}$ and $3_{class}$, $0_{ae}$ and $1_{ae}$, show different scores at the time of transition, but similar deterioration patterns. This is consistent with subtypes that have been identified in literature that are distinguished by their rate of cognitive decline [12]. The rates of decline are associated with age at the time of onset of AD.




**(a)** Classifier: patients on the left bottom corner are closer to CN patients and patients on the right top corner are closer to patients already diagnosed with AD.




**(b)** Autoencoder: patients on the right bottom corner are closer to AD patients and patients on the left top corner are closer to MCI patients.

**Figure 5.6:** Visualization of the two principal components of latent spaces for patients who have transitioned from MCI to AD. The colors indicate cluster assignments and the larger circles indicate cluster centroids.

| Cluster | $\Delta t - 12$ | $t$ | $\Delta t + 12$ |
|---|---|---|---|
| 0 | - 4.8 | 19.5 | +1.8 |
| 1 | - 8.1 | 36.5 | +3.8 |
| 2 | - 4.1 | 24.3 | +4.0 |
| 3 | - 6.1 | 31.6 | +5.3 |

**(a)** Classifier

| Cluster | $\Delta t - 12$ | $t$ | $\Delta t + 12$ |
|---|---|---|---|
| 0 | - 4.2 | 24.7 | +2.4 |
| 1 | - 3.1 | 19.3 | +2.6 |
| 2 | - 3.9 | 15.2 | +4.9 |
| 3 | - 7.3 | 34.9 | +5.5 |

**(b)** Autoencoder

**Table 5.2:** Average ADAS13 scores for clusters at transition time $t$, change with respect to that measure at 12 months before and 12 months after recorded transition time $t$. Higher scores indicate worse cognitive function.

| Cluster | $\Delta t - 12$ | $t$ | $\Delta t + 12$ |
|---|---|---|---|
| 0 | +0.38 | 3.91 | -0.21 |
| 1 | +0.79 | 2.14 | -0.32 |
| 2 | +0.77 | 2.58 | -0.08 |
| 3 | +0.41 | 2.29 | -0.15 |

**(a)** Classifier

| Cluster | $\Delta t - 12$ | $t$ | $\Delta t + 12$ |
|---|---|---|---|
| 0 | +1.24 | 2.27 | +0.07 |
| 1 | +0.27 | 2.03 | -0.27 |
| 2 | +0.41 | 3.68 | -0.08 |
| 3 | +0.21 | 2.13 | -0.12 |

**(b)** Autoencoder

**Table 5.3:** Average RAVLT learning score for clusters 12 months before, at, and 12 months after recorded transition time $t$. Lower scores indicate a worse cognitive performance.

**Brain Size, Tau and A-Beta:** For the two clusters $2_{class}$ and $3_{class}$, and $0_{ae}$ and $1_{ae}$, the autoencoder seems to identify features that are not distinguished by the classifier. Notably, cluster $0_{ae}$ shows a lower whole brain size than cluster $1_{ae}$. This seems to be correlated with cluster $0_{ae}$ containing primarily female patients (see Table 5.1). One observation is that while clusters $0_{ae}$ and $1_{ae}$ have different average ADAS scores at the time of transition, they show similar progression rates following AD diagnosis. This could also be linked to the gender variation and is discussed more in the following paragraph. The clusters $2_{class}$ and $3_{class}$ do not seem to capture these differences and show an even distribution of brain size across the clusters.

| Cluster | $\Delta t - 12$ | $t$ | $\Delta t + 12$ |
|---|---|---|---|
| 0 | +0.02 | 1.00 | +0.01 |
| 1 | -0.01 | 0.96 | +0.02 |
| 2 | +0.02 | 1.00 | -0.02 |
| 3 | +0.03 | 0.93 | -0.01 |

**(a)** Classifier

| Cluster | $\Delta t - 12$ | $t$ | $\Delta t + 12$ |
|---|---|---|---|
| 0 | +0.01 | 0.93 | -0.02 |
| 1 | -0.01 | 1.09 | 0.00 |
| 2 | 0.00 | 1.03 | -0.02 |
| 3 | +0.01 | 0.86 | -0.02 |

**(b)** Autoencoder

**Table 5.4:** Average whole brain size for clusters 12 months before, at, and 12 months after recorded transition time $t$ (scaled by e-06). Smaller brain sizes are associated with worse cognitive decay in Alzheimer's patients.

| Cluster | $\Delta t - 12$ | $t$ | $\Delta t + 12$ |
|---|---|---|---|
| 0 | 316.5 | 252.4 | 318.8 |
| 1 | 508.2 | 359.7 | 463.4 |
| 2 | 347.8 | 378.1 | 260.5 |
| 3 | 366.8 | 401.5 | 394.8 |

**(a)** Classifier

| Cluster | $\Delta t - 12$ | $t$ | $\Delta t + 12$ |
|---|---|---|---|
| 0 | 331.3 | 432.8 | 302.7 |
| 1 | 280.1 | 291.5 | 264.0 |
| 2 | 348.9 | 332.9 | 332.5 |
| 3 | 505.9 | 528.2 | 705.7 |

**(b)** Autoencoder

**Table 5.5:** Average tau for clusters 12 months before, at, and 12 months after recorded transition time $t$. Higher levels of tau are correlated with more severe cases of AD.

| Cluster | $\Delta t - 12$ | $t$ | $\Delta t + 12$ |
|---|---|---|---|
| 0 | 910.4 | 910.7 | 1070.2 |
| 1 | 614.0 | 560.7 | 500.4 |
| 2 | 684.5 | 747.2 | 684.8 |
| 3 | 573.8 | 531.3 | 505.4 |

**(a)** Classifier

| Cluster | $\Delta t - 12$ | $t$ | $\Delta t + 12$ |
|---|---|---|---|
| 0 | 787.9 | 941.9 | 834.8 |
| 1 | 890.1 | 896.7 | 1019.7 |
| 2 | 550.5 | 538.4 | 495.8 |
| 3 | 617.0 | 541.9 | 512.4 |

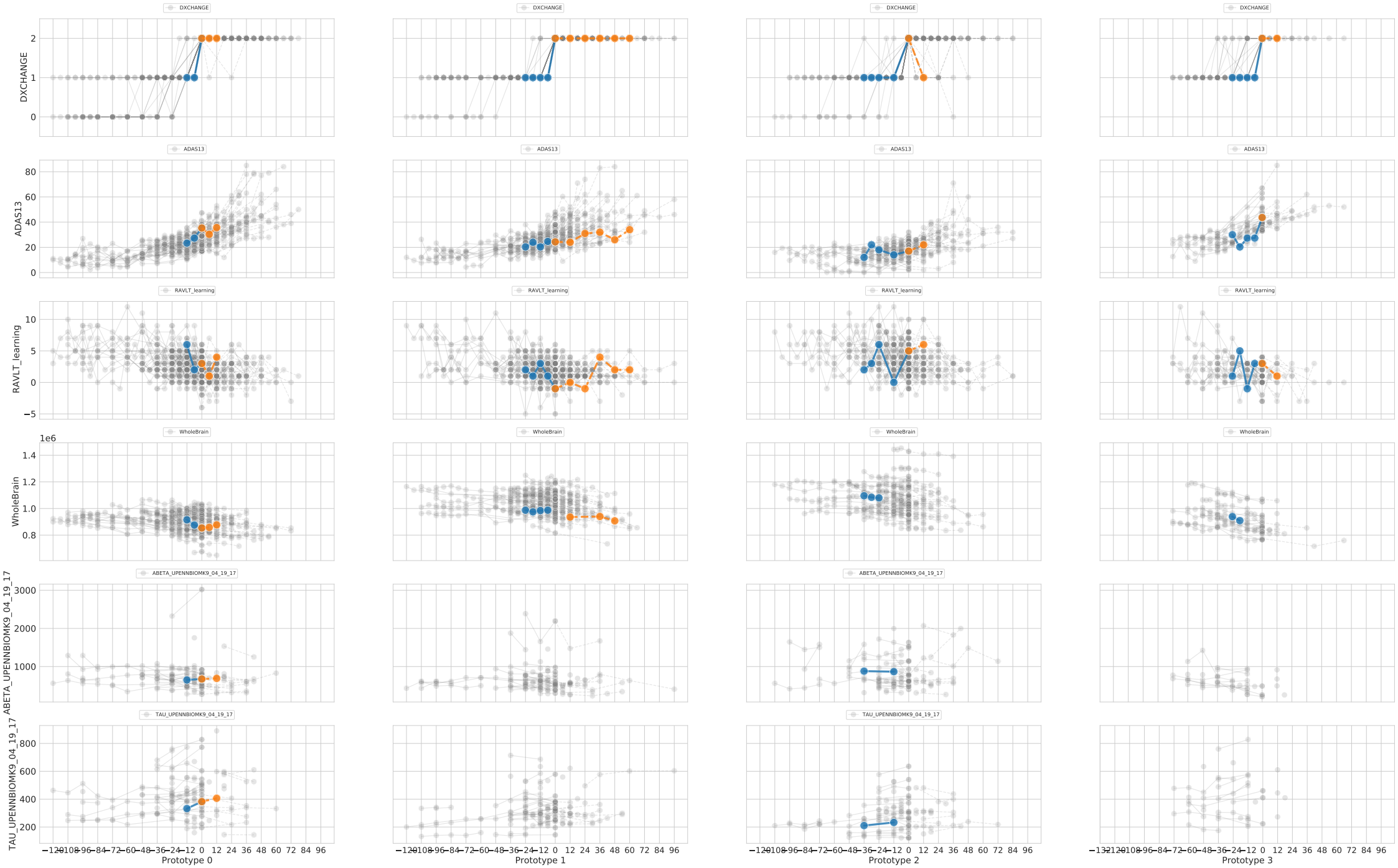
**(b)** Autoencoder

**Table 5.6:** Average A-beta for clusters 12 months before, at, and 12 months after recorded transition time $t$.

## 5.3 Prototypes

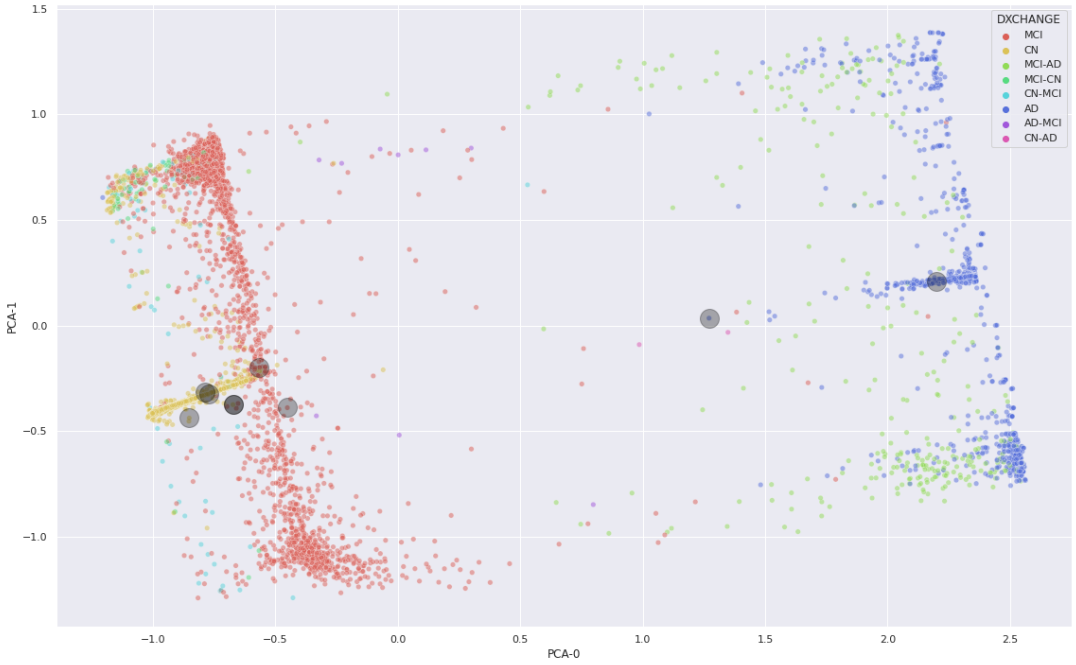
### 5.3.1 Clustering Prototypes

We compared the prototypes from the clustering on the autoencoder latent space (the cluster centroids) to the general trends in the clusters. The figure below shows the prototype progressions. The grey lines show the progressions of all patients in the cluster while the blue and orange line shows the progression of the prototype. The point where the line changes from blue to orange indicates the point at which the patient transitions from MCI to AD. For certain prototypes, no measurements were taken for certain features, hence the absence of certain colored lines.

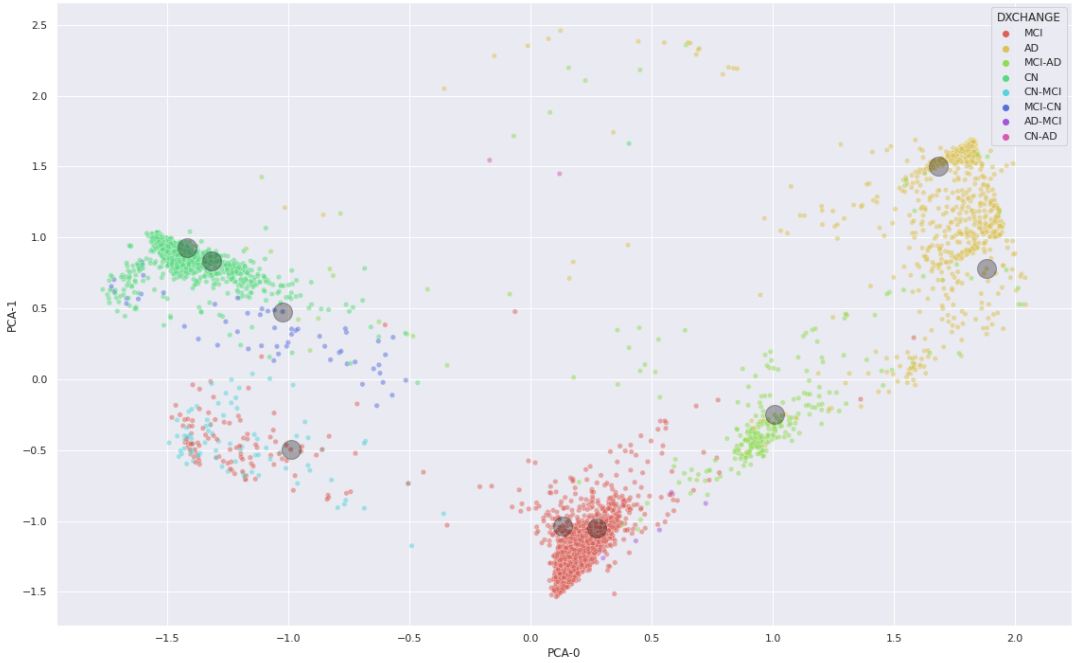

DXCHANGE
ADAS13
RAVLT_learning
WholeBrain
1e6
ABETA_UPENNBIOMK9_04_19_17
TAU_UPENNBIOMK9_04_19_17
Prototype 0
Prototype 1
Prototype 2
Prototype 3

### 5.3.2 Jointly Learned Prototypes

Visualizations of the latent space learned by the model with joint prototype optimization showed less consistent results than the other two models. While the model performed well with respect to the empirical objective, the PCA visualizations show that prototypes adhere to expected distribution across the data only occasionally. For example, the plots in Figures 5.7 and show the PCA plots of embeddings trained with 9 prototypes for different folds of the data. The prototypes in Figure 5.7b look fairly well distributed across the data with prototypes representing significant subsets of the data. In Figure 5.7a, the prototypes do not seem to follow patterns in the data. Since the PCA plots only represent two axes of variation in the data, these observations are not absolute reflections of the embeddings. However, given the regularization terms, we would expect that the prototypes not be outlier datapoints but converge towards denser areas.




**(a)**




**(b)**

**Figure 5.7:** Visualization of the two principal components for all datapoints in the latent space learned by the **prototype based model** with 9 prototypes. The figures represent the embeddings learned for two different folds of the data. The color indicates the change in diagnosis at time $t$ from time $t-1$.

# 6

# Discussion

The primary goal in this project was to identify prototypes that distinguish between subtypes of patients who transition from MCI to AD. We defined sub-goals that would allow us to reach that initial goal.

- *How should feature progression be captured ? What features are relevant for distinguishing among subgroups ?*
- *Are there identifiable and distinctive traits for the subgroups of patients ? How can subgroup quality be measured ?*
- *How can prototypes be forced to represent salient features ? What is the best way to present a prototype such that it is useful and understandable ?*
- *How can prototype quality be measured ?*

In the following sections, we discuss how well these questions were answered and how our methods were chosen to answer them.

## 6.1 Dimensionality Reduction

*How should feature progression be captured ? What features are relevant for distinguishing among subgroups ?*

Sequence data are challenging to evaluate because the complexity of analysing across features and over time grows exponentially with the size of the feature set. Further, models trained on raw high dimensional data are susceptible to highlighting superficial patterns that do not reflect underlying structure in the data. Sequence data also frequently has unequal lengths, requiring some form of uniformization so they can be effectively compared. To solve these problems, we need to compress the data to a form that preserves only salient features and that has a consistent format. There are a number of ways to perform dimensionality reduction: feature selection, matrix factorization, and others. Another form of dimensionality reduction is extracting the embedding of a model that is trained for a specific task.

The two dimensionality reduction models in this project were trained for different

tasks. The first was trained to predict future outcomes and the second was trained to reproduce its input. Our hypothesis was that variables that were significant axes for distinction among progressions were not necessarily important for prediction. We were able to test this by comparing the clusters that were learned on the embedding from the classifier with those learned from the autoencoder: the clusters differed discernibly with respect to certain variables. This confirmed that we were capturing features that were relevant to distinguishing among subgroups.

Identifying feature patterns in certain subgroups also allows elimination of certain other features that do not show a variation in distribution. Refining the set of input values is similar to dimensionality reduction technique by feature selection based on correlation between features and target values. This technique helps refine complexity of the input and helps the model put more emphasis on features that are most relevant to progression. While we did not do this in this thesis because of limited time, it is an interesting direction for future work.

## 6.2 Cluster Analysis

*Are there identifiable and distinctive traits for the subgroups of patients ?*

As we showed in Section 5.2, clusters can be distinguished based on certain traits such as cognitive function and brain size. Of the patterns we identified in the clusters, the most significant was the predominantly female cluster $0_{ae}$ with trends of normal cognitive decline but superior verbal memory. For recall, the key features of this group were the following:

- Average general cognitive performance and deterioration (ADAS13)
- Above average verbal memory skills (RAVLT learning score)
- Lower average whole brain size
- More rapid decline in hippocampal size

Past studies suggest that in general women have better verbal memory than men, and that they keep this advantage during MCI and early stages of AD [35] in spite of more severe brain atrophy. This is consistent with patients in cluster $0_{ae}$ who have better average RAVLT learning scores (which tests verbal memory) even while performing worse on the ADAS13 cognitive scoring compared to patients in cluster $1_{ae}$. Because they maintain function for longer in the disease progression, patients who exhibit better verbal memory are usually diagnosed at a later stage [35]. This subsequently leads to a worse progression of symptoms once diagnosed as the verbal advantage is lost in later stages of AD [26]. This primarily female cluster with more rapid decline of cognition after transition is therefore consistent with trajectories for certain women [26].

The other clusters did show distinctive trends for features such as rate of cognitive

decline and brain size. Moreover, the clusters that show specific rates of cognitive decline show consistent rates of evolution after diagnosis. This is relevant because those trends after diagnosis have been identified as subtypes of AD. What we leave to future work is an understanding of how the subtypes of patients who transition to AD relates to the known subtypes.

## 6.3 Prototype Analysis

*How can prototypes be forced to represent salient features ? What is the best way to present a prototype such that it is useful and understandable ?*

This question was one of the hardest to answer, and probably the one for which we have the least conclusive results. As we discuss in Section 6.4, our qualitative assessment of prototypes is limited by lack of experience with AD. A clinician would be better equipped than us to notice interesting patterns in the clusters.

### 6.3.1 Cluster Prototypes

Recalling our original task, we were interested in describing clusters by prototypical patients. Looking at how the prototypes progress compared to their clusters, they captured some trends in the key features for their clusters. For example, we considered the RAVLT learning score to be an important feature for defining cluster 0 in the autoencoder latent space. The trend we saw in the cluster was a decrease in score before transition but stabilization after transition (see Table 5.3b). Looking at the prototype for that cluster, we saw the same trend: a relatively sharp decrease in score from 12 months before transition, but then a slight improvement 12 months after transition.

However, other feature progressions with distinctive trends were not captured by the prototypes. It may be that those feature progressions were not actually relevant to the distinction among subtypes. Another possibility is that prototypes, which for the clusters were the projected cluster centroids, did not capture the most important features. A solution to identifying whether these features are irrelevant is to the clustering would be to eliminate them from the input feature set and comparing the subsequent clusters.

### 6.3.2 Jointly Learned Prototypes

In the PCA visualizations, the prototypes seemed to generally follow an arbitrary distribution. There could be several reasons for this. First, the projections are not necessarily representative of the actual embeddings. Since the plots can only show two axes of variation, there could be patterns that the model is detecting but that cannot be show in two dimensions. Second, the model was quite sensitive to hyperparameter adjustment. Changing the values of variables like regularization weight have a significant impact on how prototypes are distributed. While we did experiment with those variables, a methodological approach could refine the parameters

and produce more consistent results.

Experimenting with these parameters may require trading off on model performance, however. For example, while Figure 5.7b shows apparently good distribution of prototypes across the data, the model performance was 0.035 worse than the model whose prototypes did not look well distributed (Figure 5.7a).

In contrast to the pretrained embeddings, we did not have prototypes that were constrained to the transitioner datapoints. This was a step that was planned, but we did not end up having sufficient time to complete this. For some of the embeddings we did not even have transitioners represented in the prototype subset. Because of this, we do not have prototypes trajectories to compare to the pretrained clusters trained only on the transitioner data. Since visualizing the embedded data using dimensionality reduction does not seem to capture patterns in the data, constraining prototypes to include transitioners and examining those prototypes as we did with the cluster prototypes will be a meaninful next step. It would show whether the prototypes are indeed picking up on patterns, or whether the model needs further tuning.

## 6.4 Future Work

### 6.4.1 Quantitative Prototype Analysis

A methodology for quantitatively evaluating prototypes would give a tangible goal for a currently subjective task. One of our questions when starting this thesis was: How can prototype and subgroup quality be measured? To some degree, we answered this question by training models based on similarity to the prototypes: the quality of the prediction relates to how well the prototypes describe variance in the data. However, this does not give an answer to how we can measure the quality of prototypes. Past work using prototypes has often fixed model performance as the only measure of how well the prototypes describe the data since the prototypes' primary function is to explain the prediction [17]. However, our experience in this project has been that prototypes as underlying structure to models can show good performance without being meaningful.

### 6.4.2 Qualitative Prototype Analysis

An informed analysis of the prototypes by a clinician could help identify patterns that are novel. In this thesis, we only performed a superficial qualitative analysis on the prototypes. Our knowledge about Alzheimer's progression is limited to the subset of literature we examined throughout the thesis. Similar to feedback loops integrated into other prototype models [22], a natural next step would be to show resulting prototypes to an AD specialist so that we could get feedback on how well they captured trends compared to patients in the real world. This feedback could be fed directly back into the model for a human in the loop design.

### 6.4.3 Feature Selection

Further feature selection could effectively reduce dimensionality and help detect deeper patterns in the data. In particular, we have identified a selection of features that seem to be significant axes of variation among the trajectories. One direction for research could be refining the set of features we reconstruct with the autoencoder while keeping all features as input. This could help guide prototypes to target most relevant features. As discussed in Section 6.1, we mentioned feature selection based on correlation between features and target values. This, among other feature selection techniques, could help eliminate redundant features. However, one danger is eliminating features that are not important for outcome, but that are significant for variation among subgroups.

# 7

# Conclusion

Alzheimer's Disease renders patients incapacitated and has repercussions on both family members and close communities. It is incurable despite being the focus of many research projects and initiatives. Understanding how the disease progresses can help understand its pathology and add to the body of research that is contributing to finding treatments and medications.

The question we set out to answer at the beginning of the thesis was this: For patients who share the same final outcome (transition from MCI to AD), what are differences in their progression that are meaningful in distinguishing them from one another. We were looking both for a way of discovering subgroups of these patients and for a way to explicate these subgroups.

We hypothesized that certain features recorded at patient visits help identify subgroups even if they are not predictive. To test this hypothesis, we performed dimensionality reduction using models with two different sets of target values. One model performed prediction and the other performed reconstruction. To understand how the latent spaces varied, we performed deep clustering on the learned embeddings and examined the resulting clusters. The model that performed reconstruction highlighted gender as a major axis of variation among clusters while the model that performed prediction did not. This result suggested that there are features that are important for distinguishing among trajectories even if they are not causally related to outcome.

Since we were interested in characterizing the clusters in an interpretable manner, we represented each cluster by a prototypical point belonging to that cluster. To verify that the clusters and their prototypes were meaningful in the real world data, we used the cluster centroids in the latent space as prototype points and trained models to perform prediction based on similarity to those prototypes. Projecting the prototypes back to the raw data, we compared their progression to that of their clusters. The prototypes reflected trends in the clusters for some of the features, but not for all, implying that either the prototypes are not good representatives of the clusters or that those features are not important for subgroup distinction.

We also hypothesized that learning a latent space optimized for learning prototypes by integrating the prototype learning into the model instead of performing it post

hoc could help identify subtilities between subgroups at a finer granularity. We tested this by adding a layer of prototypes to the autoencoder model and performed reconstruction based on similarity to those prototypes. We compared the results from this model to those of the autoencoder where the prototypes were the cluster centroids. The model that trained prototypes jointly performed better than the one with fixed prototypes with respect to the objective function. This confirms that it could be interesting future work to look at how the prototypes from the two models compare when projected back to raw data.

Altogether, we identified distinct subgroups among patients who transition from MCI to AD and confirmed that some features that are important to their distinction are different from features that are predictive. Further, we confirmed that learning prototypes jointly could be an interesting method for identifying more specific prototypes and would be an interesting direction of future work.

# Bibliography


[1] Alzheimer's disease neuroimaging initiative. `http://adni.loni.usc.edu/`.

[2] Ravinder Ahuja, Aakarsha Chug, Shaurya Gupta, Pratyush Ahuja, and Shruti Kohli. *Classification and Clustering Algorithms of Machine Learning with their Applications*, pages 225–248. Springer International Publishing, Cham, 2020.

[3] Ingrid Arevalo-Rodriguez, Nadja Smailagic, Agustín Ciapponi, E. Sanchez-Perez, Andry Giannakou, M. Figuls, Olga Pedraza, Xavier Bonfill, and Sarah Cullum. Mini-mental state examination (mmse) for the detection of alzheimer's disease and other dementias in people with mild cognitive impairment (mci). *Cochrane Database of Systematic Reviews*, 2013, 10 2013.

[4] Inci M. Baytas, Cao Xiao, Xi Zhang, Fei Wang, Anil K. Jain, and Jiayu Zhou. Patient subtyping via time-aware lstm networks. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '17, page 65–74, New York, NY, USA, 2017. Association for Computing Machinery.

[5] Joseph Breault. Protecting human research subjects: The past defines the future. *The Ochsner journal*, 6:15–20, 03 2006.

[6] Jason Brownlee. A gentle introduction to lstm autoencoders. `https://machinelearningmastery.com/lstm-autoencoders/`, Nov 2018.

[7] Benjamin A Clegg, Gregory J DiGirolamo, and Steven W Keele. Sequence learning. *Trends in Cognitive Sciences*, 2(8):275–281, 1998.

[8] Scott L Fleming, Kuhan Jeyapragasan, Tony Duan, Daisy Ding, Saurabh Gombar, Nigam Shah, and Emma Brunskill. Missingness as stability: Understanding the structure of missingness in longitudinal ehr data and its impact on reinforcement learning in healthcare. *arXiv preprint arXiv:1911.07084*, 2019.

[9] Arthritis Foundation. Alzheimer's disease. `https://www.nia.nih.gov/health/alzheimers`, 2021.

[10] Marzyeh Ghassemi, Tristan Naumann, Peter Schulam, Andrew L Beam, Irene Y Chen, and Rajesh Ranganath. A review of challenges and opportunities in machine learning for health. *AMIA Summits on Translational Science Proceedings*, 2020:191, 2020.

[11] Alzheimer's Disease International. Alzheimers and dementia. `https://www.alzint.org/about/`, 2020.

[12] Kurt A Jellinger. Pathobiological subtypes of alzheimer disease. *Dementia and Geriatric Cognitive Disorders*, 49:1–13, 01 2021.

[13] Igor Korolev, Laura Symonds, and Andrea Bozoki. Predicting progression from mild cognitive impairment to alzheimer's dementia using clinical, mri, and plasma biomarkers via probabilistic pattern classification. *PLOS ONE*, 11:e0138866, 02 2016.

[14] Jacqueline Kueper, Mark Speechley, and Manuel Montero-Odasso. The alzheimer's disease assessment scale–cognitive subscale (adas-cog): Modifications and responsiveness in pre-dementia populations. a narrative review. *Journal of Alzheimer's Disease*, 63:1–22, 04 2018.

[15] Christoph Laske, Hamid R. Sohrabi, Shaun M. Frost, Karmele López de Ipiña, Peter Garrard, Massimo Buscema, Justin Dauwels, Surjo R. Soekadar, Stephan Mueller, Christoph Linnemann, Stephanie A. Bridenbaugh, Yogesan Kanagasingam, Ralph N. Martins, and Sid E. O'Bryant. Innovative diagnostic tools for early detection of alzheimer's disease. *Alzheimer's and Dementia*, 11(5):561–578, 2015.

[16] Changhee Lee and Mihaela Van Der Schaar. Temporal phenotyping using deep predictive clustering of disease progression. In *International Conference on Machine Learning*, pages 5767–5777. PMLR, 2020.

[17] Oscar Li, Hao Liu, Chaofan Chen, and Cynthia Rudin. Deep learning for case-based reasoning through prototypes: A neural network that explains its predictions. 2017.

[18] Zachary C Lipton, David C Kale, Randall Wetzel, et al. Modeling missing data in clinical time series with rnns. *Machine Learning for Healthcare*, 56, 2016.

[19] Razvan V. Marinescu, Neil P. Oxtoby, Alexandra L. Young, Esther E. Bron, Arthur W. Toga, Michael W. Weiner, Frederik Barkhof, Nick C. Fox, Stefan Klein, Daniel C. Alexander, and the EuroPOND Consortium. Tadpole challenge: Prediction of longitudinal evolution in alzheimer's disease. 2018.

[20] Rodrigo Medeiros, David Baglietto-Vargas, and Frank Laferla. The role of tau in alzheimer's disease and related disorders. *CNS neuroscience & therapeutics*, 17:514–24, 10 2011.

[21] Erxue Min, Xifeng Guo, Qiang Liu, Gen Zhang, Jianjing Cui, and Jun Long. A survey of clustering with deep learning: From the perspective of network architecture. *IEEE Access*, 6:39501–39514, 2018.

[22] Yao Ming, Panpan Xu, Huamin Qu, and Liu Ren. Interpretable and steerable sequence learning via prototypes. *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery amp; Data Mining*, page 903–913, 2019.

[23] Christoph Molnar. *Interpretable Machine Learning*. 2019. https://christophm.github.io/interpretable-ml-book/.

[24] Susanne G Mueller, Michael W Weiner, Leon J Thal, Ronald C Petersen, Clifford R Jack, William Jagust, John Q Trojanowski, Arthur W Toga, and Laurel Beckett. Ways toward an early diagnosis in alzheimer's disease: the alzheimer's disease neuroimaging initiative (adni). *Alzheimer's & Dementia*, 1(1):55–66, 2005.

[25] Melissa E Murray, Neill R Graff-Radford, Owen A Ross, Ronald C Petersen, Ranjan Duara, and Dennis W Dickson. Neuropathologically defined subtypes of alzheimer's disease with distinct clinical characteristics: a retrospective study. *The Lancet Neurology*, 10(9):785–796, 2011.

[26] Rebecca Nebel, Neelum Aggarwal, Lisa Barnes, Aimee Gallagher, Jill Goldstein, Kejal Kantarci, Monica Mallampalli, Elizabeth Mormino, Laura Scott, Wai Yu, Pauline Maki, and Michelle Mielke. Understanding the impact of sex and gender in alzheimer's disease: A call to action. *Alzheimer's and dementia : the journal of the Alzheimer's Association*, 14, 05 2018.

[27] Minh Nguyen, Nanbo Sun, Daniel C. Alexander, Jiashi Feng, and B.T. Thomas Yeo. Modeling alzheimer's disease progression using deep recurrent neural networks. In *2018 International Workshop on Pattern Recognition in Neuroimaging (PRNI)*, pages 1–4, 2018.

[28] National Institute of Aging. What are the signs of alzheimer's disease ? https://www.nia.nih.gov/health/what-are-signs-alzheimers-disease, May 2017.

[29] National Institute on Aging. Rheumatoid arthritis: Causes, symptoms, treatments. https://www.arthritis.org/diseases/rheumatoid-arthritis, 2020.

[30] World Health Organization. Dementia. https://www.who.int/news-room/fact-sheets/detail/dementia, 2020.

[31] Tiago Pinto and Everton Machado. Is the montreal cognitive assessment (moca) screening superior to the mini-mental state examination (mmse) in the detection of mild cognitive impairment (mci) and alzheimer's disease (ad) in the elderly? *International Psychogeriatrics*, 31:1–14, 11 2018.

[32] Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead, 2019.

[33] Science and Industry Endowment Fund. The australian imaging, biomarker and lifestyle flagship study of ageing (aibl). https://aibl.csiro.au, 2020.

[34] Nitish Srivastava, Elman Mansimov, and Ruslan Salakhudinov. Unsupervised learning of video representations using lstms. In *International Conference on Machine Learning*, pages 843–852. PMLR, 2015.

[35] Erin E. Sundermann, A. Biegon, L. Rubin, R. Lipton, S. Landau, and P. Maki. Does the female advantage in verbal memory contribute to underestimating alzheimer's disease pathology in women versus men? *Journal of Alzheimer's disease : JAD*, 56 3:947–957, 2017.

[36] Jacob W. Vogel, Alexandra L. Young, Neil P. Oxtoby, Ruben Smith, Rik Ossenkoppele, Olof T. Strandberg, Renaud La Joie, Leon M. Aksman, Michel J. Grothe, Yasser Iturria-Medina, and et al. Four distinct trajectories of tau deposition identified in alzheimer's disease. Nature Publishing Group, Apr 2021.

[37] K Yasojima, E.G McGeer, and P.L McGeer. Relationship between beta amyloid peptide generating molecules and neprilysin in alzheimer disease and normal brain. *Brain Research*, 919(1):115–121, 2001.

[38] Hui Zhang, Tu-Bao Ho, and Mao-Song Lin. An evolutionary k-means algorithm for clustering time series data. In *Proceedings of 2004 International Conference on Machine Learning and Cybernetics (IEEE Cat. No.04EX826)*, volume 2, pages 1282–1287 vol.2, 2004.

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**
**CHALMERS UNIVERSITY OF TECHNOLOGY**
**GOTHENBURG UNIVERSITY**
Gothenburg, Sweden
www.chalmers.se

UNIVERSITY OF
GOTHENBURG

CHALMERS
UNIVERSITY OF TECHNOLOGY