Cluster Invariant Representation Learning for Generalizing Brain Tumor Segmentation Models

Jonathan Friesen
University of Washington

Juampablo Heras Rivera
University of Washington

Nirmal Kadirkamanathan
University of Washington

Eric Gibson
University of Washington

1 Introduction

Current deep learning based tumor segmentation efforts are trained on highly-specialized datasets typically containing little variation in factors such as lesion types, medical institutions, and demographics. This specialized training limits the generalizability and robustness of the models, thereby reducing their potential for translation to a clinical setting. For this reason, we aim to create brain tumor segmentation algorithms capable of adapting and generalizing to different scenarios with little prior information or data on the target classes.

The strategy chosen involves framing the tumor segmentation generalization problem as a domain adaptation problem. In domain adaptation problems the aim is to simultaneously improve performance in all domains. In this context, we define a domain as a group in an artificial clustering of the training datasets, and we aim to improve performance over all clusters. A natural clustering for this dataset would cluster based on tumor type; however, given that we do not have this information, the idea is to produce new clusterings using an unsupervised approach. Furthermore, we analyze the sensitivity of the segmentation model performance to the clustering approach chosen, and interpret the results to try to understand which sets of features improve segmentation. To perform clustering of the training dataset, we plan to test traditional approaches including K-means clustering, Hierarchical clustering, and Gaussian Mixture Models. We then in turn consider a more recently proposed approach of clustering on the Latent Space of a (Variational) Autoencoder.

To perform the domain adaptation, we use a deep learning approach inspired by Domain Adversarial Neural Networks (DANN) [2], generalized from binary classification to multiclass classification. DANN aims to learn features that are both discriminative and domain-invariant by optimizing a neural network comprising two classifiers: a label predictor, which provides segmentations and is used both during training and testing, and a domain classifier that discriminates between domains during training. The optimization process minimizes the label classifier loss while maximizing the domain classifier loss, promoting the emergence of domain-invariant features through an adversarial optimization.

Our implementation of the domain adaptation approach consists of a backbone U-Net architecture for brain tumor segmentation with additional modifications to make it compatible with the DANN approach. Instead of classifying on domains, we optimize a model to classify based on the artificial clusters produced, effectively regularizing the model to perform well on the implicit classes present in the data.

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

2 Related Work

2.1 Clustering

In the first section of our project, we consider clustering approaches of large scale medical images. Using summary statistics to avoid the curse of dimensionality naturally yields up the classic algorithms of K-Means, Spectral and Gaussian Mixed Modeling methods. The literature on these topics are far too large to give an overview, and our engagement with them was fairly shallow.

However, we did take inspiration from the paper "Unsupervised Deep Embedding for Clustering Analysis" by Junyuan Xie, Ross Girshick, and Ali Farhadi, which introduces Deep Embedded Clustering (DEC), a method that simultaneously learns feature representations and cluster assignments using autoencoders [4]. The authors propose a new approach that optimizes a clustering objective in a lower-dimensional feature space, showing significant improvements over these other methods across various benchmarks. This type of method avoids the curse of dimensionality in clustering by performing said clustering over a low dimensional latent embedding space, and with the dynamic representation learning of an autoencoder, we would hope to find a "richer" set of features to base these clusters on, and so output a

2.2 Variational DEC

On the topic of performing DEC for our domain selection, Soleymani et al 2021 [8] propose that the use of a Variational Autoencoder, which instead of mapping into a deterministic latent space, forms an encoding map into the space of multivariate Gaussian distributions. This form of embedding is proposed to be more suited to large scale medical image clustering, as this model

- is more suited for medical images, as biological features generally tend to follow multivariate normal distributions compared to deterministic low dimensional manifolds
- tends to reduce number of parameters to be optimimzed, more robust model

The paper also tests their models against the state-of-the-art as well as DEC itself. In fact they tested on the very dataset we used, showing promising results.

2.3 Domain-Adversarial Training of Neural Networks (DANN)

The paper "Domain-Adversarial Training of Neural Networks" introduced a simple deep learning architecture for domain adaptation by means of extraction of domain-independent features. In the context of domain adaptation, where data at training and test time come from similar but different distributions, domain-independent features are those whose origin cannot be attributed to either the training or test domains. To extract domain-independent features, Domain-Adversarial Neural Networks (DANNs) optimize a neural network comprising two layers, namely a predictor (particular to the task of interest, in our case a segmentation classifier), and a domain classifier that discriminates between domains during testing. In our project, we want to use this architecture to perform a sort of domain adaption over the clusters learned in the first part, comparing generalizability of the different clustering methods.

3 Methods

Mathematically formulated our domain implicit representation problem, involves first cluster N samples $\{x_i\}_{i=1}^N \subseteq \mathcal{X} = \mathbb{R}^{\tilde{d}_1 \times \tilde{d}_2 \times \ldots \times \tilde{d}_p}$ into K clusters, from which we will form a partition of $\mathcal{X}$ into $\{\mathcal{D}_i\}_{i=1}^d$, i.e. some clustering map $c : \mathcal{X} \to \mathcal{D}$. We consider a number of approaches of obtaining such a map (such as K-Means, Hierarchical, Gaussian Mixed Model) which are standard, and so we will not elucidate here. By far the most novel and mathematically sophisticated technique we use is Variational Deep Embedding Clustering (VDEC), which we give a brief formalization of.

This algorithm itself has two main steps.

3.1 VDEC

3.1.1 Pretrain

Define an **encoder network** $q_\phi(z|x) : x_i \to z_i \in \mathcal{Z}$, mapping into a latent embedding of input sample space $\mathcal{X}$ with smaller dimension. In tandem, we define a **decoder network** $p_\theta : z_i \to x_i \in \mathcal{Z}$, a reconstruction of points in the latent space. Together, $p_\theta(q_\phi(z|x))$ define an Autoencoder. In particular, we will be pretraining a Variational Autoencoder (VAE) per Soleymani et. al.

Reframing this problem in the Bayesian language of Variational Inference, we want to form a model with some fixed prior $p(z)$ usually chosen as standard normal:

$$p(z) \sim \mathcal{N}(0, I)$$

Then, looking into the variational approximation to the posterior:

$$q_\phi(z|x) = \mathcal{N}(z; \mu_\phi(x), \sigma^2_\phi(x)I) \tag{1}$$

$$p_\theta(x|z) = \mathcal{N}(x; \mu_\theta(z), \sigma_\theta(x)I) \text{ or } \text{Ber}(x; p_\theta(z)) \tag{2}$$

The standard variational approximation comes in with respect to the Evidence Lower Bound Loss (ELBO) $\mathcal{L}_n(\theta, \phi) := -\mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)] + \mathcal{D}_{KL}(q_\phi(z|x)||p(z))$. In the context of our network, $-\mathbb{E}_{q_\phi(z|x)}[\log p_\theta(x|z)] = \mathcal{L}_{CE}(x, x')$, the reconstruction with Binary Cross-Entropy. After we have reached convergence with this loss, we move on to the second stage.

3.1.2 Latent Clustering

We now want to find a clustering mapping with domain on the latent space by optimizing the loss $\mathcal{L} = \lambda\mathcal{L}_{CE} + (1 - \lambda)\mathcal{L}_c$, where $\mathcal{L}_c := \mathcal{D}_{KL}(P(q)\|Q(z, \mu))$ and $\mu_1, ..., \mu_d \in \mathcal{Z}$ are the d centroids for d clusters. P and Q are defined as follows:

$$q_{i,j} := \frac{(1_\| z_i - \mu_i\|^2/\alpha)^{-\frac{\alpha+1}{2}}}{\sum_{k\in[d]}(1 + \|z_k - \mu_k\|/\alpha)^{-\frac{\alpha+1}{2}}} \tag{3}$$

the distance between z_i and centroid μ_j according to a Student's t-distribution metric, with α being degrees of freedom, usually defaulted to $\alpha = 1$. We label z_i according to a sofmax over $q_{i,:}$. P is then defined as the target distribution for the iteration's q values. More specifically

$$p_{i,j} := \frac{q^2_{i,j}/(u_j + v_j)}{\sum_k q^2_{i,k}/v_k}, \tag{4}$$

with $u_j = \sum_k q_{i,k}$ the soft cluster frequencies and $v_j = -\sum_i \sum_j \sqrt{\frac{\sum_k N_k}{N_j}(1 - q_{i,j})^\gamma \log q_{i,j}}$ a normalizing factor with respect to the number of samples per cluster, with N_i representing carnality of $i'th$ cluster. γ is a hyperparameter usually set to 2.

We calculate run this algorithm iteratively, initializing $\mu_1, ..., \mu_d$ by K-means initialization, iterating until some convergence criterion is met for $\mathcal{L}$, updating the Q and P as described, and performing backpropogration on ϕ, θ, the encoder and decoder's parameters respectively. This brings us to the next step of our procedure.

3.2 Generalizable Model

Given the learned clustering map $c : \mathcal{X} \to \mathcal{D}$, we want to build a model which will encourage a generalizable performance across these learned domains $\mathcal{D}_1, ...\mathcal{D}_d$. To do this, we will work with a Domain Adversarial Neural Network (DANN), described as follows:

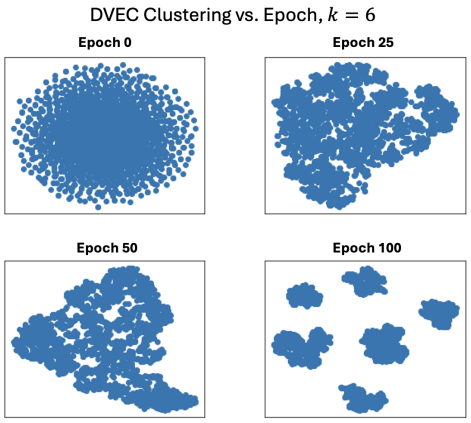

Figure 1: t-SNE visualization of Variational DEC clustering Algorithm, d = 6

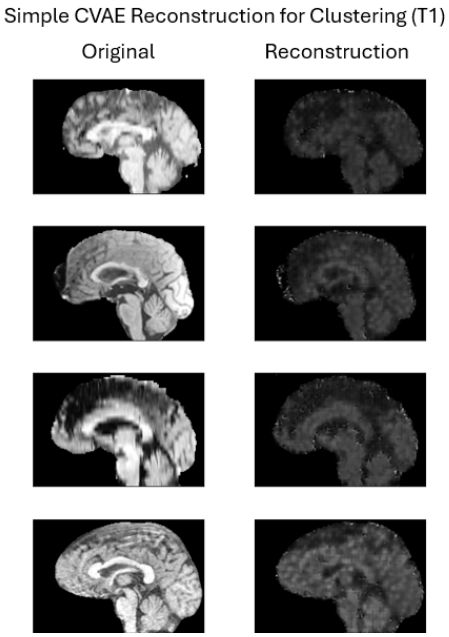

Figure 2: Reconstruction of Brain Tumors on CVAE found in pre-training

The primary componenet of our DANN comes in the form of another Autoencoder architecture, this time without the VAE layer. Our encoder $f'_\phi : x_i \to z_i \in \mathcal{Z}'$ maps to a lower dimensional, deterministic embedding of $\mathbb{R}^k$. Our decoder $g'_\theta : z_i \to x'_i \in \{0,1\}^{\tilde{d}_1 \times \ldots \times \tilde{d}_p}$, a segmentation mask over our input x. Importantly, $g'_\theta(f'_\phi(x))$ will take on the architecture of a U-Net with skip connections. This involves convolutions layers with some kernel width and stride defined at each layer.

Call the transformation made from $x^{(i)} \in \mathcal{X}^{()} \to \mathcal{Z}$, the sample space convolutional layer i, H_i, and the corresponding transpose convolutional transformation from $z \in \mathcal{Z}$ to layer i, H'_i. The skip connection is then defined as the functional relationship across each layer and its transpose: $y_i = H'_i(H_i(x^{(i)})) + x$, so at the $i'th$ transpose convolutional layer, we are given both the reconstructed x - $H'_i(H_i(x^{(i)}))$, as well as $x^{(i)}$, the image transformed by the first i convolutional layers.

We additionally train a domain classifier neural network, $G_{\theta_d} : z_i \to \{\mathcal{D}_1, ..., \mathcal{D}_d\}$, which attempts to learn the features of the latent representation which pick out domains. We denote the domain loss as $\mathcal{L}_d(\theta_f, \theta_d) := \mathcal{L}^i_d(G_{\theta_d}(G_{\theta_f}(x_i)), d_i)$, a loss between predicted and actual domain.

Our autoencoder is then optimized with the usual methods (we used ADAM), over the following loss function:

$$\mathcal{L}_{DANN}(\phi', \theta') = -\sum_{i=1}^{N} E[\log p_\theta(x_i|z_i)] + \frac{1}{n}\sum_{i=1}^{n} \mathcal{L}^i_d((z_i)_f, (z_i)_d)$$

the Binary Cross Entropy reconstruction loss, as well as the domain classification loss defined above.

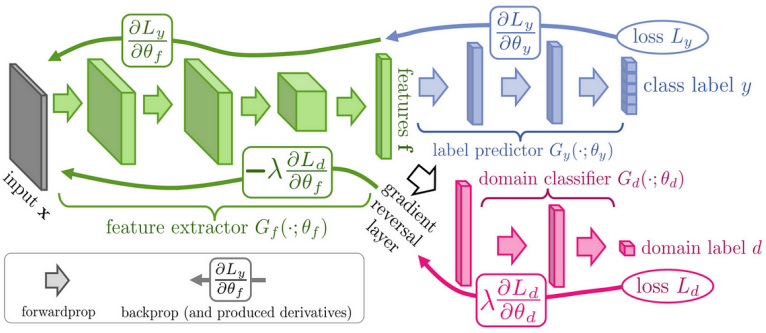

Figure 3: Visualization of DANN Architecture, from Ganin et al 2016[2]

4 Experiments

The dataset used for the project is sourced from the BraTS 2024 Generalizability Across Tumors (GoAT) challenge, which consists of multi-institutional, preoperative, routine clinically-acquired multi-parametric 3D MRI scans of dimension $240 \times 240 \times 155$ from 2200 brain tumor patients. Notably, the dataset contains scans of patients with varying types of tumors (e.g. early-stage glioma, glioblastoma, meningioma), and no explicit information regarding the type of tumor present in each image.

The data for each patient includes 4 different MRI contrasts, namely, native and post-contrast-enhanced T1-weighted, T2-weighted, and T2 Fluid Attenuated Inversion Recovery (FLAIR), provided in NIfTI format. Additionally, neuroradiologist annotations are included which outline the boundaries of the GD-enhancing tumor (ET), the peritumoral edematous/invaded tissue (ED), and the necrotic tumor core (NCR). The images are all co-registered to the same anatomical template, interpolated to the same resolution ($1\ mm \times 1\ mm \times 1\ mm$), and skull-stripped.

4.1 Initial Clustering

The MRI data in the dataset is high-dimensional ($4 \times 240 \times 240 \times 155 \approx 3.5 \times 10^7$ features per patient), posing a computational challenge. We thus began by evaluating traditional clustering methods; however, these methods suffer from the curse of dimensionality due to their low expressivity. Therefore, utilizing these methods necessitates and preliminary dimension reduction step. To this end, we extract radiomic features using the PyRadiomics package [5], which provides tabular features quantifying image intensity, shape, and texture differences. Radiomic features include first-order statistics (e.g., mean, variance), shape-based features (e.g., volume, surface area), and texture features (e.g., Gray Level Co-occurrence Matrix, Gray Level Run Length Matrix). These features are compiled into a structured, cleaned, and normalized dataset (CSV file) for detailed analysis. Table 1 illustrates example radiomic features such as tumor elongation, flatness, major axis length, energy, entropy, and maximum intensity. Elongation and flatness relate to the tumor shape's principal components, energy measures voxel value magnitude, and entropy quantifies the information content of the tumor image. We evaluated hierarchical, Gaussian Mixture Models, and k-means clustering methods on the radiomic feature data as described in this section, and results are shown in Table 2.

Table 1: Example radiomic features extracted from the GoAT dataset

Feature	Mean	StdDev	Min	Max
Elongation	0.672	0.185	0.0	1.0
Flatness	0.487	0.169	0.0	0.878
Major Axis Length	38.478	20.449	0.0	171.979
Energy	9.909E10	1.598E12	81934.920	3.898E13
Entropy	3.047	1.392	-3.2E-16	10.134
Maximum Intensity	1615.006	5121.823	50.0	91585.953

4.2 Deep Embedding Clustering

Following the modest results from the initial clustering, we hypothesized that the representations of the data used by the traditional methods are insufficient for effective clustering of the data. Furthermore, we then considered more expressive deep neural network models, as described in section 3.1. To train these models, we used 2D sagittal slices extracted from the MRI scans. The results of this clustering are shown in 2 under DVEC.

4.3 Segmentation

To evaluate the validity of our hypothesis, we trained our proposed architecture on the dataset using 5-fold cross validation for 50 epochs for 4 different clustering methods used. We chose the clusters extracted from the highest performing clustering methods for the adaptation. The models were trained with 2 NVIDIA A40 GPUs, and each model took roughly 12 hours to train. The codes to our implementation can be found at `https://github.com/juampabloheras/BraTS-GoAT`.

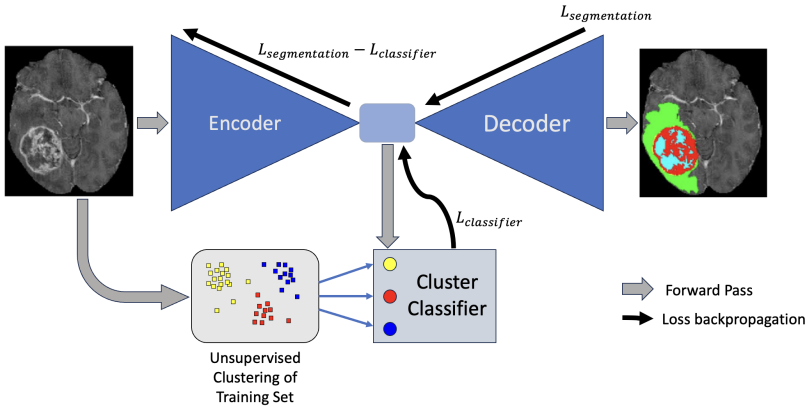

Figure 4: Diagram of our overall network architecture

5 Results

5.1 Clustering Results

After applying multiple clustering algorithms on our processed dataset, we obtained the following results, which are summarized in Table 2.

Table 2: Silhouette Scores for Clustering Results on Radiomics Processed Data

Algorithm	**Number of clusters, k**								
	2	3	4	5	6	7	8	9	10
Hierarchical	0.88	0.26	0.24	0.23	0.23	0.24	0.22	0.22	0.14
GMM	0.82	0.17	0.25	0.23	0.23	0.18	0.17	0.17	0.17
k-means	0.2846	0.3675	0.2542	0.2606	0.2879	0.3211	0.3051	0.2987	0.3586
DVEC	0.7494	0.8497	0.8495	0.7984	0.8574	0.8477	0.8642	0.6811	0.8900

Analysis of Results

The results show significant variation in silhouette scores across different clustering algorithms and cluster numbers, k. Hierarchical clustering performed best at $k = 2$ with a silhouette score of 0.88, indicating it works well for datasets with two distinct groups but declines at higher k values. The Gaussian Mixture Model (GMM) also excelled at $k = 2$ and remained stable across higher k values, showing its robustness for varying cluster numbers.

K-means achieved its best performance at intermediate k values (5 to 7), suggesting its capability to identify multiple subgroups within the dataset. DVEC showed strong performance across all k values, particularly excelling at $k = 10$ with the highest silhouette score of 0.89, indicating its effectiveness in distinguishing multiple clusters.

We chose to focus on clusters using $k = 3$ to $k = 6$ based on an intuitive analysis of cluster plots. These values were selected because they seemed to best capture the inherent structure in the data. Clustering with too few clusters might oversimplify the data, missing significant variations, while too many clusters could lead to overfitting and less meaningful groupings.

By analyzing different values of k within the range of 3 to 6, we aim to balance capturing the data's complexity with maintaining robust and generalizable clusters. This approach ensures that we do not miss out on potential subgroups within the dataset while avoiding the pitfalls of excessive clustering.

5.2 Segmentation Results

Models	Mean Dice	Dice Whole Tumor	Dice Tumor Core	Dice Enhancing Tumor
Baseline	**0.8655**	**0.8788**	**0.8701**	**0.8477**
GMM k=4 + 1	0.7724	0.6569	0.7095	0.7824
k-means k=3 + 1	0.7644	0.6737	0.7100	0.7806
DVEC k=6	0.7346	0.6437	0.7054	0.7621
DVEC k=3	0.7012	0.6414	0.6795	0.7419

In Table 5.2, the $k = 3 + 1$ for K-means and GMM indicates that the radiomics processing [5] was only able to handle around 1500 of the 2200 images, and the "+1" accounts for the images that could not be clustered due to processing limitations. This notation helps clarify that the clustering results are based on the subset of data that could be processed, with an additional cluster for unprocessed images.

Analysis of Segmentation Results

The summary of Dice scores above was generated over 5-Fold Cross Validations of each model. All of our cluster implementations were outperformed by the baseline U-NET architecture in terms of these dice metrics across each tumor region. This however does not necessarily have direct bearing on the the generalizability of our models. Due to time constraints on the project, we were not able to implement metrics to test this aspect of our models.

6 Next Steps

We have considered a few avenues for future directions this project could be taken. First of all, as was just mentioned, a concrete metric of generalizability should be implemented to test the overall domain classification task we set out to accomplish. The approach we have in mind would be a sort of variance of validation analysis, wherein we set aside some dedicated validation dataset from each cluster, and once the model is trained, test the variance of performance for the model across each of these clusters.

We recognize that the models may not have converged at the time of reporting, and training stopped at 50 epochs due to time constraints. In future work, we will run our models for more epochs, to ensure convergence before drawing conclusions. In addition, we worry the 2D scans used for clustering in VDEC was not sufficient to yield rich feature maps, so a next step would be in clustering on the full 3D scans, scaling whatever would be necessary to accomplish that task well.

References

[1] Alexander, J.A. & Mozer, M.C. (1995) Template-based algorithms for connectionist rule extraction. In G. Tesauro, D.S. Touretzky and T.K. Leen (eds.), *Advances in Neural Information Processing Systems 7*, pp. 609–616. Cambridge, MA: MIT Press.

[2] Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., March, M., & Lempitsky, V. (2016) Domain-adversarial training of neural networks. *Journal of Machine Learning Research, 17*(59), pp. 1–35.

[3] Hasselmo, M.E., Schnell, E. & Barkai, E. (1995) Dynamics of learning and recall at excitatory recurrent synapses and cholinergic modulation in rat hippocampal region CA3. *Journal of Neuroscience* **15**(7):5249-5262.

[4] Xie, J., Girshick, R., & Farhadi, A. (2016). Unsupervised Deep Embedding for Clustering Analysis. In Proceedings of The 33rd International Conference on Machine Learning, Proceedings of Machine Learning Research, 48, 478-487. Available from https://proceedings.mlr.press/v48/xieb16.html.

[5] van Griethuysen, J. J. M., Fedorov, A., Parmar, C., Hosny, A., Aucoin, N., Narayan, V., Beets-Tan, R. G. H., Fillon-Robin, J. C., Pieper, S., Aerts, H. J. W. L. (2017). Computational Radiomics System to Decode the Radiographic Phenotype. Cancer Research, 77(21), e104–e107. https://doi.org/10.1158/0008-5472.CAN-17-0339 <https://doi.org/10.1158/0008-5472.CAN-17-0339>.

[6] Munk, A, Nielsen, M.. (2024). MDD-UNet: Domain Adaptation for Medical Image Segmentation with Theoretical Guarantees, a Proof of Concept. Proceedings of the 5th Northern Lights Deep Learning Conference (NLDL), in Proceedings of Machine Learning Research 233:174-180 Available from https://proceedings.mlr.press/v233/munk24a.html.

[7] Wu, Fuping, and Xiahai Zhuang. "Unsupervised domain adaptation with variational approximation for cardiac segmentation." IEEE Transactions on Medical Imaging, vol. 40, no. 12, Dec. 2021, pp. 3555–3567, https://doi.org/10.1109/tmi.2021.3090412.

[8] Soleymani, Farzin, et al. "Deep variational clustering framework for self-labeling large-scale medical images." Medical Imaging 2022: Image Processing, 4 Apr. 2022, https://doi.org/10.1117/12.2613331.

7 Team contributions

Juampablo Heras Rivera: Traditional clustering coding, deep learning clustering coding, segmentation model coding, Plotting graphs during data analysis, problem formulation, literature review, writing up the report, coming up with the algorithm, running tests, tabulating final results, accessing the data, preliminary data analysis
Jonathan Friesen: Traditional clustering coding, deep learning clustering coding, Plotting graphs, problem formulation / literature review, writing up the report, coming up with the algorithm, running tests, coding up the algorithm, tabulating final results, accessing the data, preliminary data analysis
Nirmal Kadirkamanathan: Traditional clustering methods, writing up the report, Plotting graphs, tabulating final results
Eric Gibson: