Using Contrastive Variational Autoencoders to Disentangle Patterns Specific to Major Depressive Disorder and Diabetes

Yunfei Zhang

Master of Science
School of Informatics
University of Edinburgh
2023

Abstract

Diabetes and major depressive disorder (MDD) have shown associations with brain structure alterations. This project intends to disentangle the diabetes and/or MDD specific patterns based on the 3D T1 structural brain magnetic resonance imaging (MRI) scans. At first, the lightweight Simple Fully Convolutional Network (SFCN) and the linear bias correction (LBC) are applied to estimate the brain age gap (BAG) for each scan. Then BAG is used as a bio-marker to select the scans which have the potential to intensify the diseases-related patterns. Afterwards, a target dataset, which consists of the brain MRI scans from subjects diagnosed with diabetes and/or MDD, and a background dataset, which consists of scans from a healthy control (HC) population, are used as the inputs of contrastive variational auto-encoder (cVAE)-liked models to isolate the desired features. Experiments on different values of the total correction (TC) loss weight γ, cyclical annealing schedules on the Kullback–Leibler (KL) divergence loss weight β and ablation studies on the discriminator and the KL loss were conducted. It is discovered that a malfunctioning discriminator can lead to an ineffective learning in the latent space, and converting a VAE into a deterministic regularized auto-encoder (RAE) might help with the improvement of model performance. The desired patterns tend to cluster in the patent space, but no obvious groupings consistent with the scan types are discovered.

Research Ethics Approval

This project was planned in accordance with the Informatics Research Ethics policy. It did not involve any aspects that required approval from the Informatics Research Ethics committee.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Yunfei Zhang*)

Acknowledgements

Many thanks to my supervisors, Dr. Antonio Vergari and Dr. Filippo Corponi, for their guidance, feedback, encouragement and patience throughout this project. I am also thankful to my colleague, Yaowen Shen, for discussing the problems met in this project.

Table of Contents

Chapter 1

Introduction

1.1 Background

Major Depression Disorder (MDD) and diabetes have been proven to be associated with functional and structural alterations inside a brain [1, 2]. Their pathophysiology may be distributed across many brain regions and circuits. Normally, these kinds of brain abnormalities can be visualized and quantified effectively using magnetic resonance imaging (MRI) [3]. The structure of a brain also changes with a specific pattern while aging, which makes it possible to predict age accurately based on MRI scans [4]. The age that is derived purely from the brain imaging data is called brain age, while the age that is measured from birth to a given date is called chronological age. The two ages are not always consistent with each other. The difference between the brain age and the chronological age is referred to as the brain age gap (BAG), which signifies a deviation from a normal aging trajectory. BAG has become an important bio-marker in clinic usage indicative of potential brain abnormalities, risk of certain diseases and even mortality [5].

1.2 Related Work

- **Brain Age Prediction Models and the SFCN**

Brain age estimation (BAE) methods can be first classified by their input data types, namely sliced-based, which depends on 2D MRI scans, and voxel-based, which depends on 3D MRI scans. Usually voxel-based BAE is considered to be able to utilize more structural connectivity across different parts of the brain, but requires a significantly larger number of parameters and computational resources [4]. BAE can then be classified by its model types, namely traditional machine learning models (such as support vector regression [6], relevance vector regression [7] and Gaussian process regression [8]) and deep learning models (such as CNN [9], VggNet [10] and Transformer based [11] models).

One example of the voxel-based VggNet BAE model is the Simple Fully Convolutional Network (SFCN) [12]. Unlike most of the voxel-based models, it is highly lightweight and has been pre-trained on large-scale data (sample size of 12949) from UK Biobank (UKBB) [13]. It achieves a mean absolute error (MSE) of 2.14 years on UKBB test set, which outperforms both a more complex 3D ResNet-152 [14] and a simpler regression model elastic net [12].

- **Bias Correction on Predicted Age**

The non-Gaussian distribution of the chronological ages tends to cause the problem of "regression dilution" for regression models [15], which inevitably leads to an underfitting of the prediction. The predicted brain age is often systematically biased towards the mean of the cohort, indicating an over-prediction of the age for relatively younger individuals and an under-prediction for elderly individuals [16]. Thus it is needed to apply bias correction techniques [12] to increase the accuracy of brain age estimation.

- **Contrastive Generative Learning and the cVAE**

In standard representation learning, usually its goal is to infer the dominant variations in one dataset of interest. These variations are usually reflected in the embeddings inferred by generative models. Embeddings of similar samples tend to be close to each other, while embeddings from different samples tend to be pushed away [17]. However, the desired features sometimes do not appear as prominent latent factors, which brings the demand for contrastive analysis (CA). CA contains two datasets. Its goal is to disentangle the desired (but probably subtle) patterns enriched in one dataset against the other [18]. CA has been applied to many generative models, such as contrastive principal component analysis (cPCA) [19], probabilistic cPCA (PcPCA) [20] and contrastive variational autoencoder (cVAE) [21].

Compared with cPCA and PcPCA, which are linear models, cVAE has been proven to be able to disentangle highly non-linear features [22]. Normally, cVAE contains an additional discriminator that aims at encouraging the dependence between inferred features and a Kullback–Leibler (KL) divergence [23] loss term to update its parameters. However, under which configurations the discriminator can work successfully and how a failed discriminator can impact the whole model still remain unclear. Additionally, the KL term can sometimes vanishes [24] and impair the performance of the cVAE, but not enough research has been conducted on the techniques that can guarantee effective mitigation of the KL vanishing problem on cVAE.

1.3 Research Objectives

This project aims at identifying patterns specific to MDD and/or diabetes based on the 3D T1 structural brain MRI (T1 sMRI [25]) scans collected from the UK Biobank. To enrich the patterns, the SFCN and bias correction will first be applied to compute a BAG for each sample, which is then used as an indicator to select scans that are likely to intensify the desired patterns. Afterwards, cVAE-liked models will be implemented to disentangle the desired features based on T1 sMRI scans from subjects diagnosed with MDD and/or diabetes against scans from a healthy control (HC) population.

This project intends to explore the answer to the following questions:

▶ RQ1. Previous studies have showed that subjects diagnosed with MDD and/or diabetes tend to have a higher BAG than a HC population [5]. Utilizing the SFCN model and bias correction algorithms, do our results fit with this argument?

▶ RQ2. Under which condition can we reckon that the discriminator in a cVAE is functioning effectively? What is the impact of a discriminator that is functioning, dis-functioning, or even missing?

▶ RQ3. Does the KL vanishing problem take place during the training of the models? What are the possible strategies to alleviate this problem and do they work on cVAE?

▶ RQ4. The desired features generated by the models are expected to cluster into three distinct groups that are consistent with the corresponding scans types (MDD/diabetes/dual) in the latent space. Ideally, the desired features from HC scans should also form cluster(s) distinct from the non-HC scans. Do our results match the expectation?

The rest of this paper is structured as follows: Chapter 2 will first describe the overall pipeline of the project, then explain in detail the principles of SFCN, bias correction and cVAE, and at last describe the visualization and evaluation techniques. Chapter 3 will describe the collection and statistics of the data we used, and then analyze the results of the experiments which are designed to answer the four research questions. RQ1 will be explained and answered in "Brain Age Gap Analysis" part of section 3.2. RQ2 will be discussed and answered in the part "Discussions on the Indicators of A Successfully Trained Discriminator" of section 3.3.2, in section 3.3.3 and in section 3.4. RQ3 will be answered in the part "Analysis on The KL Vanishing Problem" in section 3.3.4 and in section 3.4. To explore QR4, visualizations of the latent space are provided in each experiment. QR4 will also be answered at the end of section 3.4. Finally, Chapter 4 will summarize the main findings and provide suggestions for future improvements.

Chapter 2

Methods

This chapter is intended for explaining in detail the overall pipeline of the program, the usage of the SFCN model, the application of the bias correction algorithm, the implementation of the cVAE model, and the techniques to evaluate the results of the project.

2.1 The Overall Pipeline

The project can be roughly divided into three stages: (1) Data collection, (2) Brain age estimation and (3) Disentangling disease specific patterns.

- **Data Collection**

  In the first stage, it's intended to collect four types of T1 brain sMRI scans, namely "HC scans" – the scans from a health control (HC) population (subjects not diagnosed with major depression disorder (MDD) or diabetes), "MDD scans" – scans from subjects diagnosed with MDD but not diabetes, "diabetes scans" – scans from subjects diagnosed with diabetes but not MDD, and "dual scans" – scans from subjects diagnosed with both MDD and diabetes. All the scans should be labeled with the chronological age of the subject at the time the scan was taken.

- **Brain Age Estimation**

  In the second stage, it's intended to first predict the brain age of the sMRI scans collected in the first stage using the SFCN model. Then based on the predicted age and the chronological age of the HC scans, a linear bias correction algorithm will be fitted. The fitted algorithm will be applied to all the scans to compute their unbiased brain age based on their previous predicted age. Subtracting the chronological age from the unbiased age, we label each scan with its brain age gap (BAG).

- **Disentangling Disease Specific Patterns**

  In the third stage, it's planned to firstly intensify the disease-specific patterns by filtering the three types of non-HC scans (the MDD scans, diabetes scans and the dual scans) with a BAG larger than a particular threshold (usually set to zero) to form a target dataset. Then we sample the HC scans so that the chronological age distribution of the sampled HC scans can match the chronological age distribution of the target dataset. The selected HC scans then form a background dataset. Secondly, utilizing the two datasets, we plan to build and train cVAE-liked models that can infer high dimensional latent features which represent the disease-specific patterns.

2.2 Predict Brain age using Simple Fully Convoluted Network

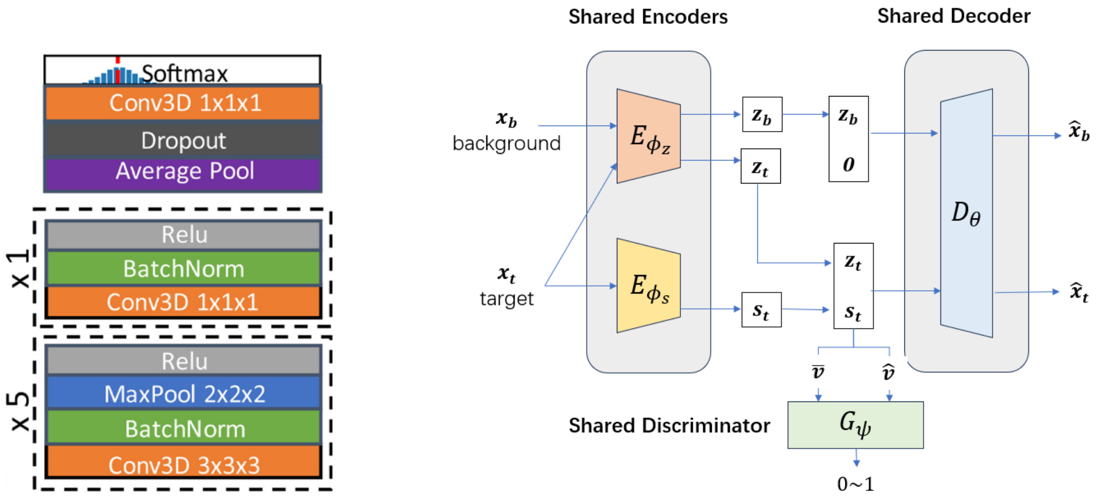
As stated in Section 1.3, we capitalize on the pre-trained Simple Fully Convoluted Network (SFCN) built by [12] to infer brain age based on T1 brain sMRI data.

- **Lightweight Model Architecture**

  SFCN is a lightweight deep neural network. It is based on VGGNet [26] with a fully convolutional structure. As displayed in Figure 2.1a, it contains 7 blocks in total, with each block having only one convolutional layer before a MaxPool layer and removing all the fully connected layers, which greatly reduces the number of trainable parameters and increases the flexibility for accommodating various input sizes [27]. The meaning of the model structure can be interpreted as: The first five blocks serve as a feature-map extractor of the input data; The sixth block further increases the nonlinearity of the previous extraction process; Then the last block converts the extracted features to age probability distribution. The relatively small model size makes the SFCN requires less memory and computation time while inference and less prone to overfitting.

- **Training and Predicting**

  The input of SFCN should be batched T1-brain-sMRI scans, with each sample being a single-channel tensor of shape $(160, 192, 160)$. For each sample, the output of SFCN is a tensor $\mathbf{q}$ of shape $(40, 1)$, representing the predicted probabilities where the subject's age falls into one-year age intervals between 42 to 82 [12]. During training, the model will minimize a Kullback–Leibler (KL) divergence [23] loss function $\mathbf{L}_{SFCN}$ between the predicted age probability distribution (i.e. $\mathbf{q}$) and the true age probability distribution. The true age probability distribution is defined as a Gaussian distribution



(a) SFCN Structure [12]

(b) cVAE structure [21]

Figure 2.1: Model Architecture

with a mean of its true chronological age and a variance of 1. Hence,the loss function is given by [1]:

$$\mathbf{L}_{SFCN} = \mathrm{KL}\left(\mathbf{q}||\mathcal{N}(\mathrm{true\_age}, 1)\right) \tag{2.1}$$

Denoting the i_{th} element of $\mathbf{q}$ as q_i, the bin center of i_{th} age interval as age_i, the predicted age y of the SFCN is computed as the weighted average of each age bin:

$$y = \sum_{i}^{40} \mathrm{q}_i \cdot \mathrm{age}_i \tag{2.2}$$

2.3 Linear Bias Correction on the Predicted Brain Age

As stated in Chapter 1, here we adopt the linear bias correction (LBC) proposed by [12]. Given a set of samples labeled by chronological age, defining $\mathbf{w}$ to be their chronological age and $\mathbf{y}$ to be their predicted brain age, we fit a least square linear regression model on $\mathbf{y} = a \cdot \mathbf{w} + b$ to obtain an optimum slope a and interval b. Defining ε to be an extremely small value to avoid dividing by zero, the corrected unbiased predicted age $\hat{\mathbf{y}}$ is then computed by $\hat{\mathbf{y}} = (\mathbf{y} - b)/(a + \varepsilon)$. The same a, b can be applied to bias correction on other unlabeled samples.

[1] Here we denote a KL divergence of any two distributions p and q as $\mathrm{KL}(p||q)$), a Gaussian distribution with mean mean a and variance b as $\mathcal{N}(a,b)$

2.4 Isolate Salient Features via Contrastive Generative Learning

As stated in Chapter 1, we adopt the basic architecture of the contrastive variational auto-encoder (cVAE) from [21], as shown in Figure 2.1b, which is composed of two shared decoders, one encoder and one discriminator. However, the cVAE by [21] is designed for 2D inputs and has most of its layers fully connected. Thus it is only suitable for data each with a relatively small volume. However, brain sMRI scans are 3D inputs with huge volume – around size 5×10^6 if flattened. Hence, it's necessary to modify the cVAE to adopt the 3D inputs, and keep the model size down to prevent over-parameterization at the same time. The principles and detailed structure of our 3D convolutional cVAE will be explained in the rest of this chapter. The number of parameters of our model is managed to be around 1×10^7 at last. It is about $\frac{3}{5}$ size of the cVAE designed by [22], which successfully identifies autism spectrum disorder (ASD) related patterns also based on brain MRI scans.

- **Problem Settings**

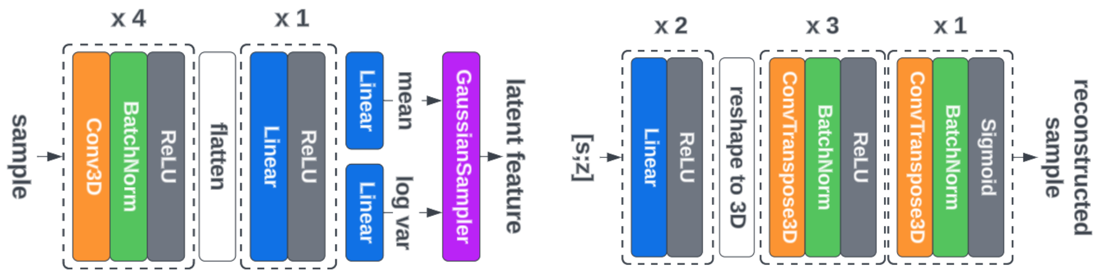
  There should be two (unpaired) datasets of observed samples, namely the target datasets $\{\mathbf{x}_t^{(i)}\}_{i=1}^{N_t}$ and the background datasets $\{\mathbf{x}_b^{(j)}\}_{j=1}^{N_b}$. The features that are shared between the two datasets are referred to as the irrelevant features $\mathbf{z}$. The features that are enriched in the target dataset relative to the other are referred to as the salient features $\mathbf{s}$, which are exactly the disease-specific patterns we intend to find in this project.

- **Assumptions**

  Any observed sample $\mathbf{x}$, no matter in the target or the background dataset, is assumed to be independent and identically distributed (i.i.d.). It is also assumed that any sample can be used to infer, and can be reconstructed from, one irrelevant and one salient feature. (The salient features for the background samples are fixed as zeros.) Both of the two latent features are assumed to be independently drawn from an anisotropic Gaussian prior: $\mathbf{s} \sim p(\mathbf{s}) = \mathcal{N}(\mathbf{0}, \mathbb{I})$, $\mathbf{z} \sim p(\mathbf{z}) = \mathcal{N}(\mathbf{0}, \mathbb{I})$.

- **Encoders**

  The process $q_s(\mathbf{s}|\mathbf{x})$ of inferring one salient feature from a sample is simplified into a Gaussian distribution $\mathcal{N}(\mathbf{s}; \mu_s, \sigma_s^2 \mathbb{I})$. The prediction of the mean μ_s and the log-variance $\log \sigma_s^2$ of the feature is conducted by a variational encoder $\mathcal{E}_{\phi_s}$ parameterized by ϕ_s. Similarly, the process $q_z(\mathbf{z}|\mathbf{x})$ of inferring one irrelevant feature from a sample is also simplified into a Gaussian distribution $\mathcal{N}(\mathbf{z}; \mu_z, \sigma_z^2 \mathbb{I})$. The prediction of the mean μ_z



(a) encoder

(b) decoder

Figure 2.2: cVAE Internal Structures

and the log-variance $\log\sigma_z^2$ of the feature is conducted by the other variational encoder $\mathcal{E}_{\phi_z}$ parameterized by ϕ_z. $\mathcal{E}_{\phi_s}$ and $\mathcal{E}_{\phi_z}$ are shared between the two datasets and work independently. Thus, given an observed sample $\mathbf{x}^{(k)}$, the latent feature inference process includes sampling:

$$\begin{aligned} \mathbf{s}^{(k)} &= \mathcal{E}_{\phi_s}(\mathbf{x}^{(k)}) \sim q_s(\mathbf{s}|\mathbf{x}^{(k)};\phi_s) = \mathcal{N}(\mathbf{s};\mu_s^{(k)},\sigma_s^{2(k)}\mathbb{I};\phi_s) \\ \mathbf{z}^{(k)} &= \mathcal{E}_{\phi_z}(\mathbf{x}^{(k)}) \sim q_z(\mathbf{z}|\mathbf{x}^{(k)};\phi_z) = \mathcal{N}(\mathbf{z};\mu_z^{(k)},\sigma_z^{2(k)}\mathbb{I};\phi_z), \end{aligned} \tag{2.3}$$

where $\mathbf{s}^{(k)}$ is omitted for background samples.

The two encoders have the same architecture as shown in Figure 2.2a. They take in single channel images each of shape $(160,192,160)$. The four 3D convolution blocks have filter size of $[32,64,128,256]$, kernel size of 3 and strides length of 2. They will convert each input to a feature map of shape $(10,12,10)$, which will then be flattened and mapped to an intermediate feature of size $(1,128)$ by a fully connected (FC) block. The encoder will then use another two FC layers to approximate the mean and log variance of the latent feature respectively, and finally output the salient or the irrelevant latent feature of size $(1,d)$ by a Gaussian sampler layer.

- **Decoder**

The reconstruction process of a sample from its salient and irrelevant features is denoted as an unknown conditional distribution $f(\mathbf{x}|\mathbf{s},\mathbf{z})$, which is modeled by a decoder $\mathcal{D}_\theta$ parameterized by θ. Thus given $\mathbf{s}^{(k)},\mathbf{z}^{(k)}$ of any observed sample indexed by k, the reconstructed sample is drawn from:

$$\hat{\mathbf{x}}^{(\mathbf{k})} = \mathcal{D}_\theta([\mathbf{s}^{(k)};\mathbf{z}^{(k)}]) \sim f(\mathbf{x}|\mathbf{s}^{(k)},\mathbf{z}^{(k)};\theta), \tag{2.4}$$

where $s^{(k)} = \mathbf{0}$ for backgroud data.

The structure of the decoder $\mathcal{D}_\theta$ is displayed in Figure 2.2b. It takes in the concatenation of a salient and an irrelevant feature of a sample, and then its first two FC blocks will first convert the input to an intermediate size of $(1,128)$ and then to the flattened size of $(1,256\times 10\times 12\times 10)$. The feature will then be reshaped back to 3D with 256 channels. Afterwards, the four 3D convolutional transpose blocks, which have filter sizes of $[128,64,32,1]$, kernel sizes of 3 and stride lengths of 2, will map the feature back to a reconstructed sample same as the input shape $(160,192,160)$. Since the last layer is the Sigmoid activation layer, all the output elements range between 0 to 1.

- **Updating the encoders and the decoder**

The encoders and the decoder are optimized by maxing the evidence lower bound (ELBO [28], $\mathcal{L}_t$ or $\mathcal{L}_b$) of the log-likelihood of each input sample. In terms of a target sample $\mathbf{x}_t^{(i)}$, its ELBO $\mathcal{L}_t$ is derived as [2]:

$$\begin{aligned}\log \mathrm{P}(\mathbf{x}_t^{(i)};\phi_s,\phi_z,\theta) &\geq \mathcal{L}_t(\mathbf{x}_t^{(i)};\phi_s,\phi_z,\theta)\\ &= \mathrm{E}_{q_s,q_z}\left[\log[f(\mathbf{x}_t^{(i)}|\mathbf{s}_t^{(i)},\mathbf{z}_t^{(i)};\theta)]\right] - \mathrm{KL}\left(q_s(\mathbf{s}_t|\mathbf{x}_t^{(i)};\phi_s)||\ p(\mathbf{s})\right)\\ &\quad - \mathrm{KL}\left(q_z(\mathbf{z}_t|\mathbf{x}_t^{(i)};\phi_z)||\ p(\mathbf{z})\right)\\ &= \mathrm{E}_{q_s,q_z}\left[\log[f(\mathbf{x}_t^{(i)}|\mathbf{s}_t^{(i)},\mathbf{z}_t^{(i)};\theta)]\right] - \mathrm{KL}\left(\mathcal{N}(\mathbf{s}_t;\mu_s^{(i)},\sigma_s^{2(i)}\mathbb{I};\phi_s)||\ \mathcal{N}(\mathbf{s};\mathbf{0},\mathbb{I})\right)\\ &\quad - \mathrm{KL}\left(\mathcal{N}(\mathbf{z}_t;\mu_z^{(i)},\sigma_z^{2(i)}\mathbb{I};\phi_z)||\ \mathcal{N}(\mathbf{z};\mathbf{0},\mathbb{I})\right)\end{aligned} \tag{2.5}$$

Similarly, the ELBO $\mathcal{L}_b$ in terms of a background sample $\mathbf{x}_b^{(j)}$ is derived as:

$$\begin{aligned}\log \mathrm{P}(\mathbf{x}_b^{(j)};\phi_z,\theta) &\geq \mathcal{L}_b(\mathbf{x}_b^{(j)};\phi_z,\theta)\\ &= \mathrm{E}_{q_z}\left[\log[f(\mathbf{x}_b^{(j)}|\mathbf{0},\mathbf{z}_b^{(j)};\theta)]\right] - \mathrm{KL}\left(\mathcal{N}(\mathbf{z}_b;\mu_z^{(j)},\sigma_z^{2(j)}\mathbb{I};\phi_z)||\ \mathcal{N}(\mathbf{z};\mathbf{0},\mathbb{I})\right)\end{aligned} \tag{2.6}$$

Defining the negative sum of the two Expectation terms above as a reconstruction loss $\mathbf{L}_{REC}$ [3] , which can be computed as the mean squared error (per voxel) between the input sample and its reconstructed sample [29]; Defining the sum of the three KL divergence terms above as a KL loss $\mathbf{L}_{KL}$ [4]. Maximizing $\mathcal{L}_t$ and $\mathcal{L}_b$ will minimize both $\mathbf{L}_{REC}$ and $\mathbf{L}_{KL}$, and thus updating the parameters in the encoders and the decoder accordingly.

- **Total Correlation**

[2] Here we denote $\mathbf{s}_t, \mathbf{z}_t$ to be the salient and the irrelevant features inferred from target samples, and $\mathbf{z}_b$ to be the irrelevant feature inferred from the background samples.

[3] $\mathbf{L}_{REC} = -\left(\mathrm{E}_{q_s,q_z}\left[\log[f(\mathbf{x}_t^{(i)}|\mathbf{s}_t^{(i)},\mathbf{z}_t^{(i)};\theta)]\right] + \mathrm{E}_{q_z}\left[\log[f(\mathbf{x}_b^{(j)}|\mathbf{0},\mathbf{z}_b^{(j)};\theta)]\right]\right)$

[4] $\mathbf{L}_{KL} = \mathrm{KL}\left(\mathcal{N}(\mathbf{s}_t;\mu_s^{(i)},\sigma_s^{2(i)}\mathbb{I};\phi_s)||\ \mathcal{N}(\mathbf{s};\mathbf{0},\mathbb{I})\right) + \mathrm{KL}\left(\mathcal{N}(\mathbf{z}_t;\mu_z^{(i)},\sigma_z^{2(i)}\mathbb{I};\phi_z)||\ \mathcal{N}(\mathbf{z};\mathbf{0},\mathbb{I})\right) + \mathrm{KL}\left(\mathcal{N}(\mathbf{z}_t;\mu_z^{(j)},\sigma_z^{2(j)}\mathbb{I};\phi_z)||\ \mathcal{N}(\mathbf{z};\mathbf{0},\mathbb{I})\right)$

It is discovered that encouraging the independence between the salient and the irrelevant features can improve the performance of cVAE [21]. The total correlation (TC) term between $\mathbf{s}, \mathbf{z}$ is defined as the negative KL divergence between the joint conditional probability distribution q_{joint} of the two latent features and the product q_{prod} of their own conditional probability distributions [21]:

$$\begin{gathered} \text{TC} = -\text{KL}\left(q_{\text{joint}} || \, q_{\text{prod}}\right), \text{with} \\ q_{\text{joint}} = q_{s,z}(\mathbf{s}, \mathbf{z}|\mathbf{x}^{(k)}; \phi_s, \phi_z) \;\; \text{and} \;\; q_{\text{prod}} = q_s(\mathbf{s}|\mathbf{x}^{(k)}; \phi_s) \times q_z(\mathbf{z}|\mathbf{x}^{(k)}; \phi_z). \end{gathered} \tag{2.7}$$

In this case, $\text{TC} = 0$ only when $\mathbf{s}, \mathbf{z}$ are independent.

Here we only apply the TC term on the target dataset. The concatenation $\bar{\mathbf{v}}^{(i)}$ of the two latent features inferred from the same target sample are considered to be drawn from the joint probability: $\bar{\mathbf{v}}^{(i)} = [\mathbf{s}_t^{(i)}; \mathbf{z}_t^{(i)}] \sim q_{\text{joint}}$. The concatenation $\hat{\mathbf{v}}^{(i)}$ of the two latent features inferred from different target samples are considered to be drawn from the probability product: $\hat{\mathbf{v}}^{(i)} = [\mathbf{s}_t^{(i)}; \mathbf{z}_t^{(k)}] \sim q_{\text{prod}}$, $k \neq i$. In practice, a batch of $\bar{\mathbf{v}}$ is formed by horizontally stacking a batch of $\mathbf{s}_t$ and the same batch of $\mathbf{z}_t$. A batch of $\hat{\mathbf{v}}$ is formed by horizontally stacking a batch of $\mathbf{s}_t$ and the same batch of $\mathbf{z}_t$, but with the position of the first half batch and the second half batch of $\mathbf{z}_t$ switched, as displayed in Table 2.1.

$\mathbf{s}_t$	$\mathbf{z}_t$	$\bar{\mathbf{v}}$		$\hat{\mathbf{v}}$	
$\mathbf{s}_1$	$\mathbf{z}_1$	$\mathbf{s}_1$	$\mathbf{z}_1$	$\mathbf{s}_1$	$\mathbf{z}_2$
$\mathbf{s}_2$	$\mathbf{z}_2$	$\mathbf{s}_2$	$\mathbf{z}_2$	$\mathbf{s}_2$	$\mathbf{z}_1$

Table 2.1: Form $\bar{\mathbf{v}}$ and $\hat{\mathbf{v}}$

- **Discriminator**

Given a concatenation $\mathbf{v}$ of any $\mathbf{s}$ and any $\mathbf{z}$, we build a discriminator $\mathcal{G}_\psi$ parameterized by ψ to estimate the probability of the concatenation that is drawn from the joint distribution: $\mathcal{G}_\psi(\mathbf{v}) = p(\mathbf{v} \sim q_{\text{joint}})$. The discriminator only has three layers: a FC layer that takes in the concatenated feature and maps it to a 1D feature, a batch norm layer, and finally a Sigmoid layer that outputs the probability.

The TC loss $\mathbf{L}_{TC}$ is defined on $\bar{\mathbf{v}}$ as $\mathbf{L}_{TC}(\bar{\mathbf{v}};\, \psi, \phi_s, \phi_z) = \log\left[\mathcal{G}_\psi(\bar{\mathbf{v}}) / \left(1 - \mathcal{G}_\psi(\bar{\mathbf{v}})\right)\right]$, and the discriminator loss $\mathbf{L}_\mathcal{G}$ is defined on both $\bar{\mathbf{v}}$ and $\hat{\mathbf{v}}$ as $\mathbf{L}_\mathcal{G}(\bar{\mathbf{v}}, \hat{\mathbf{v}};\, \psi, \phi_s, \phi_z) = -\log\left[\mathcal{G}_\psi(\bar{\mathbf{v}}) \times \left(1 - \mathcal{G}_\psi(\hat{\mathbf{v}})\right)\right]$ [21]. Minimizing $\mathbf{L}_{TC}$ will force $\mathcal{G}_\psi(\bar{\mathbf{v}}) \longrightarrow 0$, whereas minimizing $\mathbf{L}_\mathcal{G}$ will force $\mathcal{G}_\psi(\bar{\mathbf{v}}) \longrightarrow 1$ and $\mathcal{G}_\psi(\hat{\mathbf{v}}) \longrightarrow 0$. Hence, these two loss functions tend to adversarially update the parameters of our cVAE. Ideally, they will force two probabilities (or scores) to approach one half: $\mathcal{G}_\psi(\bar{\mathbf{v}}) = \mathcal{G}_\psi(\hat{\mathbf{v}}) \longrightarrow \frac{1}{2}$, leading to $\mathbf{L}_{TC} \longrightarrow \log 1 = 0$ and $\mathbf{L}_{\mathcal{G}} \longrightarrow \log \frac{1}{4} \approx 0.6021$. This means that the concatenated features drawn from q_{joint} and q_{prod} are too similar to each other for the discriminator to distinguish, which indicates that the q_{joint} has become almost the same as q_{prod}. Therefore, the salient and the irrelevant features have been encouraged to be independent of each other successfully.

- **Overall Training Procedures**

In conclusion, $\mathcal{E}_{\phi_s}$, $\mathcal{E}_{\phi_z}$, $\mathcal{D}_\theta$ and $\mathcal{G}_\psi$ are updated simultaneously by an Adam optimizer [30] during training using a total loss $\mathbf{L}_{cVAE}$, which is a weighted sum of the four loss functions mentioned above: $\mathbf{L}_{cVAE} = \alpha \times \mathbf{L}_{REC} + \beta \times \mathbf{L}_{KL} + \gamma \times \mathbf{L}_{TC} + \mathbf{L}_{\mathcal{G}}$. The detailed training loop of the cVAE within one epoch is specified in Algorithm 1. The training process will be early stopped when the validation loss is smaller than a particular number Δ for continuous N_p epochs. The total number of hyper-parameters that are needed to be specified by users is 8, namely the dimension d of the two latent features, the three weights α, β, γ of the loss functions, the batch size B, the learning rate λ, and the early stop criteria Δ and N_p.

Algorithm 1 cVAE Training Loop

```
input: Training datasets $\{\mathbf{x}_t^{(i)}\}_{i=1}^{N_t}$ and $\{\mathbf{x}_b^{(j)}\}_{j=1}^{N_b}$; Hyper-parameters $d, \alpha, \beta, \gamma, B, \lambda$
for every batch of data $\{\mathbf{x}_t^{(i)}\}_{i=1}^{B}$ and $\{\mathbf{x}_b^{(j)}\}_{j=1}^{B}$ do
    sample $\mathbf{s}_t^{(i)} = \mathcal{E}_{\phi_s}(\mathbf{x}_t^{(i)})$, $\mathbf{z}_t^{(i)} = \mathcal{E}_{\phi_z}(\mathbf{x}_t^{(i)})$, $\mathbf{z}_b^{(j)} = \mathcal{E}_{\phi_z}(\mathbf{x}_b^{(j)})$, $\forall i, j \in [1, .., B]$
    reconstruct $\hat{\mathbf{x}}_t^{(i)} = \mathcal{D}_\theta([\mathbf{s}_t^{(i)}; \mathbf{z}_t^{(i)}])$, $\hat{\mathbf{x}}_b^{(j)} = \mathcal{D}_\theta([\mathbf{0}; \mathbf{z}_t^{(j)}])$, $\forall i, j \in [1, .., B]$
    form $\bar{\mathbf{v}}^{(i)}, \hat{\mathbf{v}}^{(i)}$ and predict $\mathcal{G}_\psi(\bar{\mathbf{v}}^{(i)})$, $\mathcal{G}_\psi(\hat{\mathbf{v}}^{(i)})$, $\forall i \in [1, .., B]$
    obtain batched loss $\mathbf{L}_{cVAE} = \frac{1}{B}\sum_{i,j=1}^{B}[\alpha \cdot \mathbf{L}_{REC} + \beta \cdot \mathbf{L}_{KL} + \gamma \cdot \mathbf{L}_{TC} + \mathbf{L}_{\mathcal{G}}]$
    update parameters $\phi_s, \phi_z, \theta, \psi$ by $-\lambda \cdot \nabla \mathbf{L}_{cVAE}$ accordingly
end for
```

However, cVAE is a relatively new model and few previous works on a similar domain can be found. The range of the four main hyper-parameters d, α, β, γ is not limited to a relatively small scale, and causes enormous possible combinations of them. d could range from 2 to hundreds; α could range from a voxel level (e.g. 1) to a sample level (e.g. $160 \times 192 \times 160$); β and γ could range from 10^{-3} to 10^3. Therefore, the cVAE is reckoned extremely hard to be trained to properly disentangle the disease-specific patterns which can form clusters corresponding to the scan types.

2.5 Visualization and Evaluation Techniques

The performance of the project is estimated by the consistency between the clustering of the salient features and corresponding non-HC scan types ("MDD", "diabetes", "dual"). It can be visualized by UMAP [31] and measured by SS [32] and average NMI [33] [5].

- **UMAP**

  To visualize the distribution of the salient features, Uniform Manifold Approximation and Projection (UMAP) can be applied as a dimension reduction technique that maps the salient features to a 2D space. A 2D plot of salient features inferred by the validation set will be generated at each epoch during training a model, and a 2D plot of salient features on a test set will be generated when evaluating the model.

- **SS**

  A mean silhouette score (SS) of all the salient features indicates how well the features are matched to the cluster of their own type against other clusters. The score ranges from -1 to 1. A mean SS close to 1 means features are far away from other clusters, and thus are more likely to be clustered into the correct group. Normally, a larger mean SS implies better performance. However, if a mean SS is kept at zero for a long time during training, it is probably caused by an almost random distribution of the features generated by the model. In this case, a model with zero mean SS does not necessarily outperforms a model with a negative mean SS. The mean SS on the validation set of the target dataset will be tracked at each epoch during training. SS will also be computed while testing the model.

- **average NMI**

  We first treat the inferred salient features as unlabeled data, and use Gaussian mixture models (GMM) [34] to cluster them with the number of Gaussians specified as 3. Then GMM will assign a cluster label for each latent feature, and the normalized mutual information (NMI) between the assigned labels and the true labels (corresponding scan types) will be computed. GMM is not a deterministic process, so different labels and NMI can be obtained at each run of GMM. Here we will run GMM for 100 times and use the average NMI to measure our models. The NMI of two random variables estimates the mutual dependence between the two variables. It ranges from 0 to 1. Thus, an average NMI close to 1 means clustering of the desired features perfectly correlated

[5] As explained in Section 2.4, the salient encoder is not trained on the background dataset. Thus SS and NMI will be computed only on the target dataset. However, it is also helpful to know whether salient features inferred by the HC scans also form a distinct cluster, thus during visualization the salient features inferred from the background dataset are also plotted.

with the scan types and indicates excellent cVAE performance. The average NMI will be only computed while testing the model.

Chapter 3

Experiments

3.1 Data Collection and Analysis

- **Dataset Format**

  The T1 brain sMRI scans collected in this project are categorized as instance 2 of data field 20252 [35] in UKBB [13] "imaging/raw/t1_structural_nifti_20252" dataset. The scans use the Montreal Neurological Institute and Hospital (MNI) coordinate system [36], and are stored in files using the Neuroimaging Informatics Technology Initiative (NIfTI) format [37].

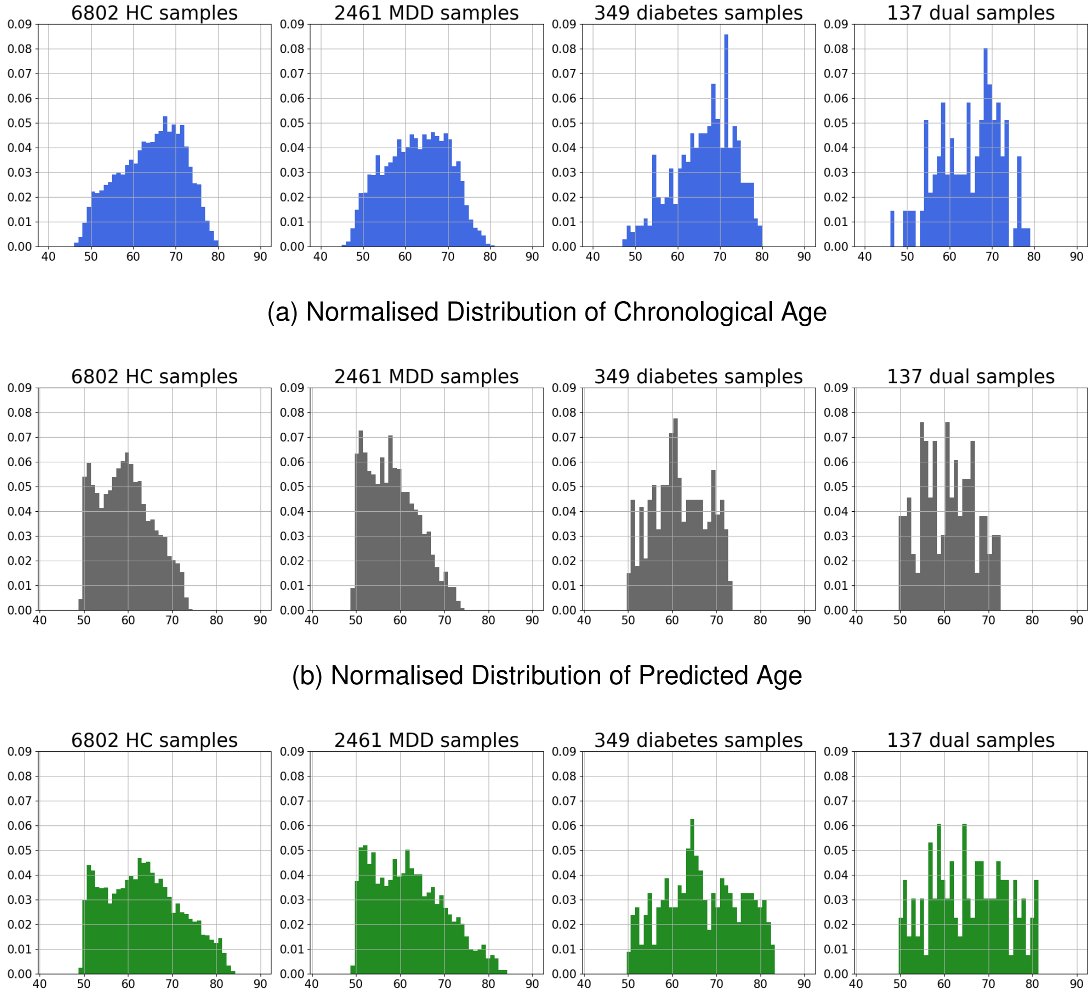
- **Data Collection**

  The chronological age of the scans is labeled via UKBB data field $f.21003.2.0$, i.e. instance 2 of "Age when attended assessment centre" [38]. The MDD status of the scans is labeled via UKBB data field $f.20126.0.0$, i.e. instance 0 of "Bipolar and major depression status" [39]. Subjects marked as 0 are considered to be not diagnosed with MDD, while subjects marked as 3 or 4 or 5 are considered to be diagnosed with MDD [40]. The diabetes status of the scans is labeled via $f.2976.2.0$, i.e. instance 2 of "Age diabetes diagnosed" [41]. Subjects marked with a *NaN* value are considered to be not diagnosed with diabetes, while subjects marked with any other non-*NaN* value are considered to be diagnosed with diabetes.

  As defined in section 2.1, the collected scans are then labeled by their type accordingly. As a result, 6802 HC samples, 2461 MDD samples, 349 diabetes samples and 137 dual samples are obtained.

- **Chronological Age Analysis**

  The normalized distributions of the chronological age of the four scan types are



(a) Normalised Distribution of Chronological Age

(b) Normalised Distribution of Predicted Age

(c) Normalised Distribution of Unbiased Age

Figure 3.1: Different Age Distributions of the Four Scan Types

displayed in Figure 3.1a. As can be seen, for all scan types, the chronological age ranges within 45 $\sim$ 82. The HC scans and MDD scans share similar distributions where most of the subjects fall into a chronological age range of 55 $\sim$ 70, while diabetes scans and dual scans share similar distributions where most of the subjects fall into a chronological age range of 65 $\sim$ 75.

3.2 Brain Age Estimation

- **Pre-processing**

The array of a scan directly extracted from the MNI-NIfTI files has a shape of $(182, 218, 182)$ and mostly has a range of $(-50, 2500)$. To make the scans suitable for

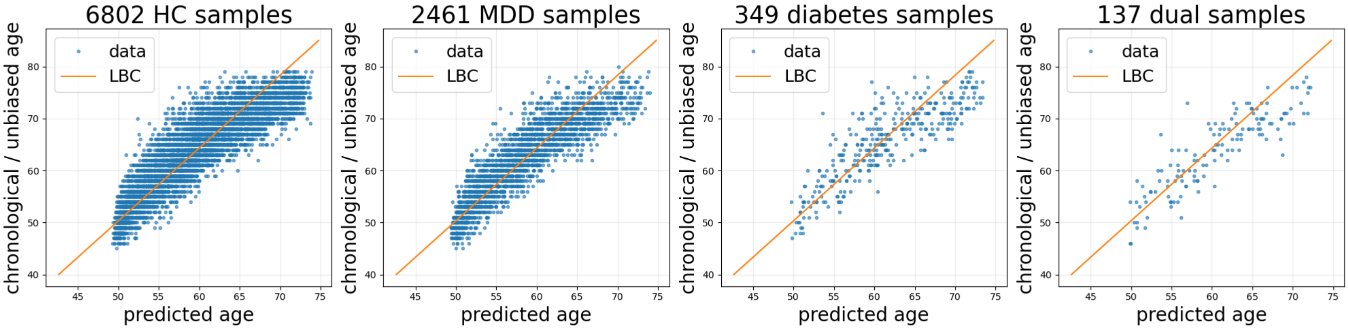


Figure 3.2: Bias Correction Results: The blue dots are data points of (predicted age, chronological age). The orange lines are LBC results of (predicted age, unbiased age).

the SFCN model, we first shrink the range of data within about $(-1, 15)$ by dividing each scan by its mean value, and then crop the scans around their center to remove skulls and reshape them to $(160, 192, 160)$. The pre-processed scans then serve as the input of the SFCN model to predict the corresponding brain age.

- **Predicted Age Analysis**

  The normalized distributions of the predicted brain age of the four scan type are displayed in Figure 3.1b. As can be seen, for all scan types the predicted age ranges within 48 $\sim$ 75. Most of the HC scans and the MDD scans have a predicted age centered around 50 $\sim$ 65, while most of the diabetes scans and dual scans have a predicted age centered around 55 $\sim$ 65.

- **Unbiased Age Analysis**

  After fitting the LBC algorithm on the HC samples as explained in Section 2.3, an optimum slope $a = 0.714$ and interval $b = 14.062$ are obtained. The LBC results are displayed in Figure 3.2, and the normalized distribution of the resulted unbiased age of the four scan types are displayed in Figure 3.1c.

  From both figures, we can see that, for all scan types, the unbiased age has a range of 48 $\sim$ 85. Compared with the range 48 $\sim$ 75 of predicted age, it can be seen that the LBC algorithm extends and shifts the whole predicted age distribution rightwards.

  Moreover, based on the HC samples in Figure 3.2, it can be discovered that the LBC algorithm does help mitigate the error of brain age prediction. Taking a predicted age of 65 on HC scans as an example, the true (chronological) age of the corresponding scans has a range of 60 $\sim$ 80. The LBC shifts the predicted age to the unbiased age of 71.342, which makes the inferred brain age closer to the middle of the true age, and thus increases the accuracy of brain age estimation.

Scan Type	MAG	mean	std	min	25%	50%	75%	max
HC	2.9692	0.0000	3.7884	-16.3517	-2.3987	-0.0464	2.4339	14.5412
MDD	2.8510	-0.0279	3.6214	-14.0778	-2.4696	-0.0012	2.3124	11.8603
diabetes	3.7047	0.9715	4.5829	-15.6052	-1.8281	1.0229	3.8486	11.7400
dual	3.5493	1.4175	4.1759	-12.9674	-1.4059	1.8467	3.8629	13.5201

Table 3.1: Brain Age Gap Statistics ("MAG": mean absolute brain age gap; "std": standard deviation.)

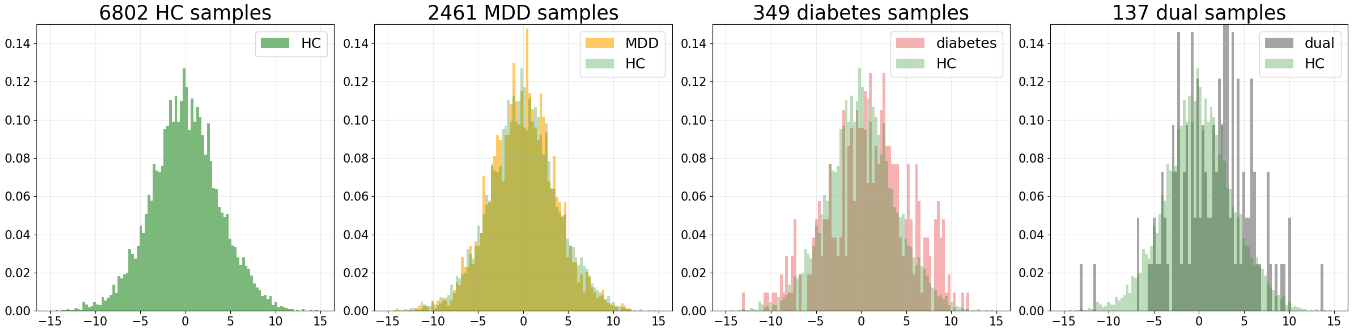


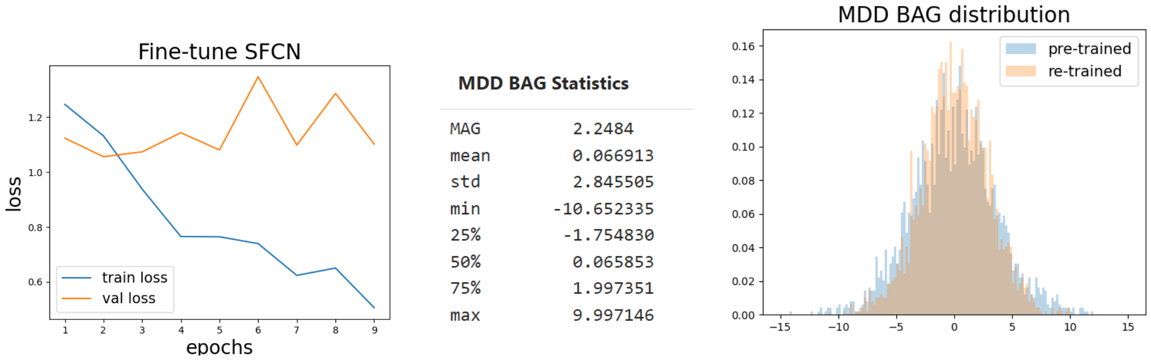
Figure 3.3: Normalized BAG Distribution against HC Type

- **Brain Age Gap Analysis**

Then each sample is marked by its brain age gap (BAG), which is computed by subtracting the chronological age from the unbiased brain age. The statistics of the BAG of the four scan types are displayed in Table 3.1, and the normalized BAG distributions of each scan type against HC type are displayed in Figure 3.3.

As displayed, for all scan types the shape of the BAG distribution is similar to a Gaussian distribution curve. Moreover, compared with the BAG of the HC type which is centered around -0.0464, the BAGs of the diabetes type and the dual type are centered around 1.0229 and 1.8467 respectively, which validates that subjects diagnosed with diabetes tend to have a higher brain age gap than healthy people. This can also be justified by that the mean absolute value of brain age gap (MAG) of both the diabetes and the due type, which are 3.7047 and 3.5493 respectively, are higher than the MAG of the HC type, which is 2.9692.

However, it can be discovered in Figure 3.3 that the MDD type almost shares the same normalized BAG distribution as the HC type. Additionally, as showed in Table 3.1, although the center of the BAG of MDD type is slightly larger than that of the HC type by 0.00452, the MAG of the MDD type is smaller than that of the HC type to a



MDD BAG Statistics

```
MAG        2.2484
mean       0.066913
std        2.845505
min      -10.652335
25%       -1.754830
50%        0.065853
75%        1.997351
max        9.997146
```

(a) (b) (c)

Figure 3.4: Loss Curve of Re-trained SFCN and new Brain Age GAP of MDD samples

larger extent by 0.1182. Both findings above contradicts the hypothesis and previous research which claim that subjects diagnosed with MDD tend to have higher BAG than HC population.

- **Re-train SFCN**

The original SFCN might not be pre-trained on only healthy subjects, which might be a reason for the similar distributions of BAG on the MDD and the HC types in our project. Hence, with the intention of obtaining a larger BAG on the MDD scans, we fine-tune the SFCN model by initializing its parameter by the pre-trained SFCN and keep training it on the HC samples (with 20% of them to be the validation set).

The loss curve is displayed in Figure 3.4a. The validation loss curve is quite wiggly and reaches its minimum at epoch 2 (with the early stop patience set as 7). Though there's not much updating on the model parameters, we take the model at epoch 2 as our new SFCN model, and use it to predict the brain age of the HC and the MDD types. After fitting a new LBC algorithm on the HC samples, the new BAG on the MDD types is obtained. Its statistics and normalized distribution against the original MDD BAG are displayed in Figure 3.4b and 3.4c.

As can be seen, the distribution of BAG of MDD samples by the re-trained model is very similar to that by the pre-trained model. Compared with the previous MDD MAG of 2.8510, the new MDD MAG decreases to 2.2484. Hence, the re-trained SFCN fails to recognize the MDD samples with larger BAG. Additionally, it is also discovered that, compared with the mean absolute error of 1.6772 between the chronological age and the predicted age of the HC samples, the new LBC algorithm increases HC BAG (i.e. the mean absolute error between the chronological age and the unbiased age) to

1.7165. This means applying LBC to the results by the new model tends to decrease the prediction accuracy. Therefore, the re-trained SFCN is not considered a proper model for the project. The rest of the project keeps utilizing the results obtained by the pre-trained SFCN model.

3.3 Disentangle Disease-Specific Features

3.3.1 Preparation

- **Dataset Collection and Pre-processing**

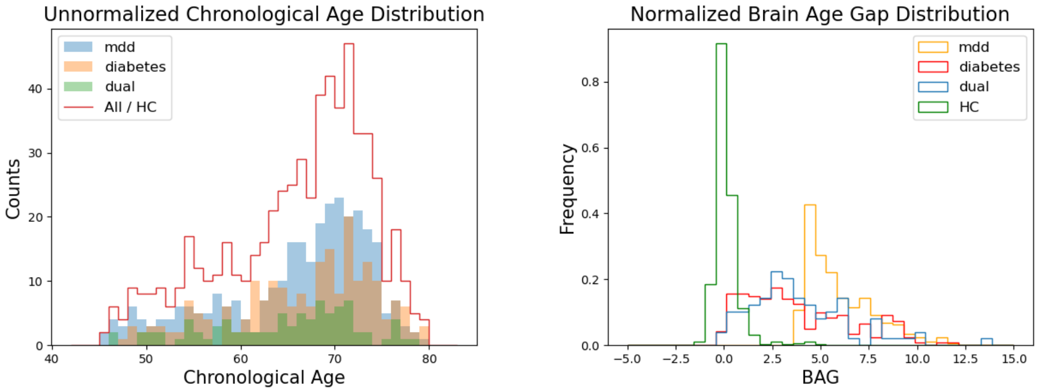
To intensify the disease-specific patterns, a filter of BAG over 4 is added to the MDD scans, and a filter of BAG over 0 is added to the diabetes and the dual scans. As a result, 307 MDD samples, 211 diabetes samples and 86 dual samples are selected to form a target dataset of sample size 604. Then 604 HC samples are selected to form a background dataset where its chronological age distribution matches the target dataset. The unnormalized chronological age distribution and the normalized BAG distribution of the 4 selected scan types are displayed in Figure 3.5. As displayed, most of the samples have a chronological age of $60 \sim 78$. The HC BAG mostly falls into $-1 \sim 2$, the MDD BAG mostly falls into $4 \sim 7$, and the diabetes BAG and the dual BAG mostly fall into $0 \sim 6$.

During training, 80% samples from both datasets are randomly selected as the training sets, and the rest 20% are used as the validation sets. Each sample will be first cropped into a shape of $(160, 192, 160)$. Then, to match the output range of the Sigmoid activation function at the last layer of the decoder, a min-max scaler is fitted on the training set and applied to all the samples to normalize the input into a range of $[0, 1]$.

- **Hyper-parameter tuning**

At the very beginning of the project, to shrink down the size of cVAE, a large kernel size (e.g. 11), a small latent feature dimension (e.g. $d = 2$) and few batch normalization layers were used. This had led to two problems during training: a) the plotted salient features were easily aligned to a row and at last overlapped into a single point after several epochs; b) gradients vanish easily during backpropagation, which causes the loss becoming *NaN* and the training being stopped. The first problem was solved by using a smaller kernel size in the encoders and a larger latent dimension. Kernel size of 3 and latent dim $d = 32$ [1] is applied after considering the cVAE for ASD paper [22].

[1] [22] has input shape of 64^3 and uses $d = 16$. We have larger input size and thus decide to choose d



(a)

(b)

Figure 3.5: cVAE Dataset

It is also discovered that the KL loss $\mathbf{L}_{KL}$ tends to explode at the second batch in the first epoch during training, and rapidly shrink down to a normal level after about three epochs. The value that KL explodes to can be influenced by the learning rate λ. After experimenting with different random seeds, it is discovered that λ of $0.0001, 0.0003, 0.0005, 0.0007$ are likely to cause exploding $\mathbf{L}_{KL}$ at a magnitude of $10^3, 10^8, 10^9, 10^{22}$, and $\lambda > 0.001$ can often result in errors in backpropagation and stop training. Considering both the exploding magnitude and the speed of training, $\lambda = 0.0003$ is chosen to be used in our experiments.

Due to the time limit of the project, to have a quicker convergence on the loss value during training, the batch size is set to a relatively small value of 10, i.e. $B = 10$. The training will be early stopped if there's no increase in the validation set for 7 epochs, i.e. $\Delta = 0, N_p = 7$.

Then the only hyper-parameters that remain to be decided are the three loss weights α, β, γ. With the intention of scaling down the range of possible values of them, we first tried to conduct a grid search on them using a training set and a validation set of sizes 80 and 20 on both datasets. Since the value of the three parameters contributes to the total loss, models with smaller loss weights are more likely to have a lower total loss. Hence, it's not suitable to judge the model performance by the best validation loss. Instead, SS at the epoch of the best validation loss is used. However, it is discovered that the training results on the same set of α, β, γ are highly unstable even after blocking all the randomness. The best validation loss might be reached within the first 5 epochs, or after 50 epochs. The corresponding SS may keep at 0 or decrease to -0.08 [2]. Therefore, here

twice of their size.

[2] Normally, for most majority of the tested settings, SS ranges from -0.1 to 0.

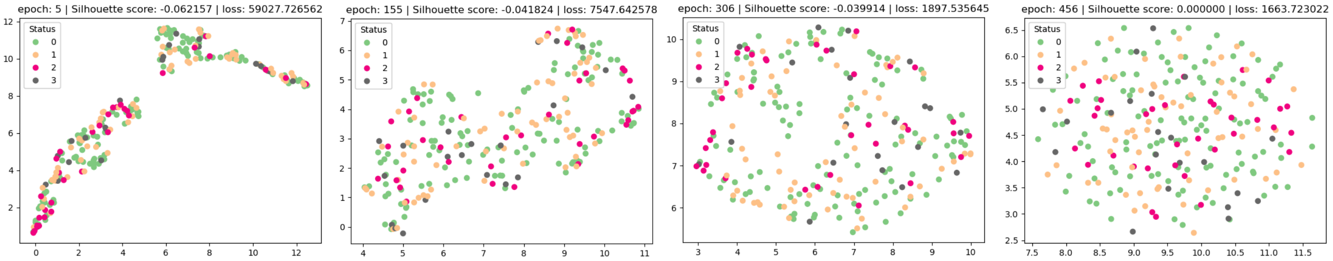


Figure 3.6: Baseline Model – Latent Space of Salient Features at epoch 5, 155, 306 and 456 (0-HC, 1-MDD, 2-diabetes, 3-dual). (SS in the plots of latent space considers HC scans, while other SS in the text only considers target samples.)

grid search by SS is not considered a robust hyper-parameter tuning strategy. Finally, we decided to adopt the value of the loss weights used in [22] as a baseline model ($\alpha = 250000$ [3], $\beta = 1$, $\gamma = 100$), and afterwards manually adjust the value of them to improve model performance. [4]

3.3.2 Baseline Model

- **Overall Training Process**

The training loss, the validation loss and the mean SS on the salient features inferred from the validation set of the target dataset [5] during training is plotted in Figure 3.7 (1). The training of the baseline model is stopped manually at epoch 456 after the validation SS keeps at 0 for more than 100 epochs, although the validation loss is still decreasing at a slow rate as displayed in Figure 3.7 (2). The model is picked at the last epoch for it has the smallest validation loss so far.

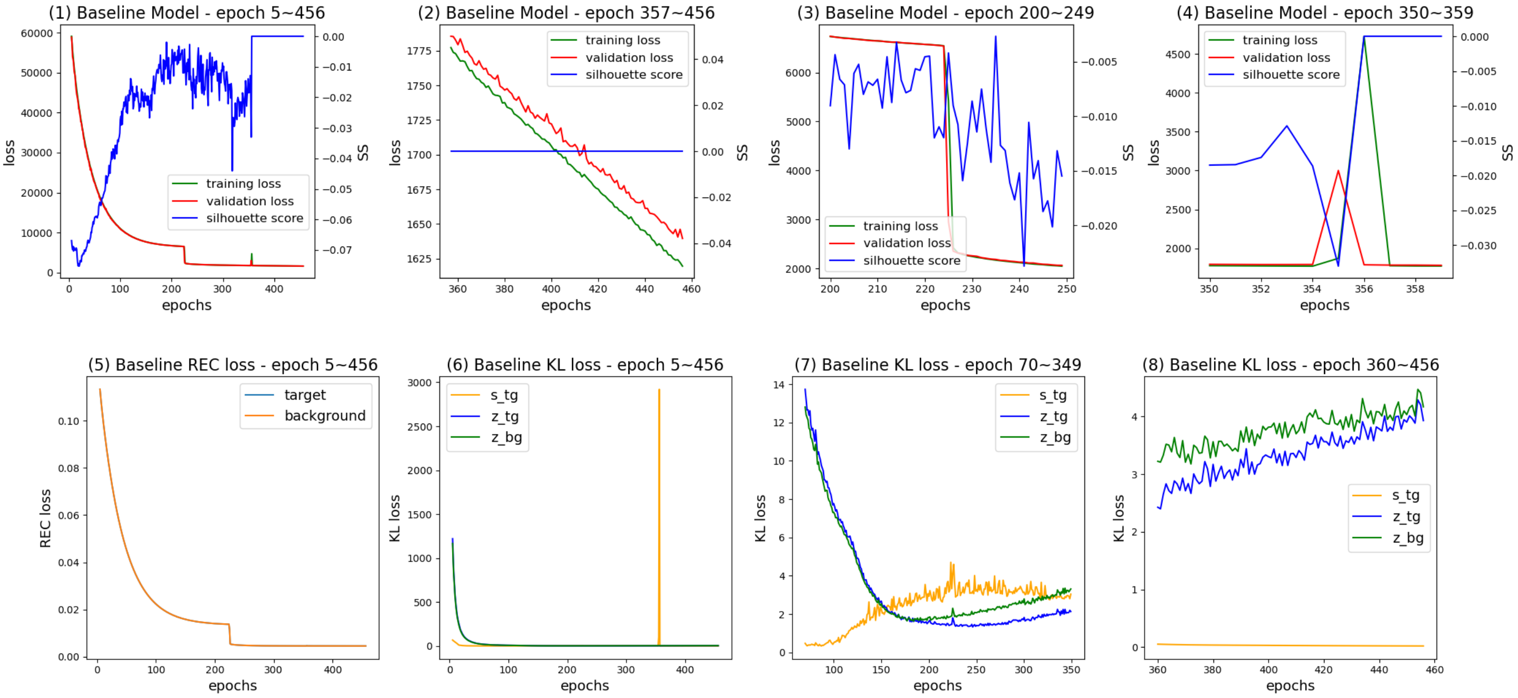
The evolution of the salient latent space of the 4 scan types from the validation set is displayed in Figure 3.6 at each one-third of the training process. There's not much grouping by the scan types that can be discovered in the plots. The first three plots do contain several clusters with SS<0, while the points in the last plot (by the picked model) seem to be randomly scattered with SS=0.

- **Two Sudden Changes in the Loss Curves**

[3] [22] uses $\alpha = 64^3$ to make reconstruction loss match their input dimension, here we adjust α to 250000 to it at about $\frac{1}{20}$ of our input dimension.

[4] For all other hyper-parameters, if not explicitly stated in below experiments, are fixed as the value described above.

[5] For the rest of the paper, we would refer to "the mean SS on the salient features inferred from the validation set of the target dataset" as simply "the validation SS" of the training process.


(1) Baseline Model - epoch 5~456

(2) Baseline Model - epoch 357~456

(3) Baseline Model - epoch 200~249

(4) Baseline Model - epoch 350~359

(5) Baseline REC loss - epoch 5~456

(6) Baseline KL loss - epoch 5~456

(7) Baseline KL loss - epoch 70~349

(8) Baseline KL loss - epoch 360~456

Figure 3.7: Baseline Model – Loss Curves (Except for the validation loss and the validation SS, all the losses are computed on training set if not specified otherwise.)

It can be seen from Figure 3.7 (1), there are two sudden changes in the loss curves at around epoch 225 and epoch 356 respectively.

The first sudden change is zoomed in and plotted in Figure 3.7 (3), where both the training and the validation loss decrease from about 6500 to about 2500 within 2 epochs. It is caused by the sudden dive in the reconstruction loss. As displayed in Figure 3.7 (5), the reconstruction losses of the target and the background data almost overlapped. They decrease rapidly in the first 100 epochs from 0.1275 to 0.0223, then decrease slowly in the next 100 epochs to 0.0140. At epoch 225 they suddenly dive into 0.0054, and then level at about 0.0046 afterwards.

The second sudden change is zoomed in and plotted in Figure 3.7 (4), where the training and the validation loss jump to about 4700 and 300 respectively from about 1800, and fall back to about 1780 within 3 epochs. It is caused by the sudden jump of the KL loss on the salient feature to about 2900, as shown in Figure 3.7 (6). Figure 3.7 (7,8) show that before the jump, KL loss on the three latent features all reach about 3.5. After the jump, the KL loss on the irrelevant features gradually goes up to about 4.2 at last, while the KL loss on the salient features levels at about 0.03 (but still with a very slowly decreasing trend).

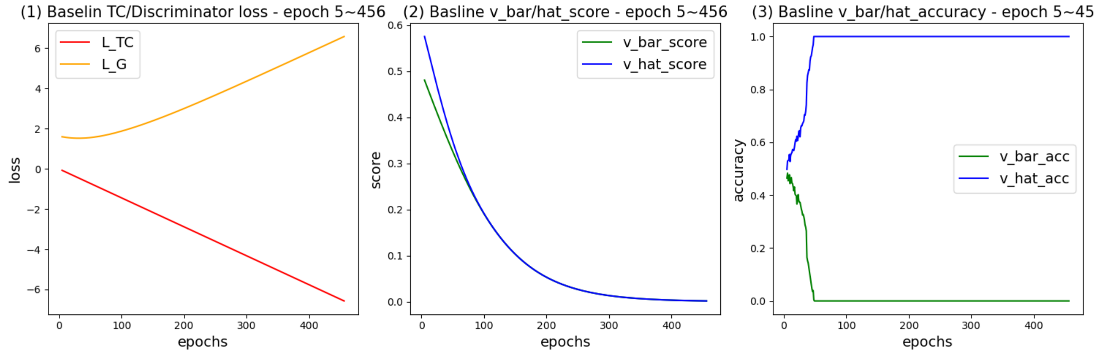


Figure 3.8: Baseline Model – Discriminator Performance

- **Analysis on TC Loss and Discriminator Loss**

The TC loss $\mathbf{L}_{TC}$ and the discriminator loss $\mathbf{L}_{\mathcal{G}}$ are plotted in Figure 3.8 (1). $\mathbf{L}_{TC}$ decreases almost linearly from -0.0070 to -6.5704, and $\mathbf{L}_{\mathcal{G}}$ increase gradually from 1.6178 to 6.5731. However, as stated in section 2.4, ideally $\mathbf{L}_{TC} \longrightarrow 0$ and $\mathbf{L}_{\mathcal{G}} \longrightarrow 0.6021$.Thus, the value of the two losses deviates greatly from our expectation.

To figure out the reason for the abnormal $\mathbf{L}_{TC}$ and $\mathbf{L}_{\mathcal{G}}$, it's needed to monitor the internal changes of the discriminator, namely the scores (probability) it assigns to the inputs and its accuracy. The average scores, $\mathcal{G}_{\psi}(\bar{\mathbf{v}})$ and $\mathcal{G}_{\psi}(\hat{\mathbf{v}})$, the discriminator assigns to $\bar{\mathbf{v}}$ and $\hat{\mathbf{v}}$ at each epoch are recorded in Figure 3.8 (2). As plotted, after epoch 50 the discriminator tends to assign a score less than 0.5 to any of its input no matter its $\bar{\mathbf{v}}$ or $\hat{\mathbf{v}}$, and after epoch 250 it learns to assign a score close to 0 to any input. The accuracy [6] of the discriminator judging whether a concatenated feature is drawn from q_{joint} or q_{prod} is recorded in Figure 3.8 (3). According to the definition of the discriminator in section 2.4, it will classify an input as being drawn from q_{prod} if its score is less than 0.5, otherwise as being drawn from q_{joint}. Hence, the scores described above tend to lead to the absolute majority (or even all) of $\bar{\mathbf{v}}$ and $\hat{\mathbf{v}}$ being classified as being drawn from q_{prod}, which is consistent with the accuracy plotted in Figure 3.8 (3).

As stated in section 2.4, a successfully trained discriminator should obtain both $\mathcal{G}_{\psi}(\bar{\mathbf{v}})$ and $\mathcal{G}_{\psi}(\hat{\mathbf{v}})$ close to $\frac{1}{2}$, rather than 0 as the baseline model. Additionally, $\mathbf{L}_{TC}$ pushes $\mathcal{G}_{\psi}(\bar{\mathbf{v}})$ to 0, whereas $\mathbf{L}_{\mathcal{G}}$ pushes $\mathcal{G}_{\psi}(\bar{\mathbf{v}})$ to 1. Therefore, a mean $\mathcal{G}_{\psi}(\bar{\mathbf{v}})$ too close to 0 can be resulted from a $\mathbf{L}_{TC}$ which is much stronger than $\mathbf{L}_{\mathcal{G}}$. Hence, it is necessary to scale down the weight γ of the TC loss, so that the discriminator can balance the scores at around $\frac{1}{2}$ and encourage the independence between the salient features $\mathbf{s}_t$ and

[6] It is worth notifying that here we're not talking about the overall accuracy on all the inputs of the discriminator, but the accuracy on $\bar{\mathbf{v}}$ and $\hat{\mathbf{v}}$ separately.

the irrelevant features $\mathbf{z}_t$.

- **Discussions on the Indicators of A Successfully Trained Discriminator**

  It's significant to define under which condition can we judge a discriminator to be trained successfully, i.e. a discriminator that can force the $\mathbf{s}_t$ and the $\mathbf{z}_t$ to be independent of each other. That the mean $\mathcal{G}_\psi(\bar{\mathbf{v}})$ and the mean $\mathcal{G}_\psi(\hat{\mathbf{v}})$ approach $\frac{1}{2}$ and that the $\bar{\mathbf{v}}$ accuracy and the $\hat{\mathbf{v}}$ accuracy approach $\frac{1}{2}$ are both a sufficient condition for an effective discriminator. This is because either of the two conditions means the discriminator tends to classify about half $\bar{\mathbf{v}}$ (and $\hat{\mathbf{v}}$) as being drawn from q_{prod} and the other half as being drawn from q_{joint}, which indicates high similarity between q_{prod} and q_{joint}.

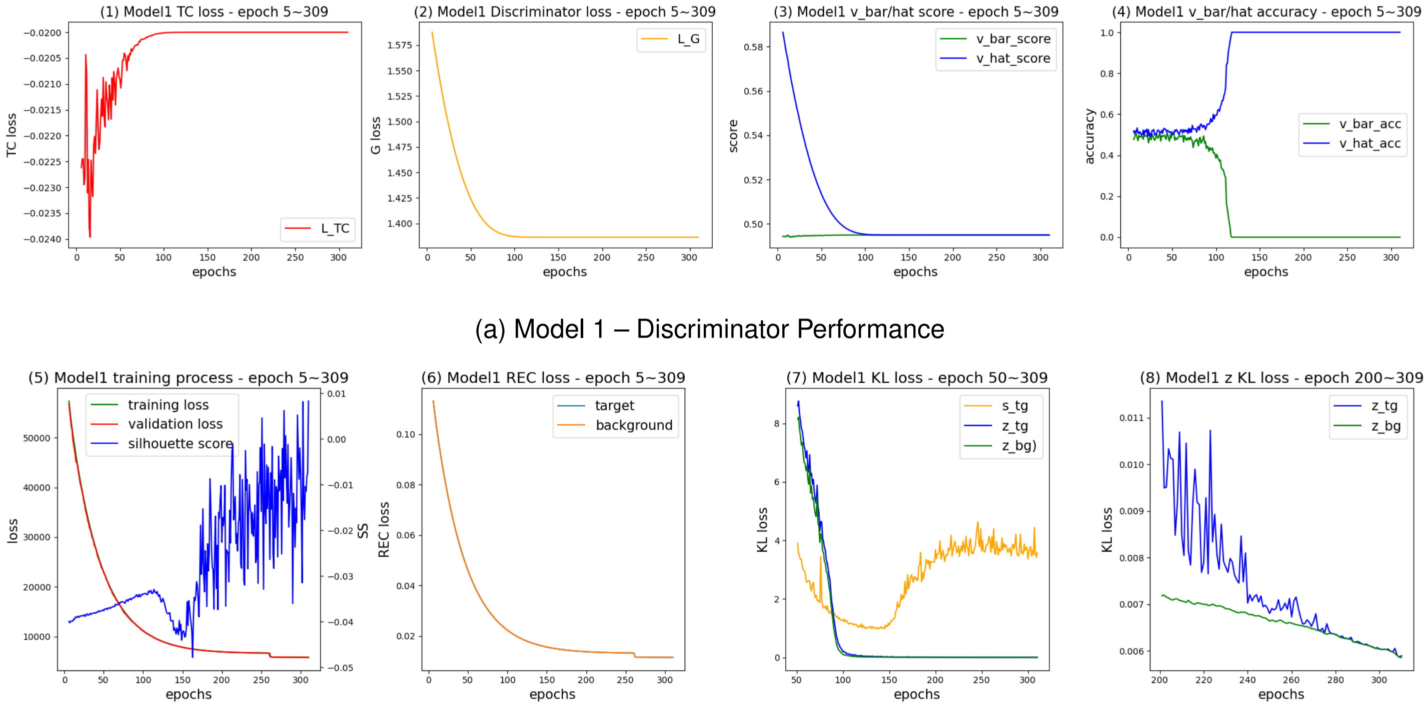
  Additionally, a mean $\mathcal{G}_\psi(\bar{\mathbf{v}})$ or a mean $\mathcal{G}_\psi(\hat{\mathbf{v}})$ far away from $\frac{1}{2}$ should be a sign of a malfunctioned discriminator. This is because the situation can be caused by either a high dissimilarity between q_{prod} and q_{joint} so that the discriminator can tell the difference between them, or the extremely large (or small) parameters of the discriminator so that the discriminator assigns 1 (or 0) to any of its input. However, a $\bar{\mathbf{v}}$ accuracy or a $\hat{\mathbf{v}}$ accuracy far away from $\frac{1}{2}$ does not necessarily lead to a failed discriminator. Considering the case when all $\mathcal{G}_\psi(\bar{\mathbf{v}}) = 0.51$ and all $\mathcal{G}_\psi(\hat{\mathbf{v}}) = 0.49$, both the scores are around 0.5 which means the independence has been reached. However, in this case, both accuracy are 1. Hence the two accuracy of $\frac{1}{2}$ is not a necessary condition of the independence in practice.

  Therefore, the indicator of a successfully trained discriminator should be either that both $\bar{\mathbf{v}}$ and $\hat{\mathbf{v}}$ score approach 0.5 or that both $\bar{\mathbf{v}}$ and $\hat{\mathbf{v}}$ accuracy approach 0.5.

3.3.3 Experiments on Smaller Value of γ and Ablation Study on Discriminator

- **Model 1:** $\alpha = 250000, \beta = 1, \gamma = 0.01$

  As suggested in the last section, we keep the value of α and β the same as the baseline model, and conduct experiments on smaller values (0.01, 0.1, 1, 10) of γ. It is discovered that $\gamma = 0.01$ can successfully lead to both $\bar{\mathbf{v}}$ score and $\hat{\mathbf{v}}$ score approaching 0.5. The curves of the discriminator-related indicators are plotted in Figure 3.9a. As displayed, after epoch 150, the TC loss levels at -0.0200, the discriminator loss levels at 1.3864, both the $\bar{\mathbf{v}}$ score and the $\hat{\mathbf{v}}$ score level at 0.4950, the $\bar{\mathbf{v}}$ accuracy and the $\hat{\mathbf{v}}$ accuracy level at 0 and 1 respectively. Hence, the discriminator under this setting is considered successfully trained.



(a) Model 1 – Discriminator Performance

(b) Model 1 – Loss Curves

Figure 3.9: Model 1 – Training process

The training of the cVAE under this setting is early stopped at epoch 309 due to no increase in the validation loss for 7 epochs. Model 1 is picked at epoch 302 for it has the smallest validation loss so far. The curves of the training loss, the validation loss and the validation SS are plotted in Figure 3.9b (5). As displayed, the validation SS fluctuates widely after epoch 180. Compared to the baseline model, one breakthrough of Model 1 should be its SS does not levels at 0, but reach some positive values at several epochs, e.g.SS=0.0062 at epoch 278, SS=0.0051 at epoch 295 and SS=0.0082 at epoch 309. Similar to the baseline model, there's also a small dive in the training and validation

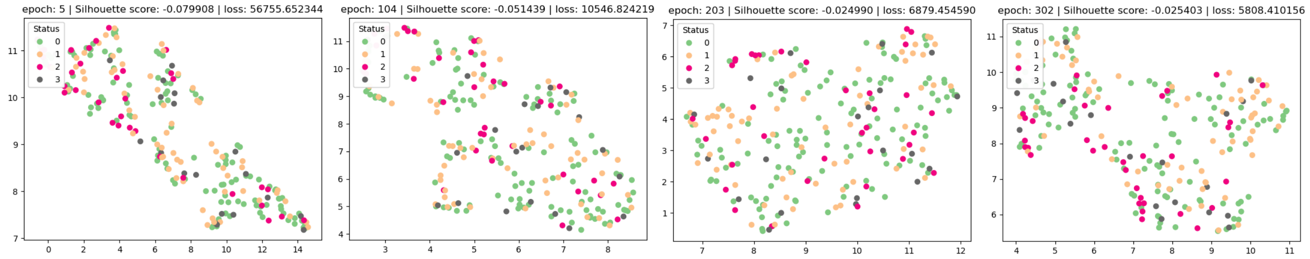


Figure 3.10: Model 1 – Latent Space of Salient Features at epoch 5, 104, 203 and 302; (0-HC, 1-MDD, 2-diabetes, 3-dual)

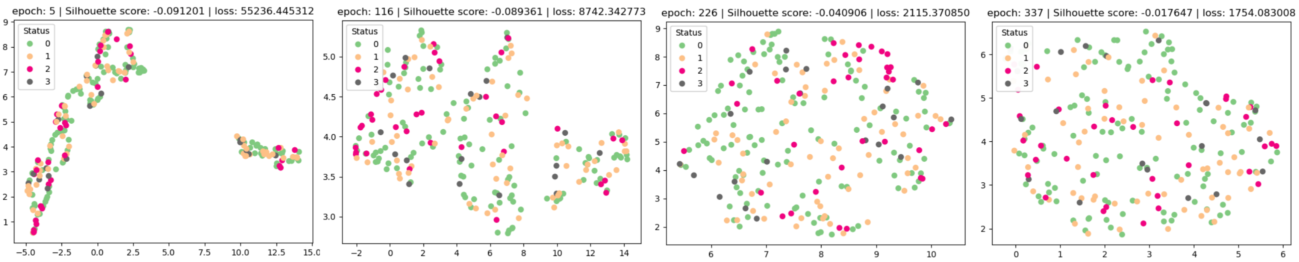


Figure 3.11: Model 2 – Latent Space of Salient Features at epoch 5, 116, 226 and 337; (0-HC, 1-MDD, 2-diabetes, 3-dual)

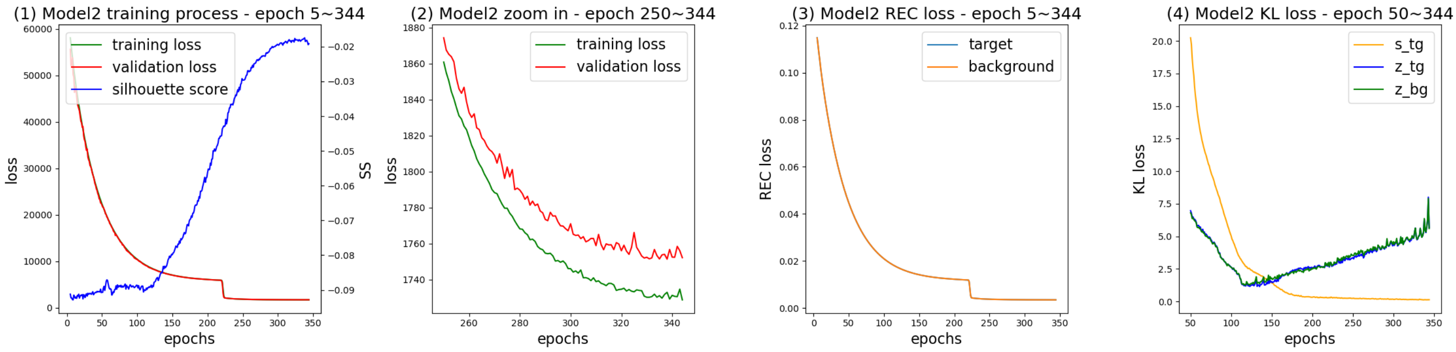


Figure 3.12: Model 2 – Loss Curves

loss curve, which is also caused by the change in the reconstruction loss as showed in Figure 3.9b (6). The evolution of the latent space of the salient features is visualized in Figure 3.10 at every one-third of the training process. The situation is still not as expected since no obvious clusters by scan types can be discovered, but compared with the baseline model, at least the points are not randomly distributed at later epochs.

- **Model 2:** $\alpha = 250000, \beta = 1$**, with Discriminator Removed**

The baseline model can be reckoned as a sample model with a failed discriminator, and Model 1 can be reckoned as another sample model with an effective discriminator. To further study how the TC loss and the discriminator loss can influence the performance of the cVAE, an ablation study on a cVAE without a discriminator is worth conducting. All other settings are kept the same as Model 1.

The training of the cVAE under this setting is early stopped at epoch 344 due to no validation loss increase for 7 epochs. Model 2 is picked at epoch 337 for it has the smallest validation loss so far. The evolution of the latent space of the salient features are visualized in Figure 3.11 at every one-third of the training process. The training loss curves is depicted in Figure 3.12. It can be discovered that compared with the baseline and Model 1, the curve of the validation SS of Model 2 becomes much more smooth.

Models	KL loss on $\mathbf{s}_t$			Avg KL loss on $(\mathbf{z}_t, \mathbf{z}_b)$		
	min	max	trend	min	max	trend
Baseline $(357 \sim 456)$	**0.0576**	**0.0216**	$\searrow$	2.6111	4.3797	$\nearrow$
Model 1 $(210 \sim 309)$	3.3605	4.6359	$\leadsto$	**0.0059**	**0.0089**	$\searrow$
Model 2 $(245 \sim 344)$	0.1222	0.2591	$\searrow$	3.3001	7.9542	$\nearrow$
Model 3 $(315 \sim 414)$	1.0311	$1.8960\mathrm{e} \times 10^{19}$	$\searrow\nearrow\searrow$	**0.0060**	**0.0296**	$\searrow$

Table 3.2: KL loss on the Last 100 Epochs (texts in bold marks vanishing KL; $\searrow$ – decreasing, $\leadsto$ – wiggly but mainly flat, $\nearrow$ – increasing)

The SS first fluctuated around -0.0892 at the first 130 epochs, then gradually climbs to around -0.0184 at epoch 300, and fluctuate around this value afterwards. Unfortunately, the upward trend of SS stops after epoch 300, and the SS fails to reach any positive value [7]. Compared with the baseline model, whose SS curve has a range of $[-0.0753, 0]$, and Model 1, whose SS curve has a range of $[-0.0478, 0.0083]$, Model 2 has a lower range on SS during training, which is $[-0.0964, -0.0174]$. Therefore, it might be concluded that the discriminator tends to bring a wiggly SS curve (which might indicate more drastic changes in the latent space) and higher SS while training.

3.3.4 Experiments on β annealing and Ablation Study on KL loss

- **Analysis on The KL Vanishing Problem**

While training VAEs, the KL loss term can sometimes become extremely small and then vanish, which makes the inferred latent features match their prior distribution closely. This problem can limit the diversity of the latent space, which makes the whole model overly focus on reconstructing the input sample and prevents the encoder from learning meaningful and well-structured latent representations.

As depicted in Figure 3.7 (7,8), Figure 3.9b (7,8) and Figure 3.12 (4), for all the three models obtained above, the KL loss of either the salient features $(\mathbf{s}_t)$ or the irrelevant features $(\mathbf{z}_t, \mathbf{z}_b)$ tends to approach 0 at the end of the training process. Table 3.2 summarizes the range and the trend of the KL loss at the last 100 epochs of the three models. The KL vanishing problem is considered to take place if the minimum value is below 0.1 and the trend is not increasing ($\nearrow$). The happening of the problem has been

[7] Even though we have tried increasing the early stop patience to 15 to continue the training of the model for about another 50 epochs, no obvious increase in SS can be discovered. For the consistency of experiments, here we only report the results with early stop patience set as 7.

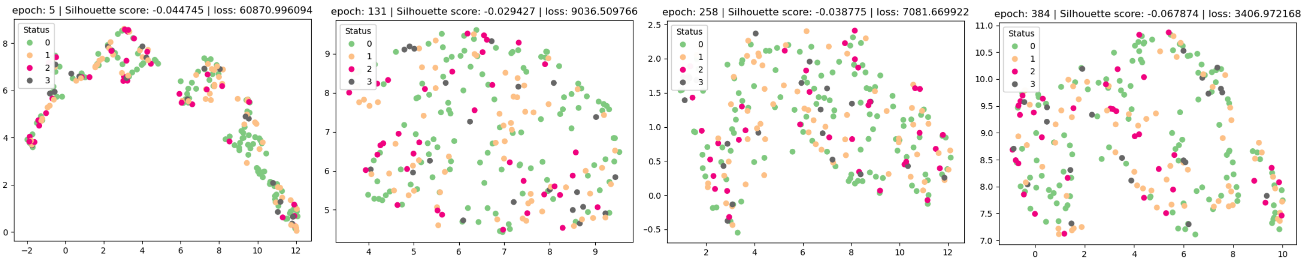


Figure 3.13: Model 3 – Latent Space of Salient Features at epoch 5, 131, 251 and 384 (0-HC, 1-MDD, 2-diabetes, 3-dual)

marked in bold in the table. As displayed, the KL loss on the salient features of the baseline model and the KL loss on the irrelevant features of Model 1 vanishes. The KL loss on the salient features of Model 2, though not reckoned as vanishing, also becomes relatively small (only slightly higher than 0.1). Therefore, it is important to mitigate the KL vanishing problem during training.

Two strategies are adopted to alleviate the problem. The first is the cyclical annealing schedule [24] on the KL loss weight β. The second is to conduct the ablation test by avoiding the use of KL loss via transforming the cVAE into a deterministic contrastive regularized auto-encoder (cRAE) [42].

- **Model 3:** $\alpha = 250000, \beta = 1, \gamma = 0.01$**, with Cyclical Annealing Schedule on** β

The annealing schedule usually sets β to a small value (e.g. 0) at the first several epochs to let the reconstruction loss dominate the total loss, and then progressively increase the KL β. With a cyclical schedule, it's intended to repeat the procedure of increasing β for multiple times. However, as mentioned in section 3.3.1, the KL loss tends to explode at the very beginning, so we set $\beta = 1$ for the first l epochs and then start the cyclical annealing schedule. Let C denote the length of a cycle, β_t denote the value of KL loss weight at epoch t, the schedule is then formed by:

$$\beta_t = \begin{cases} 1, & t < l \text{ or } (t > l \text{ and } m \geq \frac{C}{2}) \\ m \cdot \frac{2\beta}{C}, & t > l \text{ and } m < \frac{C}{2} \end{cases}, \quad m = (t - l) \bmod C \tag{3.1}$$

It should be noticed that β_t contributes to the total loss $\mathbf{L}_{cVAE}$. As β_t increases, $\mathbf{L}_{cVAE}$ tends to increase as well. Hence, an early stop patience N_p smaller than C is not suitable with this schedule, as the increase in the validation loss may be caused by the increasing β_t rather than overfitting on the validation set. Here $N_p = \frac{3C}{2}$ is applied, so that the training can stop only if the validation loss increases under the same value of β_t.

With all other settings kept the same as Model 1, experiments on $C = 10, 20, 30$ with

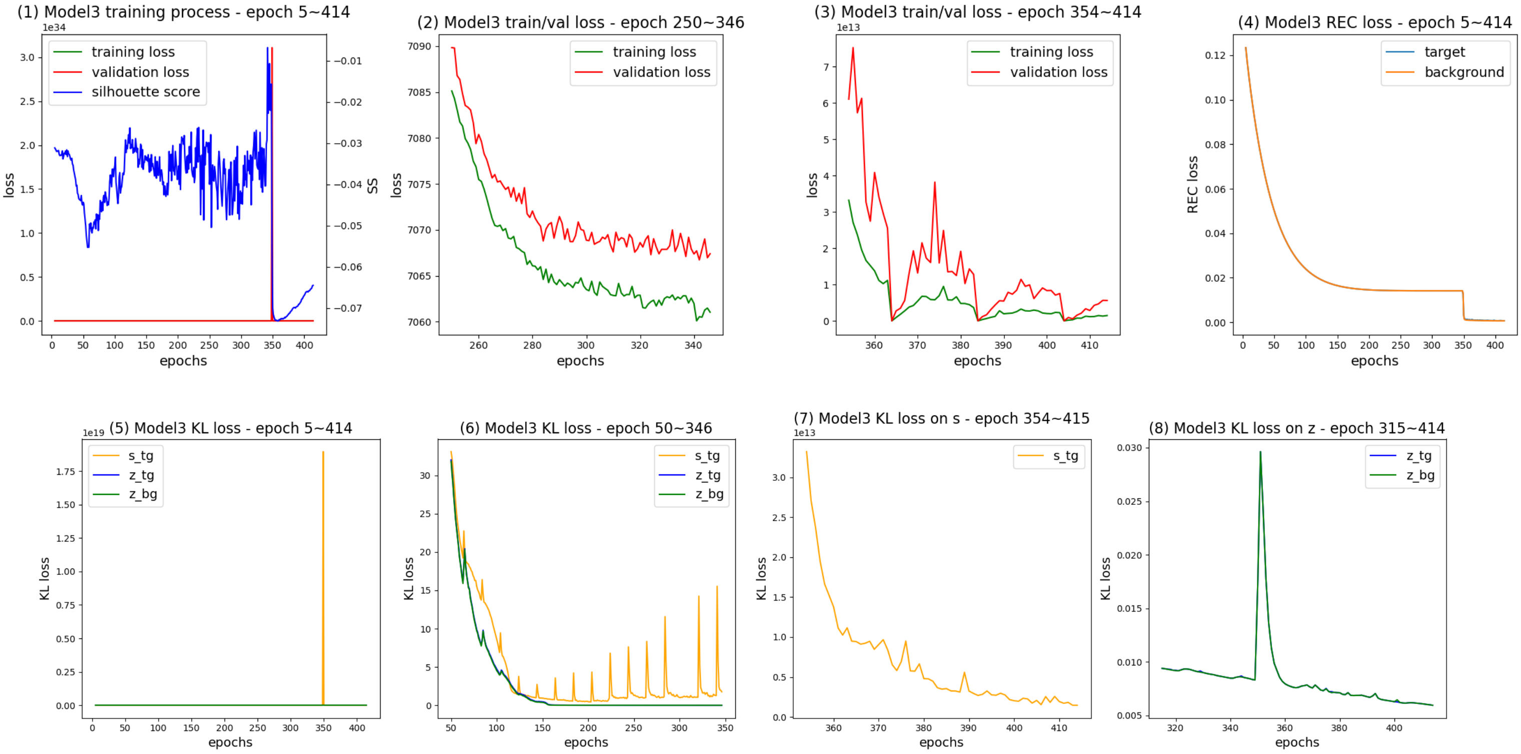


Figure 3.14: Model 3 – Loss Curves

fixed $l = 5$ have been conducted. However, the annealing schedule fails to mitigate the KL vanishing problem in our case. For all the experiments, the KL loss on the irrelevant features decrease to lower than 0.1 at the last 100 epochs. Here we report the results of the experiment with $C = 20$ as it has the highest minimum KL loss on the irrelevant features. The training stops at epoch 414 and Model 3 is picked at epoch 384 as it has the smallest validation loss so far. The evolution of the latent space on salient features is visualized in Figure 3.13 at every one-third of the training process. The loss curves of the cVAE are plotted in Figure 3.14. The discriminator is also trained successfully in this setting. The plots of the discriminator-related indicators are put in Figure A.1 in Appendix A.

As can be seen from Figure 3.14 (1, 5), there exists an explosion of training loss (to 10^{18}) and validation loss (to 10^{34}) at around epoch 350, which is caused by the explosion of the KL loss on the salient features. From Figure 3.14 (6), it can be seen that the for each annealing cycle the salient feature tends to "explode" to a much lower value compared with the explosion at epoch 350. As displayed in Figure 3.14 (5, 6, 7, 8), the KL loss on the salient features decreases to about 1.0 before the explosion, jumps abruptly to about 1.89×10^{19} at the explosion, then returns back to about 6.41×10^{13}

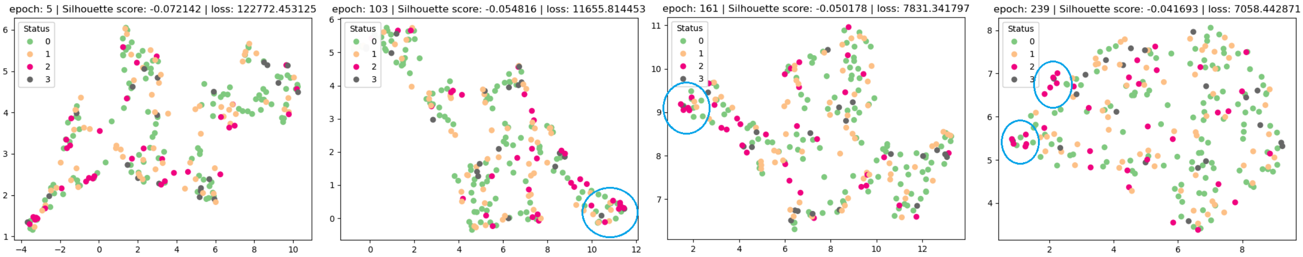


Figure 3.15: Model 4 – Latent Space of Salient Features at epoch 5, 103, 161 and 239 (0-HC, 1-MDD, 2-diabetes, 3-dual)

and keeps decreasing to about 1.46×10^{12} afterwards. The KL loss on the irrelevant features decreases to below 0.1 at about 160 epochs, reaches about 0.0084 before the explosion, jumps abruptly to about 0.030 at epoch 351, then returns back to about 0.0082 and keeps decreasing to about 0.0060 afterwards. The range and trend of the KL loss on the last 100 epochs are also recorded in Table 3.2.

It can be concluded that the annealing schedule we experiment with does not fit our cVAE, as it fails to mitigate the KL vanishing on the irrelevant features and encourage the KL exploding on the salient features. Other settings on C and l, or other annealing schedules might be needed. Since the vanishing problem tends to take place on either the the KL loss on the salient features or the KL loss on the irrelevant features, it is suggested to assigning different weights to the two KL terms, and applying different schedules on them. It is also unclear that which KL term will vanish while training, thus studies on under which conditions which KL term tends to vanish can also help. Additionally, as can be seen from Figure 3.14 (3), the loss curves after the explosion oscillate extremely widely and unstably, with the higher values reaching 10^{13} and the lower values only around 5×10^3. This is possibly caused by that the parameters of the model might be in a narrow valley of the loss landscape during training, and a fixed learning rate moves the parameters to some large peaks in the landscape. Hence, implementing a learning rate schedule that can decrease the value of λ as training goes on is also suggested.

- **Model 4: cRAE with** $\alpha = 250000, \beta = 1, \gamma = 0.01$

Another way to mitigate the KL vanishing problem is to avoid the usage of the KL loss. According to [42], a VAE can become deterministic by substituting squared L2 norm on latent features for the KL term to form a regularized auto-encoder (RAE). Here, we replace the $\mathbf{L}_{KL}$ in Model 1 by $\mathbf{L}_f = \frac{1}{2}||\mathbf{s}_t||_2^2 + \frac{1}{2}||\mathbf{z}_t||_2^2 + \frac{1}{2}||\mathbf{z}_b||_2^2$ to form a contrastive RAE (cRAE). All other settings are kept the same as Model 1.

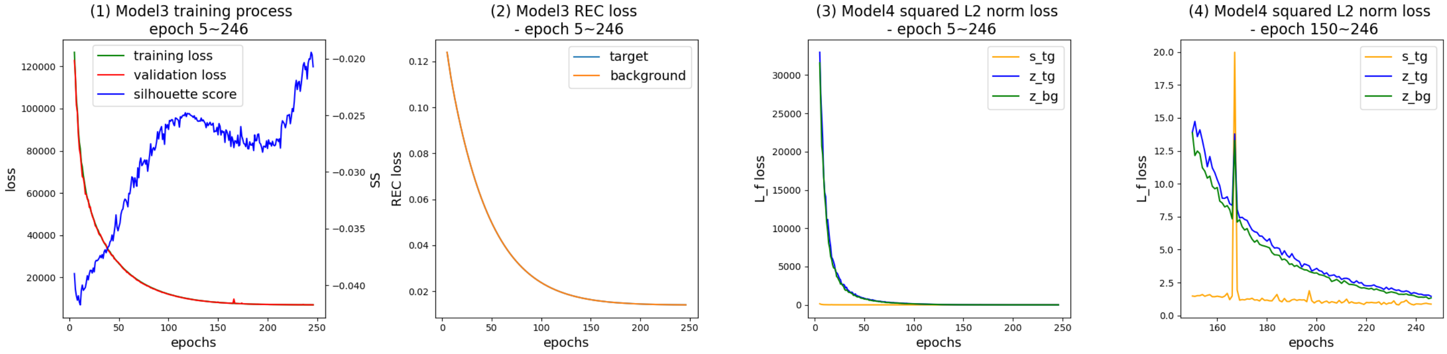


Figure 3.16: Model 4 – Loss Curves

The training is early stopped at epoch 246 as no validation loss increase for 7 epochs. Model 4 is picked at epoch 239 as it has the lowest validation loss so far. The loss curves are plotted in Figure 3.16. As can be seen, the validation SS curve oscillates much less widely when compared with the baseline, Model 1 and Model 3, which can indicate a smoother updating of the latent space. The reconstruction loss, similar to all the previous models, decreases smoothly from around 0.12 to around 0.01. The $\mathbf{L}_f$ on the salient features, despite several small explosions, decreases sharply from 146.82 to 4.75 within the first 50 epochs and levels at around 1.08 after epoch 100. The $\mathbf{L}_f$ on the irrelevant features decreases smoothly from above 3000 to 1.3931. The discriminator-related indicators are plotted in Figure A.2 in Appendix A, which indicates a normal performance of the discriminator since its $\bar{\mathbf{v}}$ score and $\hat{\mathbf{v}}$ score both approach $\frac{1}{2}$. The evolution of the latent space on the salient features is visualized in Figure 3.15 at every one-third of the training process. It can be discovered that several salient features of the diabetes samples tend to form small clusters at the lower right corner in the plot of epoch 103, and at the middle left part in the plot of epoch 161 and 239, as circled in blue in Figure 3.15.

3.4 Evaluation

The models are evaluated on an unseen test set. The test set is picked randomly from the set of all the available samples collected from UKBB excluding the scans used during training. The test set contains 40 samples for each scan type ("HC" / "MDD" / "diabetes" / "dual").

As explained in section 2.5, SS and average NMI are computed on the MDD, diabetes and dual scans for each of the five models described above. The latent space of

Models	Settings							Val	Test	
	α	β	γ	$\mathcal{G}$	Vanish	Anneal	Loss	SS	SS	Avg NMI
baseline	250000	1	100	×	✓	×	KL	0.000000	0.000000	0.000000
1	250000	1	0.01	✓	✓	×	KL	**0.008120**	**-0.026349**	0.017399
2	250000	1	None	None	×	None	KL	-0.017647	-0.026674	0.014950
3	250000	1	0.01	✓	✓	✓	KL	-0.069846	-0.037510	0.017335
4	250000	1	0.01	✓	None	None	L2	-0.020680	-0.028302	**0.017470**

Table 3.3: Model Evaluation Column Meaning: "$\mathcal{G}$" – whther the discriminator is trained successfully; "Vanish" – whether the KL vanishing problem takes place; "Anneal" – whether the cyclical annealing schedule on KL β is implemented; "Loss": whether the KL loss or the squared L2 norm loss is used. The highest values of validation SS, test SS, test average NMI among the four improved models are marked in bold, and the lowest values are underlined.

the salient features on the test set is visualized in Figure 3.17. The evaluation results on the test set, the validation SS at the epoch the model is picked, and related model settings are summarized in Table 3.3. The highest values of SS and NMI among the four improved models (Model 1 $\sim$ 4) are highlighted in bold, and the lowest values are underlined.

It can be seen from Table 3.3 that, in terms of test SS, the lowest and highest value are obtained by Model 3 and Model 1. Model 1 and 2 yield similar values of test SS. In terms of test average NMI, the lowest and highest value are obtained by Model 2 and Model 4. Model 1, 3 and 4 yield very close values of NMI. In general, Model 1 (the one with a successfully trained discriminator) can be considered to have the best performance, since it obtains the highest test SS and the second-highest average NMI among the four improved models, although the visualizations of the distributions on the latent space of Model 1,2,4 seem quite similar.

As displayed in Figure 3.17, it seems that the baseline model learns to distribute the features almost randomly. Thus its test SS and average NMI are 0, because GMM tends to assign the same label to all the data points even with the class number being set to 3. This means clustering in the latent space can be hardly discovered. The model probably fails to disentangle any disease-specific features. Hence the baseline model can not be reckoned as a valid model of the project. It can also be discovered that Model 3 (the one with a successfully trained discriminator and an annealing schedule on KL β) yields the slimmest distribution in the latent space. It obtains the lowest test SS among all the

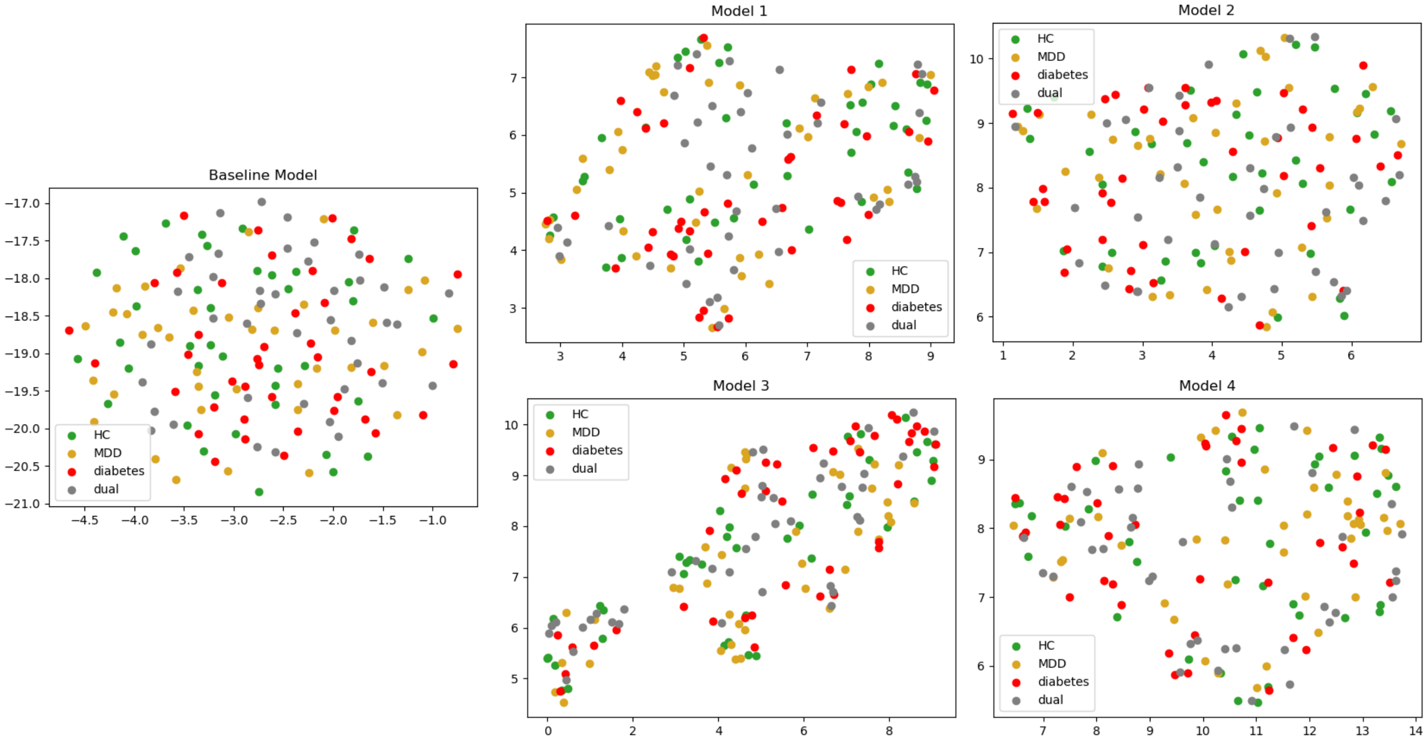


Figure 3.17: Latent Space of Salient Features on Test Set

models.

Combining the results of the baseline model, Model 1 and Model 2, it can be concluded that a missing or failed discriminator does harm the performance of a cVAE. The cVAE with malfunctioned discriminator (the baseline) yields the worst performance as it fails to form any cluster in the latent space. In terms of either the test SS or the test average NMI, the cVAE with a successfully trained discriminator (Model 1) outperforms the cVAE without a discriminator (Model 2). Combining the results of Model 1, Model 3 and Model 4, it can be concluded that the annealing schedule on KL β fails to improve the performance of a cVAE, as Model 3 has both a lower test SS and a slightly lower test average NMI compared with Model 1. The implementation of cRAE (Model 4 vs. Model 1) helps to improve the clustering performance in terms of the test average NMI, but impairs the performance when considering the test SS. Moreover, it is worth notifying that all the results so far are quite negative. Although except for the baseline model, salient features inferred by the other four models tend to form clusters in the latent space, none of the models can be declared to be trained successfully, as for now no obvious groupings consistent with the scan types are discovered in the latent space.

Chapter 4

Conclusion

This project makes attempts on using cVAE-liked models to disentangle MDD and/or diabetes specific patterns, and compare the performance between the models using test SS and test average NMI.

At first, SFCN and LBC are applied to compute the BAG of each sample. As expected, the distribution center and the MAG of diabetes and dual scans are higher than those of the HC scans. However, the MDD and the HC scans share similar normalized distributions, and the MDD scans have lower MAG than the HC scans, which do not match our expectation. Then scans are selected with positive BAG to enrich the desired patterns, and then the experiments on cVAE-liked models are conducted.

The baseline model adopts the hyper-parameter setting from [22]. However, its discriminator fails to function effectively, which might result in an almost random distribution in the latent space of the salient features and fails to disentangle the disease-related features. It is discorvered that either the mean $\mathcal{G}_\psi(\bar{\mathbf{v}})$ and the mean $\mathcal{G}_\psi(\hat{\mathbf{v}})$ approaching $\frac{1}{2}$ or the $\bar{\mathbf{v}}$ accuracy and the $\hat{\mathbf{v}}$ accuracy approaching $\frac{1}{2}$ is a sufficient condition for an effective discriminator. Hence, experiments on smaller TC loss weight γ were conducted and found that $\gamma = 0.01$ can lead to a successful training of the discriminator in our project. An ablation study on the discriminator was also conducted. Considering the evaluation results, it can be concluded that an effective discriminator is crucial to the performance of cVAE, as Model 1yields the best SS and Model 2 yields the lowest test average NMI.

After taking a detailed look at the loss curves of the baseline model, Model 1 and Model 2, we found that either the KL loss on the salient features or the KL loss on the irrelevant features tends to approach zero at the later stage of the training. Thus the cyclical annealing schedule on KL loss weight β is implemented. However, the schedule fails to prevent the KL vanishing problem as the KL loss on the irrelevant features of Model 3 decreases to around 0.0060 at last. Moreover, the schedule might encourage the KL exploding issue on the salient features. Considering the evaluation results, the schedule also fails to improve the performance of the cVAE, as both test SS and test average NMI of Model 3 are smaller than those of Model 1. Afterwards, we intend to avoid the KL vanishing problem by replacing the KL loss term with the squared L2 norm term. In this way, a variational AE becomes a deterministic RAE. Considering the evaluation results, the RAE can improve the performance of the cVAE since Model 4 has the highest test average NMI among all the models.

However, unfortunately, although according to the visualization plots, the four improved models tend to form clusters in the latent space of the salient features, there's no obvious groupings consistent with the scan types can be found. It should also be kept in mind that due to the time limit of the project and the length of time to train a model (roughly 21 hours per training), not enough experiments have been done to fully explore the research questions. Hence further studies on the implementation of cVAE for the MDD and/or diabetes specific patterns are needed. Here we make suggestions on the following for future research:

▷ 1. Increasing the number of samples used for training and balancing the number of samples of different scan types. The dataset used for cVAE in this project might still be too small considering the large size of the model. Data augmentation techniques might help with balancing the samples.

▷ 2. Increasing the latent dimension d to above 128. A $d = 32$ can be too small to present the changes of the disease-related patterns.

▷ 3. Implementing learning rate schedules. The updating of the parameters can sometimes enter a narrow valley in the loss landscape (as in Model 3). A learning rate of a constant value might lead to highly unstable loss changes and result in unexpected peaks in the loss curves.

▷ 4. Using different weights for the KL loss on the salient features and the KL loss on the irrelevant features. This can help figure out which KL loss tends to vanish and apply mitigation strategies separately.

▷ 5. Applying different optimisers to the cVAE and its discriminator. The discriminator can be seen as an extra module that is independent of other modules of the cVAE. Updating them in an asynchronous way might help improve the performance.

Bibliography

[1] L. K. M. Han, R. Dinga, T. Hahn, C. R. K. Ching, L. T. Eyler, L. Aftanas, M. Aghajani, and et al., “Brain aging in major depressive disorder: results from the ENIGMA major depressive disorder working group,” *Molecular Psychiatry*, vol. 26, no. 9, pp. 5124–5139, Sep. 2021, number: 9 Publisher: Nature Publishing Group. [Online]. Available: https://www.nature.com/articles/s41380-020-0754-0

[2] M. K. Jha, C. R. Chin Fatt, A. Minhajuddin, T. L. Mayes, J. D. Berry, and M. H. Trivedi, “Accelerated brain aging in individuals with diabetes: Association with poor glycemic control and increased all-cause mortality,” *Psychoneuroendocrinology*, vol. 145, p. 105921, Nov. 2022.

[3] M. Symms, H. R. Jäger, K. Schmierer, and T. A. Yousry, “A review of structural magnetic resonance neuroimaging,” *Journal of Neurology, Neurosurgery & Psychiatry*, vol. 75, no. 9, pp. 1235–1244, Sep. 2004, publisher: BMJ Publishing Group Ltd Section: Neuroscience for neurologists. [Online]. Available: https://jnnp.bmj.com/content/75/9/1235

[4] M. Tanveer, M. A. Ganaie, I. Beheshti, T. Goel, N. Ahmad, K.-T. Lai, K. Huang, Y.-D. Zhang, J. Del Ser, and C.-T. Lin, “Deep learning for brain age estimation: A systematic review,” *Information Fusion*, vol. 96, pp. 130–143, Aug. 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S156625352300088X

[5] J. Wrigglesworth, P. Ward, I. H. Harding, D. Nilaweera, Z. Wu, R. L. Woods, and J. Ryan, “Factors associated with brain ageing - a systematic review,” *BMC Neurology*, vol. 21, no. 1, p. 312, Aug. 2021. [Online]. Available: https://doi.org/10.1186/s12883-021-02331-4

[6] F. Liem, G. Varoquaux, J. Kynast, F. Beyer, S. Kharabian Masouleh, J. M. Huntenburg, L. Lampe, M. Rahim, A. Abraham, R. C. Craddock, S. Riedel-Heller,

T. Luck, M. Loeffler, M. L. Schroeter, A. V. Witte, A. Villringer, and D. S. Margulies, “Predicting brain-age from multimodal imaging data captures cognitive impairment,” *NeuroImage*, vol. 148, pp. 179–188, Mar. 2017.

[7] K. Franke, G. D. Clarke, R. Dahnke, C. Gaser, A. H. Kuo, C. Li, M. Schwab, and P. W. Nathanielsz, “Premature Brain Aging in Baboons Resulting from Moderate Fetal Undernutrition,” *Frontiers in Aging Neuroscience*, vol. 9, p. 92, 2017.

[8] J. H. Cole, J. Underwood, M. W. A. Caan, D. De Francesco, R. A. van Zoest, R. Leech, F. W. N. M. Wit, P. Portegies, G. J. Geurtsen, B. A. Schmand, M. F. Schim van der Loeff, C. Franceschi, C. A. Sabin, C. B. L. M. Majoie, A. Winston, P. Reiss, D. J. Sharp, and COBRA collaboration, “Increased brain-predicted aging in treated HIV disease,” *Neurology*, vol. 88, no. 14, pp. 1349–1357, Apr. 2017.

[9] L. Dular and Špiclin, “Improving Across Dataset Brain Age Predictions Using Transfer Learning,” in *Predictive Intelligence in Medicine*, ser. Lecture Notes in Computer Science, I. Rekik, E. Adeli, S. H. Park, and J. Schnabel, Eds. Cham: Springer International Publishing, 2021, pp. 243–254.

[10] N. K. Dinsdale, E. Bluemke, S. M. Smith, Z. Arya, D. Vidaurre, M. Jenkinson, and A. I. L. Namburete, “Learning patterns of the ageing brain in MRI using deep convolutional networks,” *NeuroImage*, vol. 224, p. 117401, Jan. 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1053811920308867

[11] S. He, P. E. Grant, and Y. Ou, “Global-Local Transformer for Brain Age Estimation,” *IEEE transactions on medical imaging*, vol. 41, no. 1, pp. 213–224, Jan. 2022.

[12] H. Peng, W. Gong, C. F. Beckmann, A. Vedaldi, and S. M. Smith, “Accurate brain age prediction with lightweight deep neural networks,” *Medical Image Analysis*, vol. 68, p. 101871, Feb. 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1361841520302358

[13] C. Sudlow, J. Gallacher, N. Allen, V. Beral, P. Burton, J. Danesh, P. Downey, P. Elliott, J. Green, M. Landray, B. Liu, P. Matthews, G. Ong, J. Pell, A. Silman, A. Young, T. Sprosen, T. Peakman, and R. Collins, “UK Biobank: An Open Access Resource for Identifying the Causes of a Wide Range of Complex Diseases of Middle and Old Age,” *PLOS Medicine*, vol. 12, no. 3, p.

e1001779, Mar. 2015, publisher: Public Library of Science. [Online]. Available: https://journals.plos.org/plosmedicine/article?id=10.1371/journal.pmed.1001779

[14] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition." IEEE Computer Society, Jun. 2016, pp. 770–778, iSSN: 1063-6919. [Online]. Available: https://www.computer.org/csdl/proceedings-article/cvpr/2016/8851a770/12OmNxvwoXv

[15] S. M. Smith, D. Vidaurre, F. Alfaro-Almagro, T. E. Nichols, and K. L. Miller, "Estimation of brain age delta from brain imaging," *NeuroImage*, vol. 200, pp. 528–539, Oct. 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1053811919305026

[16] A.-M. G. de Lange and J. H. Cole, "Commentary: Correction procedures in brain-age prediction," *NeuroImage : Clinical*, vol. 26, p. 102229, Feb. 2020. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7049655/

[17] A. Jaiswal, A. R. Babu, M. Z. Zadeh, D. Banerjee, and F. Makedon, "A Survey on Contrastive Self-Supervised Learning," *Technologies*, vol. 9, no. 1, p. 2, Mar. 2021, number: 1 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: https://www.mdpi.com/2227-7080/9/1/2

[18] R. Louiset, E. Duchesnay, A. Grigis, B. Dufumier, and P. Gori, "SepVAE: a contrastive VAE to separate pathological patterns from healthy ones," Jul. 2023. [Online]. Available: https://arxiv.org/abs/2307.06206v1

[19] A. Abid, M. J. Zhang, V. K. Bagaria, and J. Zou, "Exploring patterns enriched in a dataset with contrastive principal component analysis," *Nature Communications*, vol. 9, no. 1, p. 2134, May 2018, number: 1 Publisher: Nature Publishing Group. [Online]. Available: https://www.nature.com/articles/s41467-018-04608-8

[20] D. Li, A. Jones, and B. Engelhardt, "Probabilistic Contrastive Principal Component Analysis," *ArXiv*, Dec. 2020. [Online]. Available: https://www.semanticscholar.org/paper/Probabilistic-Contrastive-Principal-Component-Li-Jones/258dfbbba3aebdb3b8e78f0921a273bdde53c576

[21] A. Abid and J. Zou, "Contrastive Variational Autoencoder Enhances Salient Features," Feb. 2019, arXiv:1902.04601 [cs, stat]. [Online]. Available: http://arxiv.org/abs/1902.04601

[22] A. Aglinskas, J. K. Hartshorne, and S. Anzellotti, “Contrastive machine learning reveals the structure of neuroanatomical variation within autism,” *Science*, vol. 376, no. 6597, pp. 1070–1074, Jun. 2022, publisher: American Association for the Advancement of Science. [Online]. Available: https://www.science.org/doi/10.1126/science.abm2461

[23] J. M. Joyce, “Kullback-Leibler Divergence,” in *International Encyclopedia of Statistical Science*, M. Lovric, Ed. Berlin, Heidelberg: Springer, 2011, pp. 720–722. [Online]. Available: https://doi.org/10.1007/978-3-642-04898-2$_3$27

[24] H. Fu, C. Li, X. Liu, J. Gao, A. Celikyilmaz, and L. Carin, “Cyclical Annealing Schedule: A Simple Approach to Mitigating KL Vanishing,” Jun. 2019, arXiv:1903.10145 [cs, stat]. [Online]. Available: http://arxiv.org/abs/1903.10145

[25] F. Alfaro-Almagro, M. Jenkinson, N. K. Bangerter, J. L. R. Andersson, L. Griffanti, G. Douaud, S. N. Sotiropoulos, S. Jbabdi, M. Hernandez-Fernandez, E. Vallee, D. Vidaurre, M. Webster, P. McCarthy, C. Rorden, A. Daducci, D. C. Alexander, H. Zhang, I. Dragonu, P. M. Matthews, K. L. Miller, and S. M. Smith, “Image processing and Quality Control for the first 10,000 brain imaging datasets from UK Biobank,” *NeuroImage*, vol. 166, pp. 400–424, Feb. 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1053811917308613

[26] K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition,” Apr. 2015, arXiv:1409.1556 [cs]. [Online]. Available: http://arxiv.org/abs/1409.1556

[27] J. Long, E. Shelhamer, and T. Darrell, “Fully convolutional networks for semantic segmentation,” in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015, pp. 3431–3440, iSSN: 1063-6919.

[28] D. P. Kingma and M. Welling, “Auto-Encoding Variational Bayes,” Dec. 2013, arXiv:1312.6114 [cs, stat]. [Online]. Available: http://arxiv.org/abs/1312.6114

[29] P. Ghosh, M. S. M. Sajjadi, A. Vergari, M. Black, and B. Schölkopf, “From Variational to Deterministic Autoencoders,” May 2020, arXiv:1903.12436 [cs, stat]. [Online]. Available: http://arxiv.org/abs/1903.12436

[30] D. P. Kingma and J. Ba, “Adam: A Method for Stochastic Optimization,” Jan. 2017, arXiv:1412.6980 [cs]. [Online]. Available: http://arxiv.org/abs/1412.6980

[31] L. McInnes, J. Healy, and J. Melville, “UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction,” Sep. 2020, arXiv:1802.03426 [cs, stat]. [Online]. Available: http://arxiv.org/abs/1802.03426

[32] M. Shutaywi and N. N. Kachouie, “Silhouette Analysis for Performance Evaluation in Machine Learning with Applications to Clustering,” *Entropy*, vol. 23, no. 6, p. 759, Jun. 2021. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8234541/

[33] P. A. Estevez, M. Tesmer, C. A. Perez, and J. M. Zurada, “Normalized Mutual Information Feature Selection,” *IEEE Transactions on Neural Networks*, vol. 20, no. 2, pp. 189–201, Feb. 2009, conference Name: IEEE Transactions on Neural Networks.

[34] D. Reynolds, “Gaussian Mixture Models,” in *Encyclopedia of Biometrics*, S. Z. Li and A. Jain, Eds. Boston, MA: Springer US, 2009, pp. 659–663. [Online]. Available: https://doi.org/10.1007/978-0-387-73003-5$_1$96

[35] “UKBB Data-Field 20252.” [Online]. Available: https://biobank.ndph.ox.ac.uk/showcase/field.cgi?id=20252

[36] M. Brett, I. S. Johnsrude, and A. M. Owen, “The problem of functional localization in the human brain,” *Nature Reviews Neuroscience*, vol. 3, no. 3, pp. 243–249, Mar. 2002, number: 3 Publisher: Nature Publishing Group. [Online]. Available: https://www.nature.com/articles/nrn756

[37] M. Larobina and L. Murino, “Medical Image File Formats,” *Journal of Digital Imaging*, vol. 27, no. 2, pp. 200–206, Apr. 2014. [Online]. Available: https://doi.org/10.1007/s10278-013-9657-9

[38] “UKBB Data-Field 21003.” [Online]. Available: https://biobank.ndph.ox.ac.uk/showcase/field.cgi?id=21003

[39] “UKBB Data-Field 20126.” [Online]. Available: https://biobank.ctsu.ox.ac.uk/crystal/field.cgi?id=20126

[40] “UKBB Data-Coding 100695.” [Online]. Available: https://biobank.ctsu.ox.ac.uk/crystal/coding.cgi?id=100695

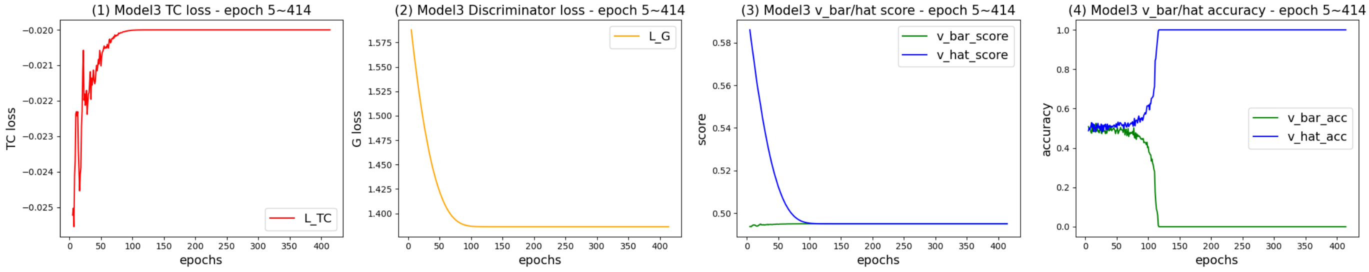
[41] “UKBB Data-Field 2976.” [Online]. Available: https://biobank.ctsu.ox.ac.uk/crystal/field.cgi?id=2976

[42] P. Ghosh, M. S. M. Sajjadi, A. Vergari, M. Black, and B. Scholkopf, “From Variational to Deterministic Autoencoders,” Sep. 2019. [Online]. Available: https://openreview.net/forum?id=S1g7tpEYDS

Appendix A

Plots

- **Model 3**


(1) Model3 TC loss - epoch 5~414

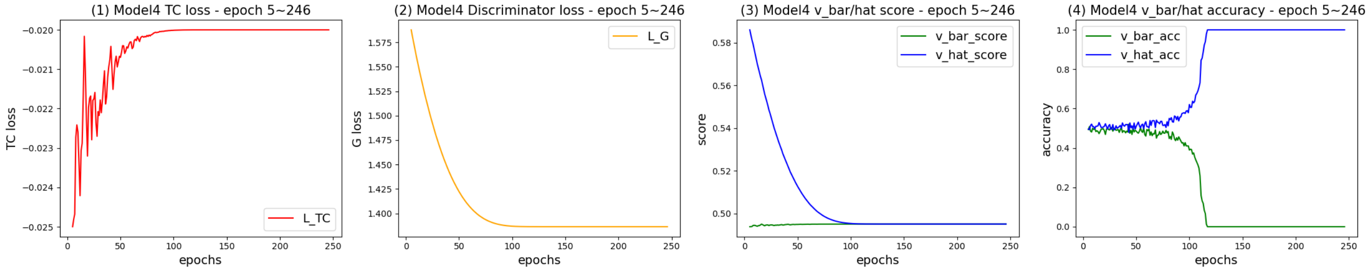
(2) Model3 Discriminator loss - epoch 5~414

(3) Model3 v_bar/hat score - epoch 5~414

(4) Model3 v_bar/hat accuracy - epoch 5~414

Figure A.1: Model 3 – Discriminator Performance

- **Model 4**


(1) Model4 TC loss - epoch 5~246

(2) Model4 Discriminator loss - epoch 5~246

(3) Model4 v_bar/hat score - epoch 5~246

(4) Model4 v_bar/hat accuracy - epoch 5~246

Figure A.2: Model 4 – Discriminator Performance