An Analytical Framework of Tissue-patch clustering for Quantifying Phenotypes of Whole Slide Images

Wisuwat Sunhem

Submitted in fulfilment of the requirements for the
Degree of Master of Science by Research

School of Computing Science
College of Science and Engineering
University of Glasgow

July 2022

Abstract

Histopathology is considered the most practical diagnostic method for patient with early stage cancer. This is because at the very first pre-screening, patient's tissue samples are delivered to pathologist for examining evidence of cancer. Computational scientists aid pathologist by heavily producing research on machine learning-based morphological pattern recognition of tissue image. Many data modelling investigations on histopathology have been conducted in supervised manner and some of them were further employed in real-life clinical diagnosis. This study proposes an approach to developing clusters of tissue tile. The main aim is to obtain 'high-quality clusters' with respect to phenotypic annotations. In order to achieve this goal, two colorectal datasets namely 100k-nct and TCGA-COAD are experimented, one of which is directly annotated with tissue type, and other dataset is annotated through derivation from patient metadata, quiescent status. Four main independent variables were explored in this study (i) feature extraction by Resnet50, InceptionV3, VGG16 and an unsupervised generative model, PathologyGAN. (ii) feature space transformer including original feature, 3D-PCA feature and 3D-UMAP feature and (iii) clustering algorithms namely Gaussian Mixture Model and Hierarchical clustering and their primary hyper-parameters. As a result, Resnet50 empowered by UMAP outperformed the most in clustering tissue type on 100k-nct dataset at v-measure of 0.74. The other dataset of which quiescent status is derived from patients encountered nearly zero in v-measure. However, clustering this quiescence-based dataset on 3D-UMAP PathologyGAN yielded far higher V-measure than the rest of cluster configurations and illustrates ability to capture quiescence-related phenotype through visualisation.

Keywords: phenotype cluster, deep learning, generative adversarial model, tumour tissue, manifold learning, dimension reduction.

Contents

Abstract **i**

Acknowledgements **viii**

Declaration **ix**

1 Introduction **1**

1.1 Research aim 3

1.2 Research Contribution 3

1.3 Research Questions 4

1.4 Thesis Statement 4

1.5 Thesis Outline 5

2 Background **6**

2.1 Literature Review 6

2.1.1 Machine learning in digital pathology 6

2.1.2 Image representation 7

2.1.3 Clustering Framework 8

2.1.4 Clustering Quality Measurement 9

2.2 Image representation 11

2.2.1 PathologyGAN 11

2.2.2 VGG-16 14

2.2.3 RESNET-50 16

2.2.4 InceptionV3 20

2.3 Dimension Reduction 22

2.3.1 Principal Components Analysis (PCA) 22

2.3.2 Uniform Manifold Approximation and Projection (UMAP) 25

2.4 Clustering Algorithms 27

2.4.1 Hierarchical-based Clustering (HC) 28

2.4.2 Gaussian Mixture Model (GMM) 30

2.5 Cluster Quality Assessment 35

2.5.1 Homogeneity 35
2.5.2 Completeness 36
2.5.3 V-Measure 36
2.5.4 Silhouette 37

3 Experimental Framework Design 38
3.1 Datasets 38
3.1.1 100,000 histological images of human colorectal cancer and healthy tissue (100k-nct) 38
3.1.2 The cancer genome atlas colon adenocarcinoma (TCGA-COAD) 39
3.2 Experiment setting 42
3.2.1 Sub-tissue patch representation extraction 42
3.2.2 Dimension reduction or feature transformation 43
3.2.3 Splitting train, validation, test 43
3.2.4 Clustering configuration 44
3.2.5 Cluster quality evaluation 45

4 Result and Discussion 46
4.1 Reduced Dimension Feature Space Visualisation 46
4.2 Optimal number of cluster suggested silhouette score 47
4.2.1 Silhouette score on 100k-nct 47
4.2.2 Silhouette score on TCGA-COAD 48
4.3 Cluster Quality Evaluation 51
4.3.1 100k-nct cluster quality evaluation 51
4.3.2 TCGA-COAD cluster quality evaluation 53
4.4 Further interpretation 55
4.4.1 Discussion on clustering algorithm 56
4.4.2 Dimension reduction or feature transformation 56
4.4.3 Image representation 57
4.4.4 Explore what inside each and between cluster 59

5 Conclusion 63
5.1 Summary of thesis contributions and experiments 63
5.2 Response to research questions 63
5.3 Limitations and suggested future research works 64

Bibliography 65

List of Tables

3.1 Tissue type description 39
3.2 TCGA-COAD's WSIs quiescence labels description 40

4.1 Summarised experimental result 59

List of Figures

2.1 Generative Adversarial Networks (GANs) architecture 11

2.2 The grid pictures on the left belong to the Relativistic Average Discriminator model, while the right relate to the Hinge loss model. 12

2.3 High-level architecture of PathologGAN 13

2.4 Decomposing larger filters into the smaller filters used in VGG-16 14

2.5 Different configurations of VGG 15

2.6 The architecture of VGG16 15

2.7 A residual neural network in its canonical form. A layer l-1 is skipped over activation from layer l-2. 16

2.8 A building block of residual learning 17

2.9 RestNet high-level architecture 18

2.10 ResNet-50 Architecture 18

2.11 A deeper residual function F for ImageNet. Left: a building block (on 56×56 feature maps) for ResNet34. Right: a “bottleneck” building block for ResNet-50/101/152 19

2.12 InceptionV1 Architecture 20

2.13 InceptionV3 Architecture 21

2.14 (Left) The original 3-dimensional dataset. The red, blue, green arrows are the direction of the first, second, and third principal components. (Right) Scatterplot after PCA reduced from 3-dimensions to 2-dimensions 22

2.15 An example of 2D PCA on iris dataset 24

2.16 Low dimensional simplices. It is constructed from the convex hull of k+1 points. It is a line segment between two zero simplices. It is a triangle with three 1 simplices as “faces”. 25

2.17 A simplicial complex process. (above left) initial noise sine wave data points. (above right) a basic open cover of the dataset. (below) A simplicial complex built from a dataset of a noisy sine wave. 26

2.18 Representation of agglomerative and divisive approach 29

2.19 Dendrogram with clusters marked 30

2.20 Mixture components with data points visualisation 30

2.21 Gaussian Distribution (Normal distribution) 31
2.22 Gaussian distribution curve in difference mean and variance 32
2.23 Expectation-Maximisation Algorithm 34
2.24 Illustration of the elements and its distant from each cluster when computation. 37

3.1 Samples from the NCT-CRC-HE-100k dataset 38
3.2 Number of tissues type label on NCT-CRC-HE-100k 39
3.3 Whole slide image of TCGA-CZ-5467 case id 40
3.4 224x244 pixel sub tile samples from WSI of TCGA-G4-6317 case id 40
3.5 number of different quiescence labels 41
3.6 Ratio of tiles annotated by different quiescence labels and after filtered out undefined label 41
3.7 Ratio of tiles annotated by different quiescence labels 44
3.8 Ratio of tiles annotated by different quiescence labels without label 2 44

4.1 3D PCA (top) and 3D UMAP (below) representation in NCT-100k and labelled with tissue type from PathologyGAN, ResNet50, InceptionV3 and VGG16 respectively. 47
4.2 3D PCA (top) and 3D UMAP (below) representation in TCGA-COAD and labelled with quiescence label from PathologyGAN, RESNET50, InceptionV3 and VGG16 respectively. 48
4.3 Silhouette score of 100k-nct trained by GMM and HC. 49
4.4 Highest Silhouette score from GMM and HC clusters on 100k-nct. 50
4.5 Number of clusters suggested by Silhouette score on 100k-nct. 50
4.6 Suggest number of cluster by Silhouette score in GMM and HC on 100k-nct on 3D UMAP 50
4.7 Silhouette score of clusters trained by Gaussian Mixture Models and Hierarchical clustering on TCGA-COAD. 51
4.8 Highest Silhouette score from GMM and HC clusters on TCGA-COAD. 52
4.9 Number of clusters suggested by Silhouette score on TGCA-COAD. 52
4.10 Suggest number of cluster by Silhouette score in Gaussian Mixture Models (GMM) and Hierarchical clustering (HC) on TCGA-COAD on 3D UMAP . . . 53
4.11 Extrinsic scores from GMM of different representations on 100k-nct 53
4.12 Extrinsic scores from HC of different representations on 100k-nct 54
4.13 Homogeneity, Completeness and V-measure of GMM and HC clusters suggested by Silhouette score on 100k-nct 55
4.14 V-measure of clusters trained by GMM and HC, which number of clusters suggested by Silhouette Score 56
4.15 V-measure score of different representations on TCGA-COAD 57

4.16 Homogeneity, Completeness and V-measure of GMM and HC clusters suggested by Silhouette score on TCGA-COAD 58
4.17 100k-nct Dendrogram linkage of RESNET 50 UMAP from (optimal silhoutte) GMM clustering and counts of samples under each node 60
4.18 TCGA-COAD Dendrogram linkage of RESNET 50 UMAP (optimal silhoutte) on GMM and HC clustering and counts of samples under each node 61
4.19 TCGA-COAD Dendrogram linkage of pathologyGAN UMAP (best V-measure) on GMM clustering and counts of samples under each node 61
4.20 Sub-tile cluster assignments from GMM pathologyGAN 3dUMAP (Highest V-measure) 62

Acknowledgements

This work has been fully supported by my supervisors: Dr. Ke Yuan. I also would like to thank my second supervisor Dr. Bjørn Sand Jensen for comments that greatly improved this thesis. Many thanks to my colleague Adalberto Claudio Quiros who has helped provide guidance and expertise in machine learning for digital pathology. Finally, a huge thank to my lovely family and friends.

Declaration

The work leading to Master Thesis has been conducted in the school of Computing science. This work was supervised by Dr. Ke Yuan and Dr. Bjørn Sand Jensen from Computing science school.

Except chapters 1, 2, and partial content of chapter 3 which contain introductory material, all conduction in this thesis was delivered by the author unless explicitly stated otherwise.

Chapter 1

Introduction

With the growing interest in data driven approaches, enormous data is being produced and actively collected via digital platforms running worldwide. Needless to say, data cannot be fully utilised without an effective analysis in order to either be monetised or provide social benefits. Model requiring supervised learning is the most common branch of data modelling. It heavily relies on data annotation which is becoming impractical due to the exponential growth in data generation. Another main issue is the fact that supervisory bias, in which labelled samples influence model learning process, can misguide the analysis because its main objective is to maximise the predictive power. This issue is to be taken into account when interpreting data under certain sensitive domains. Clinical data science is undeniably considered one of the most sensitive subject areas as blackbox methods and inaccurate interpretation are crucially serious [1].

Unsupervised learning is often utilized as a solution when concerns regarding supervisory learning become increasingly significant. Clustering is the most widely used method in unsupervised machine learning. It is widely employed to solve data-driven problems when labels are unavailable. Technically, clustering is objected to find a way of grouping samples of which their similarities are high in value on a defined feature space. The mechanism yields an inclusive information of a label-free dataset. The resulting summary provides insight into data that has a complex original structure and may be more challenging for further analysis.

Cancer research is one of the most popularly used in the sector of health associated technology development. Advanced DNA sequencing has been constantly developed with the ambition of early stage cancer detection for patients with developing cancer. [2] The sequencing technology derives a patient's genomic profile from the traversal across the DNA. Indeed, it helps save many lives. Unfortunately, DNA sequencing is not employed at the entry point of cancer diagnosis. The common practice delivered to new patients is to examine tissue samples through microscope. [3] The examination of patient tissue samples using a microscope is formally known as Pathology. This process allows for the early detection of cancers. However, pathology requires highly skilled pathologists to manually examine sectioned tissues on a case-by-case basis. It's a human laborious task that causes fatigue so thus human error is inevitable. Another crucial challenge is that individual tumours are highly identical. The issue obstructs pathologists from achieving a definite conclusion on tumour patterns.

Digital Pathology aids pathologists in tissue interrogation. It employs computational methods to speed up the histopathological process. This is known as "computational pathology". [4] In terms of digital image processing, Whole Slide Image (WSI) is considered a huge file that causes a significant burden to computational units. It's normally gigapixel in size. [5] To curb this issue, a WSI is better split into a set of smaller equal-sized tiles for further processes. Although individual WSI is unique, each title, which together constitute a unique WSI, represents fundamental visual elements shared across different WSIs. Hence, abundances of similar tiles from two WSIs can be features for determining how similar the two WSIs are.

Clustering is a well-established and efficient computational method for grouping data instances based on their similarities within a defined feature space. When applied to tissue tiles, this process results in the creation of "phenotype clusters." The cluster representatives, such as centroids, serve as a catalog of phenotypic information, similar to a catalog of gene expression. By analyzing the abundance of phenotypic clusters in a Whole Slide Image (WSI), a patient's phenotype profile can be constructed. This information can then be used by histopathological experts to identify known and unknown visual features that are related to the patient's genomic profile and clinical conditions. The importance of clustering tissue tiles lies in the ability to effectively categorize and analyze complex pathological data, leading to a better understanding of disease progression and patient outcomes. Also, there is the feasibility of using phenotype clusters in supervised learning by providing additional information about the relationships between the features and the labels, potentially leading to improved results in digital pathology.

In order to ensure that resulting clusters are innate references, cluster training and its configurations are expected to proceed without external annotations. This limitation puts a considerable challenge into the process of obtaining high quality clusters. Two undeniable questions are the most suitable clustering algorithm and the correct number of clusters that should be used for a working dataset. Since data can be represented in various ways, choosing an appropriate feature space for a dataset is impactful as cluster quality is highly sensitive to different feature sets [6]. The sensitivity is also extended to data preprocessing steps including but not limited to dimension reduction. Linear and non-linear transformation could extremely affect clustering depending upon the assumption of how data is embedded in the original feature space and of course that of clustering itself [7]. None of the above configurations are ever found without an effective mathematical measurement of cluster quality. The scoring can be positively or negatively reflective to how well the data samples are grouped based upon statistical assumption. However, external labels cannot be imposed during model selections for the phenotype clusters to be unbiased and deprived from any supervisory objectives. Thus, only intrinsic scores are mainly valid for seeking hyperparameter configurations. Extrinsic scores which require external annotations can be further used in result analyses

In this thesis, two popular clustering algorithms, namely Gaussian Mixture Model (GMM) as a sophisticated partition-based clustering algorithm and Hierarchical Clustering (HC) as an instance-based clustering algorithm, are applied to two different colorectal image datasets. One dataset, called 100k-nct, has tissue type labels directly defined, while the other, COAD, has labels derived from patient level information. To perform clustering, features are extracted from the data samples using several potential deep Convolutional Neural Network (CNN) models, including PathologyGAN, ResNet50, InceptionV3, and VGG16. In addition to the original feature representations, the effects of linear and non-linear feature transformations, implemented through Principal Component Analysis (PCA) and Uniform Manifold Approximation and Projection (UMAP), are also investigated. All of these components are organized and implemented as a phenotypic cluster analytical framework with the goal of being applicable to Whole Slide Image (WSI) datasets, such as colorectal cancer.

1.1 Research aim

This research has three main aim:

1. To develop a robust clustering framework that yields high quality clusters of sub-tissues.
2. The individual phenotype clusters recognise tissue visual characteristics without supervisory biases.
3. Cluster assignment abundances of tissue tiles can reflect patient molecular or clinical profiles.

1.2 Research Contribution

The contribution of this research lies in investigating a potential framework that generates high quality clusters of tissue-patches of WSI. The abundances of cluster membership are expected to represent the phenotype of WSI.

1.3 Research Questions

To work towards our research aims, we need to answer the following research questions:

1. What components are required in a clustering framework in order to differentiate types of tissue tiles without the associated labels provided during cluster training?

2. What patient clinical or molecular attributes can be derived from cluster membership of a WSI?

3. What measure metric is capable of evaluating the framework configuration that could produce the high quality cluster?

4. What is the effect on the high quality clusters from different feature representations ?

5. Which kind of feature transformation has the potential to help improve fabricating the high quality cluster?

1.4 Thesis Statement

The primary purpose of this thesis is to develop a comprehensive clustering framework for tissue tiles concerning the capability of quantifying the phenotype of whole slide images (WSIs). As the assignment of clusters for each tile represents a phenotypic element, the assigned tiles can be quantitated to potentially define a patient's histopathological phenotype profile and describe their cancer status. The main functionality of this framework is to seek optimality in clustering configurations, e.g., choices of representation extraction, ways to clusterable features, clustering algorithm and the related parameters based on a given tissue dataset. Both intrinsic evaluation and the annotation-required extrinsic evaluation will be employed to reach the ideal tile-based clusters. The intrinsic methods are free of bias from any particular objective, while extrinsic ones illustrate the applicability in diagnostic usages. In this study,The sub-tissue images to train the model stem from patients with colorectal cancers.

1.5 Thesis Outline

This thesis is composed of the following chapters:

- **Chapter 2 Background:** This chapter begins with defining the tumour-tile based analysis and unsupervised machine learning technique to address in histopathology field by reviewing the literature. Then it provides a background on describing various feature extraction strategies to represent tissue-patch, data preprocessing to robust representative features on images such as dimension reduction, how fundamental clustering algorithms function, and how to assess high-quality clusters by numerous particular metrics.

- **Chapter 3 Experimental Framework design:** This chapter describes the two colorectal cancer datasets and the design of experiments to assemble the framework of clustering on sub-tissue tiles in diverse elements. The data section will describe the selected WSI dataset with attributes, metadata and labels. In the experiments design section, we describe an experiment's design for obtaining high-quality clusters generated from a clustering approach. Several tissue representations are introduced to find out the best for unsupervised computational histopathology. Also, linear and non-linear dimension reduction is applied with the purpose of undercovering a more clusterable feature space. Prototype-based clustering and hierarchical-based clustering are then explored on those configurations and its performance evaluated by quality cluster assessment techniques.

- **Chapter 4 Result and discussion:** This chapter describes the experiment outcomes and manifests optimal unsupervised clustering framework configuration toward the quality clusters assessment. And also interpret and explain the significance of results of the experiment.

- **Chapter 5 Conclusion:** This chapter describes the thesis's contributions, summarises the experiments, responds to the research questions, and discusses limitations, suggestions, and future work.

Chapter 2

Background

This research aims to develop a comprehensive clustering framework of tissue patches for quantifying the phenotype of Whole Slide Image. The main objective of this framework is to determine the appropriate clustering configuration for a given tissue dataset, including representation extraction, dimension transformation, clustering algorithm, and related parameters.

This chapter begins by defining tumour-tile based analysis and unsupervised machine learning for the histopathology discipline through a literature review. Then it provides background information on describing various strategies to represent tissue-patch, data preprocessing to robust representative features on images such as dimension reduction, how fundamental clustering algorithms function, and how to evaluate high-quality clusters in a number of specific measurements.

2.1 Literature Review

2.1.1 Machine learning in digital pathology

In digital pathology, image processing and machine learning are widely employed in collaboration to recognise morphological patterns appearing in digital WSI [8, 9]. As a result, it helps facilitate practitioners in diagnostic purposes. However, very prior art of machine learning requires the intervention of knowledge, particularly in feature extraction and model training framework design, which are unique across digital pathology tasks with different objectives. Instead of jumping straight into execution of data wrangling, clinical data scientists are mandated to spend their foremost effort on understanding the relevant fundamental knowledge. Thank to the arrival of deep learning, the required effort to understand a new task-related foundation is reduced when performing data modelling [10]. It has permitted the acceleration of technology development in digital pathology.

Convolutional Neural Network (CNN) is a well-known class of deep learning for image analyses. It was born with the ability to learn how to extract useful visual features from a large sample-size dataset automatically. This full capacity of CNN was widely recognised after the introduction of ImageNet dataset [11]. The dataset of million images of several classes has been the most popular benchmarking image dataset for a decade. Highly potential knowledge related to Computer Vision that many CNN architectures have learnt from ImageNet is widely transferable to solve many computer vision problems, in which medical analysis is included [12]. The success of transfer learning on ImageNet influences the development of numerous CNN architectures to work well on input images with a resolution below the square of a thousand [13]. As mentioned in the introduction, the resolution of a WSI is typically around 100,000 x 100,000 pixels, which is far beyond the size of input images expected by many mature CNN architectures. Therefore, victorious works on WSI examination were typically done by splitting WSI into smaller equal-sized patches and then aggregating analytical results from those patches instead of using the entire WSI [14–16].

Most previous research works on digital pathology were conducted under the framework of supervised learning. In multiple papers, supervised learning-based research focuses range from tackling fundamental image processing tasks such as cell detection and segmentation [17, 18] to the most prevalent supervised learning problem, classification of cell and tissue types [19, 20]. Some research results derived from the data modelling on histopathology data were further employed to clinical diagnosis [21]. Nonetheless, knowledge discovered by approaches with supervisory bias is not interchangeable across different tasks but usually valid for a specific goal. Adaptation of a supervised learning-based model from one task to another demands an extensive fine-tuning [22]. That is because the learned features are not proven to be generalised. Those approaches require supervisory bias to achieve a specific goal. Hence, the findings from a supervised learning-based experiment cannot be claimed as the progress of general visual element discoveries towards an alternative to DNA sequencing technology.

2.1.2 Image representation

It is undeniable that practical features play an impactful role in obtaining a high-quality machine learning model. No matter how sophisticated machine learning algorithm is employed, it cannot compensate for the quality of features in order to succeed in a data modelling task [23]. Selecting a feature set is very sensitive and directly influences the quality of model training. Prior to the use of deep learning techniques, the process of selecting features extensively was time-consuming and required a significant amount of effort, but it was necessary for achieving a high-quality model.

The arrival of deep learning constituted a highly efficient solution to the the burden of exhaustive search in the traditional feature selection, called representation learning. It simultaneously creates feature extractor while training a deep classifier. The feature extractor is guided concerning a specific objective function, either supervisory or nonsupervisory [24]. Another essential benefit of deep representation learning is the ability to be transferable to different domains which only requires soft tuning.

Training a supervised deep learning model from scratch is inefficient, especially in image classification, which takes a long time to converge and risks overfitting [25]. Transfer learning is introduced as an effective solution this weakness with the enormous success. Multiple research works [26–28] used initialised deep neural network architecture parameters, fitted by ImageNet, the most common benchmarking image classification dataset. By removing the top fully-connected layers of a deep classifier, the remaining parts work as a feature extractor. Regarding famous CNN architectures to be examined as the potential feature extractors, ResNet50, InceptionV3 and VGG16 as the toplist of deep architectures mentioned in all those research papers.

Deep feature extraction is not limited to supervised learning fashions. There are several attempts to that in unsupervised ways. One of the first unsupervised feature learning techniques is Autoencoder (AE) [29]. It is widely recognized as a non-linear technique for reducing dimensionality that can be applied to both structured and unstructured data, including images. Due to no constraint applied during training vanilla AEs, it is hard to ensure that the feature space lies on a valid assumption especially for a complex but small dataset. Variational Autoencoder (VAE) was introduced to solve this issue by constraints imposed into feature space [30]. Dissimilar to supervised learning in which accuracy defends its performance, unsupervised feature learning suffers from no exact scoring available which leads to the demand for trust by visualising the mechanisms inside the function [31]. Fortunately, Ian Goodfellow proposed a new training framework for neural networks via an adversarial objective, Generative Adversarial Network (GAN). Throughout the adversarial technique, one of the pairs of models, the generative one, can illustrate what latent space is representing. Moreover, my colleague and I also published a recent research work presented a GAN architecture, called PathologyGAN [32], and provided promising proof of success in using GAN for tissue slide's feature extraction.

2.1.3 Clustering Framework

Clustering can be roughly divided into two main categories including hierarchical clustering and partitional clustering. Hierarchical-based clustering allows a sample belongs to more than one sub-cluster based on a different level of hierarchy [33]. In contrast, partition-based or prototype-based clustering is non-overlapping. Guassian Mixture Model (GMMs) and K-means are the well-known representatives of partitional clustering. GMM is considered as generalised version of K-means while K-means is seen as the least complex [37]. Thus, when considering which type of clustering is to be used, GMM is the more reasonable to be benchmarked against hierarchical clustering (HC).

Finding the most appropriate clustering algorithm for a dataset and decided to depend solely on its original feature space, which is normally high in dimension, is not risk-free. Running a complex clustering algorithm on a huge dataset can take forever. The dataset of sub-tissues of gigapixel WSIs is to be considered. It is also unavoidably true that curse of dimensionality is far more harmful to clustering performance than the classification counterpart [38]. This is because curse of dimensionality directly affects the process related to distance metrics. Compared to many classification algorithms, clustering is more reliant on measurable distance between samples.

Luckily, significant works on effective dimension reduction are actively produced to improve performance of feature embedding and in due course the clustering performance. It ranges from the simplest approach which is Principal Component Analysis (PCA) to the preservation of global structure and local structure by manifold learning such as UMAP (Uniform Manifold Approximation and Projection). There is no clear evidence of which one is better. The study in [39] shows that only PCA to preserve global structure with respect to valid linear interpolation is more than sufficient to obtain a high quality cluster. On the other hand, [40] illustrates a big improvement on clustering through recovering the underlying manifold during dimension reduction prior to the execution of clustering. These two techniques will be extensively explored towards the high quality cluster of tissue patches.

2.1.4 Clustering Quality Measurement

Clustering's ultimate goal is to assign data samples into "correct" clusters. the word "correct" is variable enough to show an immediate challenge. Different clustering algorithms behave differently regardless of several critical parameters of each to be properly adjusted in order to obtain optimality. Cluster training is executed by comforming a cluster pre-defined structure including selected feature space, number of clusters, distance metric [41]. Because the goodness of clustering is subject to individual perceptions so thus cluster algorithms are destined to provide a variety of clustering outcomes.

As mentioned previously, performance validation of clustering is far more challenging than that of supervised learning such as classification. It is due to the fact that, during model training and validation, external labels are not allowed to influence the clustering framework. Although cluster quality indices consist of intrinsic and extrinsic methods, the latter one which requires sample labels to calculate can be used only in performance reporting and visualisation. Intrinsic score is deemed to reflect statistical properties of clusters including cluster compactness and intercluster density [42]. There are attempts to propose rigid clustering approaches across wide-ranging domains. Silhouette scores are among the most popular intrinsic methods. The cluster quality represented by the score is utilised for different purposes. The most common functionality of the score is to obtain optimal configuration e.g. number of clusters [42,43]. However, Ciortan, M et al took on silhouette score to directly analyse clustering performance [44].

Once a definitive clustering model is generated, external annotation is usually used to evaluate clustering quality. Despite the belief that labels are restricted not to influence model selection, it is highly recommended to investigate how clustering corresponds to those cluster labels [45]. In literature, extrinsic measures work alongside intrinsic counterparts in a variety of collaborative approaches. For example, two research works used extrinsic and intrinsic measures together to test the performance of models. One of the research piece presented a finding indicating that both measures highly correlate one another while the other research founded no correlation at all. However, there is a study which employs the two method for two different purposes. In the research, silhouette scoring is used to define cluster configurations and only extrinsic scores are for performance evaluation

As this research's aim is to find a robust clustering framework for tissue tiles, specific genomic or clinical annotations can not be relied upon. Figuring out how intrinsic metrics relate to the evaluation by extrinsic ones is a main contribution of this thesis. Linear and non-linear transformations such as dimension reduction could involve in discovering the relationship between an intrinsic score and an extrinsic one. A research study [40] proposed that a manifold learning could come into play in improving cluster quality and results in higher quality clusters based on both intercluster properties and external annotation correspondences.

2.2 Image representation

2.2.1 PathologyGAN

PathologyGANs [32] is an approach to the use of machine learning in digital pathology, using Generative Adversarial Networks (GANs) to learn cancer tissue representations. GANs are capable of learning high-fidelity and varied representations of data from a target distribution. This generative model captures visual characteristics of whole tissue architectures and provides an interpretable latent space, enhancing the capacity of generative models to capture phenotypic representations.

The generator shows that the distinct region of the learnt feature space are associated with certain tissue characteristics. [54] and enable the creation of tissue representations without the need for costly labels and representations that are not only correlated with a predicted result , but also on the similarities between the characteristics of tissue samples.

Generative Adversarial Network (GANs)

Generative Adversarial Network (GAN) is a class of machine learning framework powered by deep learning methods proposed by Goodfellow l. et al [53]. GAN is categorised in an unsupervised learning approach that is able to automatically learn the patterns and output the examples based on the fed input data. GANs have achieved increasing popularity as a generative model applicable to a variety of disciplines. GANs are an exciting and rapidly advancing field that meet the promise of generative models by generating realistic examples across a variety of subject areas, most notably in image-to-image translation tasks, as well as in generating photorealistic images of objects, scenes, and people.

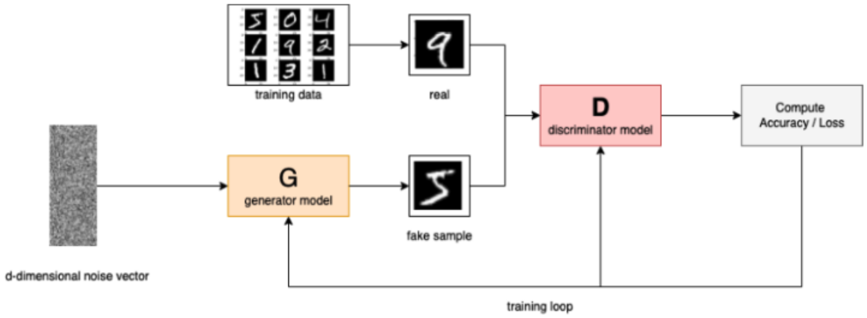


Figure 2.1: Generative Adversarial Networks (GANs) architecture

The GAN architecture is composed of two sub-models: a generator model for creating new instances and a discriminator model for identifying whether each instance is genuinely derived from domain or fabricated by the generator model

The multilayer perceptrons of G generator and D discriminator are trained concurrently. As shown in equation2.1 $G(z)$ generates a batch of samples by mapping random noise, $z \sim p_z(z)$, which, together with real-world instances from the domain, $x \sim p_x(x)$, are sent to $D(x)$, which classifies them as single scalar output (genuine or fictitious). D is then updated to improve its ability to distinguish between genuine and fictitious samples in the subsequent round. More significantly, G is modified depending on how effectively the produced samples confused the discriminator. The goal of a GANs is to find the equilibrium in the min-max problem. Describe in equation 2.1

$$min_G max_D V(D,G) = E_{x\sim p_{(x)}}[logD(x)] + E_{z\sim p_z(z)}[log(1-D(G(z)))] \tag{2.1}$$

PathologyGAN Architecture

PathologyGAN used BigGAN [55] as a baseline architecture and introduced changes which empirically improved the Fréchet Inception Distance (FID) and the structure of the latent space.

The reason for choosing BigGAN, the model has been shown to be a successful GAN in replicating datasets with a diverse number of classes and large amounts of samples. From theoretically, the model will be able to learn and replicate the diverse tissue phenotypes contained in WSI, being able to handle the large amount of tiles/patches resulting from diving the WSIs. PathologyGAN followed the same architecture as BigGAN by employing Spectral Normalisation in both generator and discriminator, self-attention layers, and adding orthogonal initialization and regularisation.

Moreover, pathologyGAN used the Relative Average Discriminator [56] instead of Hing loss as GAN's objective. where the discriminator's goal is to estimate the probability of the real data being more realistic than the fake. From the experiments, they find that changing the GAN's objective function makes model convergence faster and produce higher quality images, capturing the morphological structure of the tissue.

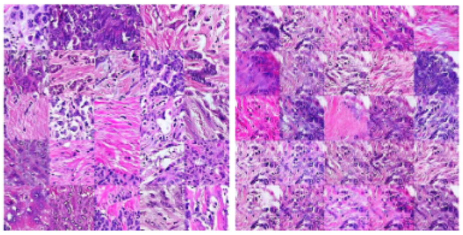


Figure 2.2: The grid pictures on the left belong to the Relativistic Average Discriminator model, while the right relate to the Hinge loss model.

The discriminator and generator loss functions are defined as follows: Equations 2.2, 2.3, where P is the distribution of actual data, Q denotes the distribution of fictitious data, and $C(x)$ denotes the non-transformed discriminator output or critic:

$$L_{Dis} = -\mathbb{E}_{x_r \sim P}[log(D(x_r))] - \mathbb{E}_{x_f \sim Q}[log(1-D(x_f))], \tag{2.2}$$

$$L_{gen} = -\mathbb{E}_{x_r \sim Q}[log(D(x_f))] - \mathbb{E}_{x_f \sim P}[log(1-D(x_r))], \tag{2.3}$$

When $E_{x_r \sim P}$ denotes the expectation over the distribution of the real data and $E_{x_f \sim Q}$ denotes that over the distribution of fake data. $D(x_r)$ and $D(x_f)$ are the output of the discriminator when it is given a real image and a fake image, respectively.

$$\sim D(x_r) = sigmoid(C(x_r) - \mathbb{E}_{x_f \sim Q} C(x_f)),$$

$$\sim D(x_f) = sigmoid(C(x_f) - \mathbb{E}_{x_r \sim p} C(x_r)),$$

$$w = M(z), z \sim P_z.$$

In encoder loss function, L_Enc, the mean square error between latent vectors w and their reconstruction from generated images $w' = R(G(w))$ as equation 2.4

$$L_{Enc} = \mathbb{E}_{z \sim P_z}\left[\frac{1}{n}\sum_{i=1}^{n}(w_i - w_i')^2\right] \; where \; w' = E(G(w)), w = M(z). \tag{2.4}$$

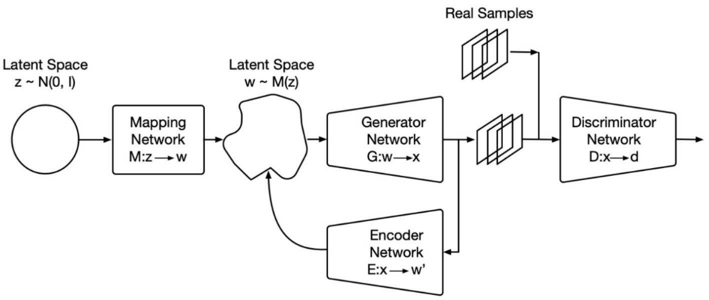


Figure 2.3: High-level architecture of PathologGAN

2.2.2 VGG-16

Karen Simonyan and Andrew Zisserman developed the VGG network concept [60] in 2013 and competed in the 2014 ImageNet Challenge with the resulting model. The name VGG is from the Visual Geometry Group at the University of Oxford, which they were members of. The VGG model was a significant step forward in the effort to assist computers in "seeing" the world. This ability has been honed over many decades in the area of Computer Vision (CV). The VGG model is a significant advancement that paved the way for several others in this field.

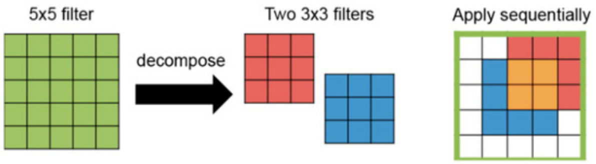


Figure 2.4: Decomposing larger filters into the smaller filters used in VGG-16 [61]

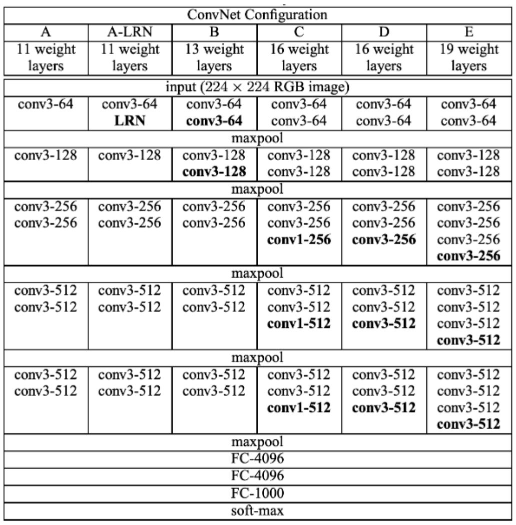
The 3 × 3 filter size is considered to be the smallest that effectively captures the concepts from left to right, top to bottom, and so on. Thus, decreasing the filter size further may have an effect on the model's ability to understand the spatial features of the image. In contrast to the large receptive fields in the first convolutional layer, this model proposes using a very small 3 × 3 receptive field all over the network with a stride of one pixel. The first layer of AlexNet had a receptive field of 11 x 11 with stride 4, whereas ZFNet had a receptive field of 7 x 7 with stride 2. Using a 3 x 3 filter makes VGG stand out. Two subsequences 3 x 3 filters provide a 5 x 5 effective filter. Accordingly, three 3 x 3 filters correspond to a 7 x 7 filter. A combination of many 3 x3 filters can serve as a larger receptive area. Additionally, it significantly decreases the number of weight parameters in the model by lowering the size of the filter.

VGG Architecture

The convolution network receives a fixed-size 224x224 RGB image during training. The only preprocessing is subtracting the mean RGB value from each pixel. 3x3 convolutional layers are used to capture the notions of left/right, up/down, and centre. It is because this size is the smallest size that could capture the direction in the kernel of the filter. The stride of the convolution is fixed at one pixel, and the spatial padding of the convolution layer's input is also set to one pixel to maintain spatial resolution after convolution. Five maximum-pooling layers are added after parts of the convolutional layers to accomplish spatial pooling (not all the convolution layers are followed by max-pooling).

Following a stack of convolutional layers, three Fully-Connected (FC) layers are used: the first two have 4096 channels each, while the third performs numbers of channels, one for each target class (Following Fig 5. There are 1000 channels based on 1000- way ILSVRC classification).

Finally, there is a soft-max layer. In all networks, the configuration of the fully connected layers is identical. And all hidden layers are equipped with the rectification (ReLU) non-linearity.



ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input (224 × 224 RGB image)					
conv3-64	conv3-64 **LRN**	conv3-64 **conv3-64**	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 **conv3-128**	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 **conv1-256**	conv3-256 conv3-256 **conv3-256**	conv3-256 conv3-256 conv3-256 **conv3-256**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

Figure 2.5: Different configurations of VGG [60]

VGG16 Architecture

The more layers in a CNN model, the more complicated functions the model can fit. For an Artificial Neural Network (ANN), adding layers does not always result in improved performance. A VGG Network with more layers, such as VGG20, VGG50, or VGG100. The weights of a neural network are updated by the backpropagation method, which makes slight adjustments to each weight in order to reduce the model's loss with chain-rules. However, when the gradient travels to the initial layers, the value increases with each local gradient. As a consequence, the gradient becomes more tiny, resulting in extremely minor modifications to the initial layers. This results in a significant increase in training time.

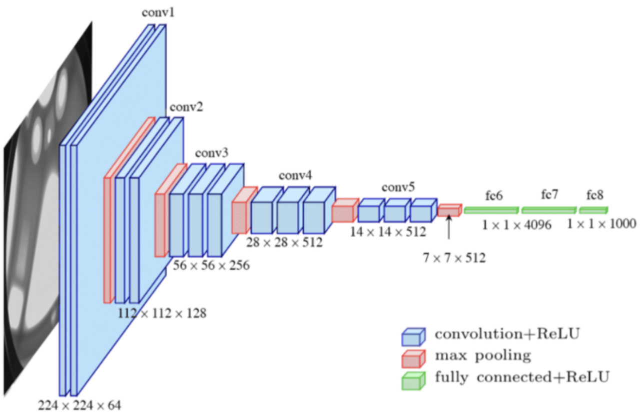


Figure 2.6: The architecture of VGG16 [62]

VGG16 consists of 13 convolutional layers, 5 maximum-pooling layers, and three fully connected layers. As a result, the total number of layers with configurable parameters is 16 (13 convolutional layers and 3 fully connected layers). That is why the model is called VGG16. The first block contains 64 filters; this number is increased in subsequent blocks until it reaches 512. Two fully connected hidden layers and one output layer complete this model. Both fully connected layers have the same number of neurons, 4096. The output layer has numbers of neurons, one for each category according to the task.

2.2.3 RESNET-50

A residual neural network (ResNet) is an artificial neural network(ANN) that employs a skip connection to shortcut to skip over some layers and feature a batch normalisation. A deep residual network framework was proposed “Deep Residual Learning for Image Recognition” [17] in 2016. It came first in ImageNet detection, ImageNet localization, COCO detection, and COCO segmentation in the ILSVRC & COCO competitions of 2015. The model was able to train a network with 152 layers using this method while maintaining a lower complexity than VGGNet. It obtains a top-5 error rate of 3.57 percent, which is higher than the mistake rate experienced by humans on this dataset.

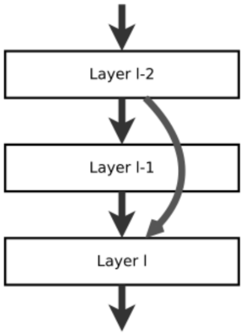


Figure 2.7: A residual neural network in its canonical form. A layer l-1 is skipped over activation from layer l-2.

The reason residual neural networks are able to outperform a regular neural network, there has been a tendency toward going deeper, solving more complicated problems, and improving classification and recognition problems. Training neural networks gets more difficult and the accuracy begins to saturate and subsequently decline.

Deep residual learning

$$H(x) = F(x) + x \tag{2.5}$$

As x denotes the input of the first of these layers, and $H(x)$ identifies the desired underlying mapping. Allowing the stacked nonlinear layers match another mapping of $F(x)$.

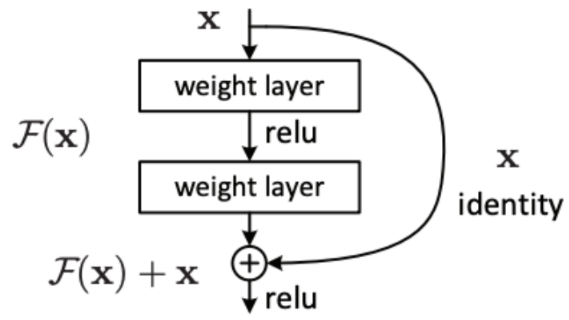

x
weight layer
$\mathcal{F}(\mathbf{x})$
relu
weight layer
x
identity
$\mathcal{F}(\mathbf{x}) + \mathbf{x}$
relu

Figure 2.8: A building block of residual learning

Residual learning is adopted to every few stacked layers. A building block is shown in figure 2.8 define as equation 2.6.

$$y = F(x, w_i) + x \tag{2.6}$$

Where x and y are the input and output vectors of layers considered. The function $F(x, W_i)$ represents the residual mapping to be learned. From figure 2.8, it contain 2 layers and simplify as equation 2.7.

$$F = W_2\sigma(W_1x) \tag{2.7}$$

Which σ denotes ReLu (Skipping the bias to simplify the notation). The shortcut connection is operated by $F + x$ and element-wise addition. And it makes an addition after the second nonlinear adopted. The shortcut connection is not an extra parameter or computational complexity when compared to a plain network. However, the dimension of x and F must be equal in equation 2.7. If it not in the case when changing the input/output channel, performing a linear projection W_s by the the shortcut connection has to employ to matching the dimensions:

$$y = F(x, w_i) + W_sx \tag{2.8}$$

Residual Learning (skip connections) could address both of these issues. By skipping layers during the initial training phases, the network is effectively simplified. This accelerates learning by minimising the influence of vanishing gradients due to the decrease in the number of layers propagating through. As the network gains knowledge of the feature space, it gradually fills up the skipped layers.

RestNet Architecture

The ResNet structure included the insertion of shortcut connections in order to convert a plain network to its residual network counterpart. The plain network was based on VGG neural networks. ResNets have fewer filters and are less sophisticated than VGGNets [60]. Additionally,

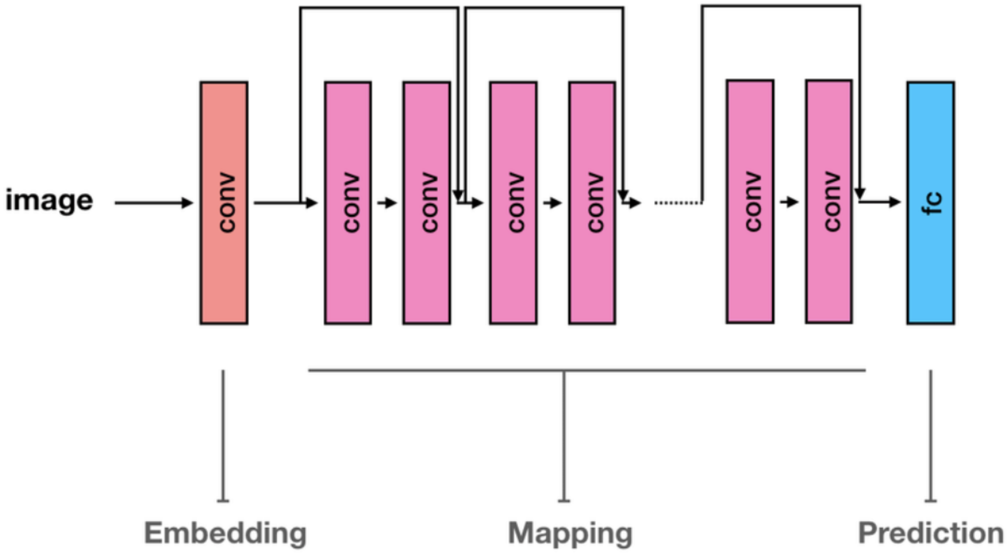


Figure 2.9: RestNet high-level architecture [59]

it followed two straightforward design principles: each layer had the same number of filters for the same output feature map size, and the number of filters was doubled when the output feature map size was half to maintain the time complexity per layer. The shortcut connections were added to this plain network. While the input and output dimensions were the same, the identity shortcuts were directly used.

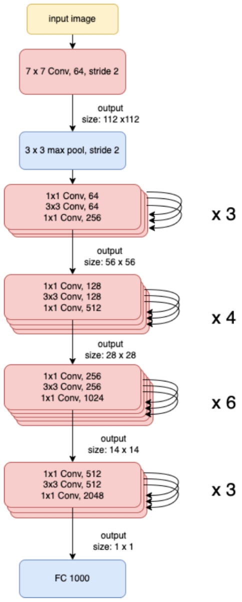


Figure 2.10: ResNet-50 Architecture

ResNet50 is a ResNet version that has 48 Convolutional layers, 1 MaxPool layer, and 1 Average Pool layer. It operates on $3.8x10^9$ floating points. There was a minor change made for ResNet 50 and higher numbers of layers. For the lower number of layers, shortcut connections skipped two layers; now, they skip three levels. Additionally, 1 x 1 convolution layer was added.

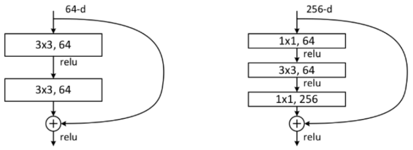


Figure 2.11: A deeper residual function F for ImageNet. Left: a building block (on 56×56 feature maps) for ResNet34. Right: a “bottleneck” building block for ResNet-50/101/152

As Figure 2.11, a convolution with a kernel size of 7 x 7 and 64 distinct kernels, each with a stride size of 2, results in a single layer. Following that, max pooling is added to the network, along with a stride size of 2. Following there comes a 1 x 1, 64 kernel, followed by a 3 x 3, 64 kernel, and finally a 1 x 1, 256 kernel. These three layers are repeated three times in total, giving us 9 layers in this stage. Following that, a kernel of 1 x 1,128 is stacked onto the network, followed by a kernel of 3 x 3,128 and finally a kernel of 1 x 1,512. This phase was performed four times, totaling 12 layers. Following it is a kernel of 1 x 1,256 and two further kernels of 3 x 3,256 and 1 x 1, 1024, which are repeated six times for a total of 18 layers. And then a 1 x 1, 512 kernel was combined with two additional 3 x 3, 512 and 1 x 1, 2048 kernels, which was done three times for a total of nine layers. Following that, the network adds an average pool and concludes with a fully linked layer having 1000 nodes and a softmax function, which results in a single layer.

2.2.4 InceptionV3

Convolutional networks are the core of most modern computer vision technologies. Deep convolutional networks have been popular, delivering significant advances in many benchmarks. While larger models and higher computing costs generally improve quality, computational efficiency and low parameter count are still enabling aspects for use cases like mobile vision and big data. Compared with the architectural simplicity of VGGNet comes at a high cost of evaluating the network demands a lot of processing. However, GoogLeNet's Inception architecture [64] was also built to function well under memory and computational constraints. For example, GoogleNet utilised just 5 million parameters, compared to AlexNet's 60 million. VGGNet also used 3x as many parameters as AlexNet. The InceptionV3 model was designed in strategies to handle growth networks while maximising the efficiency of the extra processing by using factorised convolutions and aggressive regularisation.

InceptionV1 (GoogLeNet)

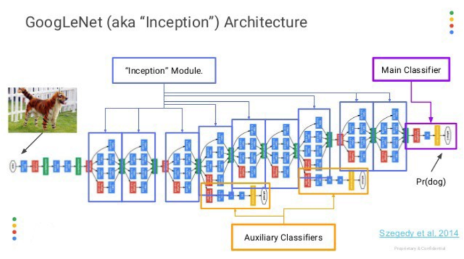


Figure 2.12: InceptionV1 Architecture [65]

InceptionV1 (GoogLeNet) focused on growing the network depth to extract more features and improve the model's learning capabilities. GoogleLeNet is a 22 layer deep network developed using the Inception module. The Inception module's premise was that neurons that extract characteristics should learn together. Earlier convolutional designs varied kernel size to extract optimum features. The InceptionNet design focuses on parallel processing and extraction of several feature maps.

Convolutions 1x1, 3x3, 5x5 and 3x3 max pooling are all performed using the Inception module. Then it constructs the next feature by combining all the processes results. Not all operations, such as pooling or convolution, are executed sequentially. The inception module retrieves various features from each convolution or pooling operation. For example, 1 x1 and 3x3 convolutions provide different information. A single feature map with all characteristics will be created when the separate processes are executed concurrently. This will improve the model's accuracy by focusing on multiple features simultaneously. The output dimension of each extracted feature map will vary according to the varying kernel sizes. In order to make the output dimension of each operation consistent, the individual feature maps are concatenated together using padding.

InceptionV3

InceptionV3 is an improved model version of InceptionV1. The network was optimised in the Inception V3 model for better model adaptability. It is more efficient, has a deeper network, but does not make it slower, and uses Classifiers as regulariser as auxiliary components. The InceptionV3 model has 42 layers and a reduced error rate than the previous version. Three key components of the success of InceptionV3 over the previous series includes

- **Factorization into Smaller Convolutions:** where one large sized convolution is replaced by a number of smaller convolutions through a mathematical function called 'factorisation' , which results in in a relative reduce in number of parameters of 28%.
- **Utility of Auxiliary classifiers:** It is served as a regulariser inside the Inception V3 model architecture. The network with an auxiliary classifier has been proved to be more accurate than the network without one.
- **Efficient Grid Size Reduction:** the modified network will have overall smaller grid size but be compensated by a increasing number of filters.

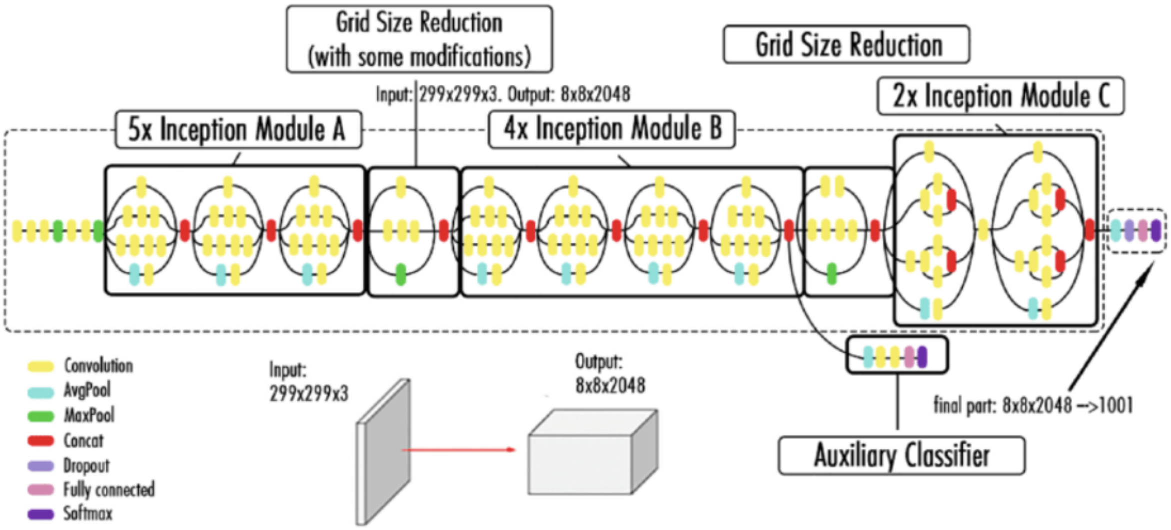


Figure 2.13: InceptionV3 Architecture [66]

2.3 Dimension Reduction

In applications such as image processing [71], computational biology [72] , and clustering [73], high-dimensional datasets are regularly encountered. In molecular biology, for instance, human DNA gene expression profiles generally include hundreds of genes; this is the issue dimension. A common 2D picture in image processing contains 1282 = 16,384 pixels or dimensions. Dimension Reduction aims to reduce the number of features under analysis, each of which is a dimension that partially describes the objects. As additional characteristics are added, the data become very sparse, and the analysis is plagued by the curse of dimensionality. Additionally, smaller data sets are simpler to handle.

2.3.1 Principal Components Analysis (PCA)

Principal component analysis (PCA) is the process of calculating the principal components and using them to execute a change of basis on the data, often keeping just the top few components and rejecting the rest. PCA was introduced in 1901 by Karl Pearson [74]. Principal components of a set of points in a real coordinate space are a sequence of unit vectors, where the i^{th} vector is the direction of the line that best matches the data while remaining orthogonal to the first $i-1$ vectors. In this context, the best-fitting line is one that minimises the average squared distance between the points and the line. Different individual dimensions of the data is linearly uncorrelated with respect to these directions, which create an orthonormal basis. Principal Components Analysis accomplishes dimension reduction through the following steps.

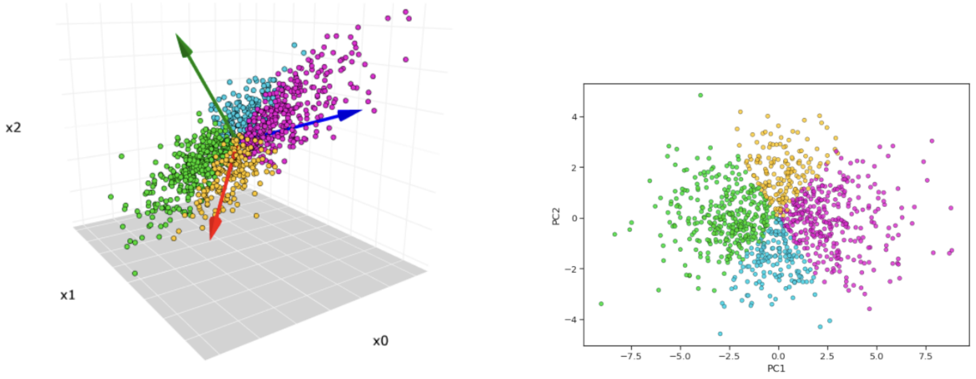


Figure 2.14: (Left) The original 3-dimensional dataset. The red, blue, green arrows are the direction of the first, second, and third principal components. (Right) Scatterplot after PCA reduced from 3-dimensions to 2-dimensions

1. *Standardise the data*

   Generally, the variables composing the dataset will have distinct units and means. This can result in complications such as the calculation yielding extremely huge numbers. To increase the efficiency of the procedure, it is recommended to centre the data at mean 0. This is accomplished by removing the mean from the data and dividing by the standard deviation. This maintains the relationships while ensuring the overall variance equals 1.

2. *Calculate the Covariance Matrix*

   PCA aims to extract the majority of information from a dataset by determining the principal components that maximise the variance between observations. Covariance matrix is a symmetric matrix with rows and columns equal to the number of data dimensions. Calculating the covariance between the pairwise means reveals the degree to which the characteristics or variables vary from each other.

3. *Calculate the Eigenvectors and Eigenvalues of the Covariance Matrix*

   Eigenvectors are vectors that do not change direction when a matrix is transformed. Eigenvalues are scalars that represent the vector's magnitude. The covariance matrix eigenvectors point towards the biggest variance. More variation is explained by a larger Eigenvalue. The greatest Eigenvector corresponds to the first principal component, which explains the most variance, the second largest Eigenvector to the second principal component, etc.

4. *Reduce Dimensionality*

   The principal components are efficient feature combinations that minimise feature overlap. Getting rid of redundant data already helps reduce dimensionality. Given that each new principal component reduces the overall variance explained, we may further reduce dimensionality by deleting the least relevant principal components. Finally, projecting the data from initial feature space to the principal component space is executed. So, we could describe the PCA algorithm, Assuming you have data consisting of a set of observations of p variables, and we wish to reduce the data so that each observation can be represented by L variables, $L < p$. Suppose further that the data is organised as a collection of n data vectors $x_1 \ldots x_n$ with each x_i representing a single grouped observation of the p variables.

$$\mu_j = \frac{1}{n} \sum_{i=1}^{n} x_{ij} \tag{2.9}$$

Firstly, the empirical mean is computed the mean feature value from each j column. So, the result is vectors in $px1$ dimensions.

$$C = \frac{1}{n}\Sigma_i^n = 1(x_i - \mu)(x_i - \mu)^T \tag{2.10}$$

After getting the empirical mean, applying mean subtraction to X to centering the data by minimising mean square error of the mean of data, and then calculating the pxp covariance matrix by conjugate transport operation in variances.

$$Cv_i = \lambda_i v_i \tag{2.11}$$

Next, Computing Eigenvalues λ_i and Eigenvectors v_i of covariance matrix where i is 1 to the number of features q. And we can estimate the high-valued Eigenvectors. Beginning from arranging all Eigenvalues λ_i in descending order and choosing a threshold value θ.

$$(\Sigma_{i=1}^{s}\lambda_i)(\Sigma_{i=1}^{q}\lambda_i)^{-1} \leq \theta \tag{2.12}$$

s is number if high valued λ_i chosen to satisfy the relationship. After that, selecting a subset of the Eigenvectors corresponding to select high valued λ_i.

$$P = V^T x \tag{2.13}$$

To extract low dimensional feature vectors (principal components) from raw feature matrix. V is the matrix of principal components and x is the feature vector. The first column P is the projection of the datapoint onto the first principal component, and the following columns are the following principal components.

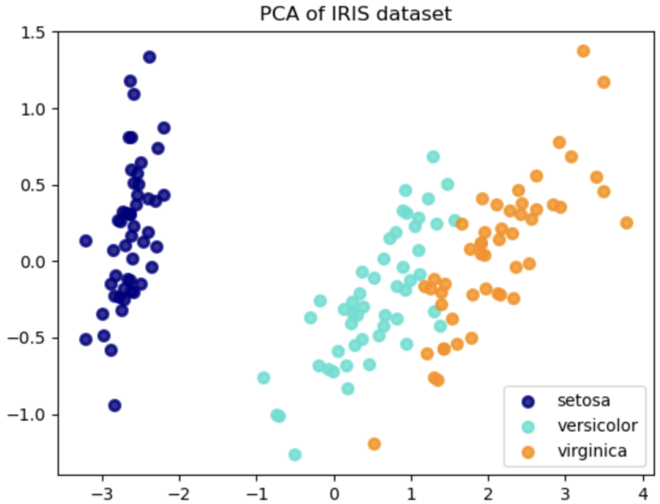


Figure 2.15: An example of 2D PCA on iris dataset

2.3.2 Uniform Manifold Approximation and Projection (UMAP)

UMAP (Uniform Manifold Approximation and Projection) is a dimension reduction algorithm based on manifold learning and topological data analysis. It provides a solid general framework for tackling manifold learning and dimension reduction. The theoretical framework of UMAP is based on Riemannian geometry and algebraic topology. This method was proposed in 2018 by Leland McInnes el et. [75].

Simplicial Complexes and Topological Data

Simplicial complexes are a way to build topological spaces from basic components. This reduces the complexity of dealing with topological spaces's continuous geometry to basic combinatorics and counting. In general, and in dimension reduction in particular, this strategy of controlling geometry and topology is crucial. The first step is to supply some "simplices". A simplex is a geometrically simple k-dimensional object. A k-simplex is produced by obtaining the convex hull of $k+1$ independent points. That makes generalisation to arbitrary dimensions trivial. Constructing a simplicial complex. Simplices can give building blocks. A simplicial complex is a grouping of simplices joined by faces. This method could create a broad class of topological spaces by glueing simplices of various dimensions together.

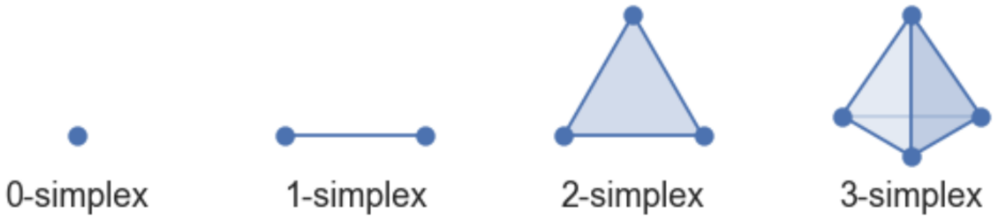


Figure 2.16: Low dimensional simplices. It is constructed from the convex hull of k+1 points. It is a line segment between two zero simplices. It is a triangle with three 1 simplices as "faces". [75]

In order to obtain the topological space, a strategy is applied to a finite amount of data samples. If the data samples originate from some underlying topological space, we must construct an open cover of that space in order to discover its topology. If our data is in metric space (we can measure distance between points), another way to replicate an open cover is to create balls of a predefined radius around each data point. As demonstrated in figure 2.17, a simplicial complex built from a dataset of a noisy sine wave.

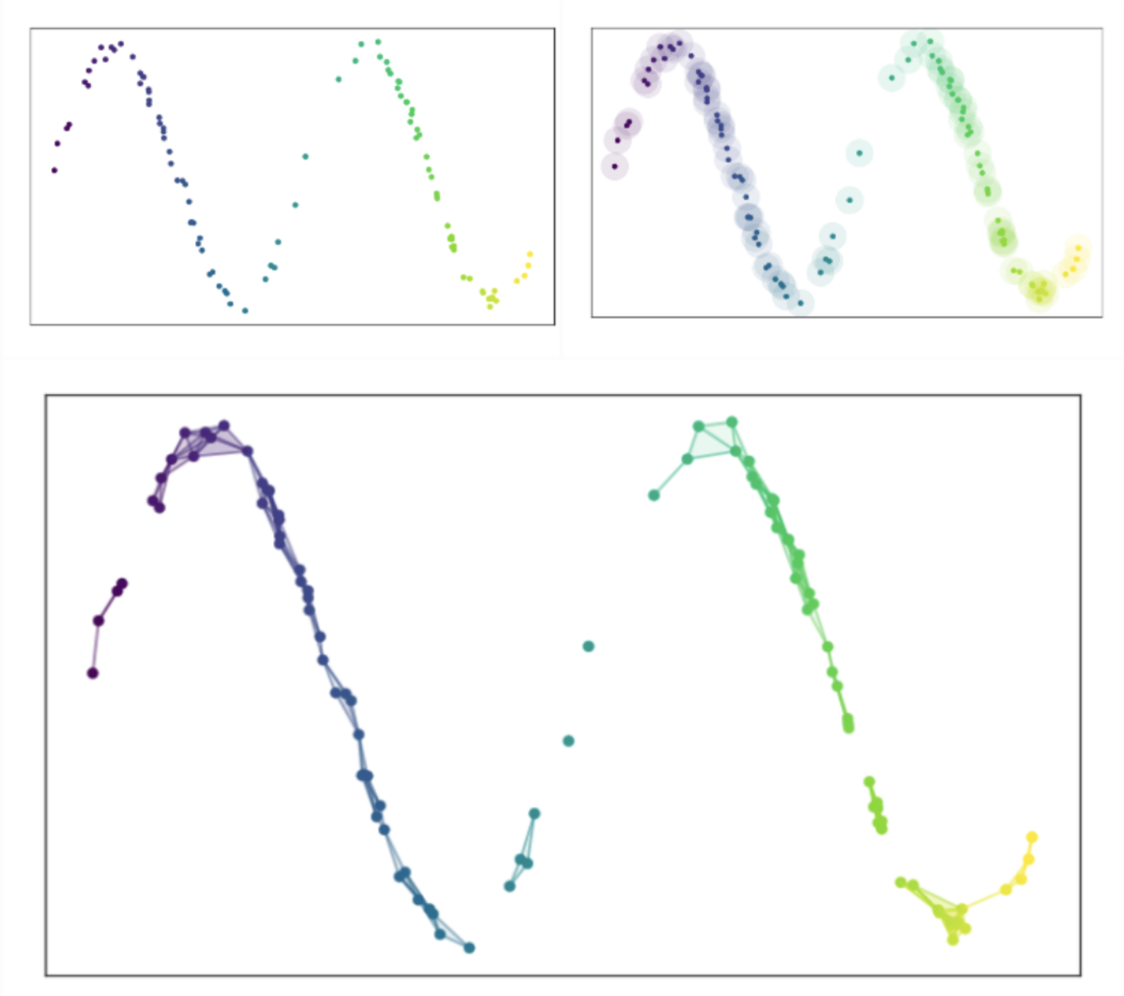


Figure 2.17: A simplicial complex process. (above left) initial noise sine wave data points. (above right) a basic open cover of the dataset. (below) A simplicial complex built from a dataset of a noisy sine wave [75]

Adapting fuzzy topological representation to real world data

In real world data, using some basic Riemannian geometry [76] and assuming the data is equally distributed, we could compute a local concept of distance for each point. In fact, the local sense of distance is relative to the proximity to the nearest neighbour point, each of which could have a different one. The manifold can have more than one part that is linked to it. Instead, it needs that every place on the manifold is in a small enough area around the linked point. This is what the term "local" means.

It is clearly seen that local measurements are incompatible. Each point has its own local metric, therefore the distance from point a to point b may be 1.5, while from point b to point a may be merely 0.6. Mathematically, we have a group of fuzzy simplicial sets, and their union is a well-defined operation. In graph terminology, this means that if two edges of weight a and b disagree, they should be merged into a single edge with weight as equation 2.14.

$$a+b-a\cdot b \tag{2.14}$$

The weights are basically the probability of an edge (1-simplex) existing. The combined weight represents the likelihood of one or more edges. When all the fuzzy simplicial sets are combined, we get a single fuzzy simplicial complex, which we may think of as a weighted graph. We are simply applying the edge weight combination formula on the entire graph (with non-edges having a weight of 0). So now, we will get the fuzzy topological representation of the data with combined edge weights.

Building a Low-Dimensional Representation

The low dimensional representation should have a comparable fuzzy topological structure. Finding an appropriate low dimensional representation relies on measuring the similarity of fuzzy topological structures. Using this metric, we can discover the low dimensional representation with the closest fuzzy topological structure. Obviously, the optimization approaches available will vary depending on the features of our measure of proximity. We regarded the weights associated with simplices as the probability of the simplex occurring when we were merging them. Since both topological structures have the identical 0-simplices, to compare the two probability vectors indexed by the 1-simplices, the cross entropy is selected.

$$\sum_{e\in E} w_h(e)\cdot log(\frac{w_h(e)}{w_l(e)})+(1-w_h(e))\cdot log(\frac{1-w_h(e)}{1-w_l(e)}) \tag{2.15}$$

Specifically, if E is the set of all possible 1-simplices and we have weight functions such that $w_h(e)$ is the weight of the 1-simplex e in the high dimensional case and $w_l(e)$ is the weight of e in the low dimensional case, then the cross entropy will be used to minimise the loss as a type of force-directed graph layout algorithm. As a result of this process of pull and push, the low dimensional representation will finally settle into a state that closely matches the general structure of the source data

2.4 Clustering Algorithms

Clustering is a process of finding cluster structure in dataset. it attempts to classify the data points that have been labelled into different groups or clusters. Similar members of clusters or groups should be grouped together as much as possible, and distinct members of clusters should be distinguished as much as possible. Because no class label is used in the learning process, it is an unsupervised classification. Kaur Mann et al. [33] define that a quality clustering method will produce high superiority groups with minimal inter-class similarity. The superiority of a clustering result relies on both the method's similarity measure and its implementation. A clustering technique's superiority is determined by its ability to uncover hidden patterns. The distance function may represent a cluster's similarity. Data mining requires certain criteria for data clustering. These are scalability, attribute handling, and attribute handling. Handling changing data, finding random groupings, domain knowledge is required to determine input parameters. Adaptable to noise and outliers Insensitive to input record order, dimensionality, User-specified limitations, Readability and usability

2.4.1 Hierarchical-based Clustering (HC)

Hierarchical clustering (a.k.a. hierarchical cluster analysis or HCA) is a cluster analysis technique used in data mining and statistics that aims to create a hierarchy of clusters. This technique is based on connectivity-based clustering methods. It clusters the data using the distance matrix criterion. It builds clusters incrementally [33]. Hierarchical clustering's benefits are embedded granularity, flexibility and ease of dealing with any degree of resemblance or dissimilarity. However, any attribute type of Hierarchical clustering drawbacks are lack of clarity in termination criteria and the most hierarchical algorithms do not enhance previously created clusters. Hierarchical clustering's benefits are at the expense of efficiency. Compared to the linear complexity of K-means and Expectation-Maximisation, the most popular hierarchical clustering techniques have a complexity that is at least quadratic in the number of samples [52].

The hierarchical Clustering (HC) objectives are defined and optimised in the space of binary trees with n leaves, where n is the number of data points. Every binary tree with n leaves represents a series of exactly n. Generally, techniques for hierarchical clustering are classified into two categories [47].

Agglomerative Clustering

This is a "bottom-up" approach: each observation begins in its own cluster, and as one moves up the hierarchy, pairs of clusters are combined. This approach creates a tree of clusters, also known as nodes. The following criteria are utilised to cluster the data in this method: minimum distance, maximum distance, average distance, and centre distance. The stages in these procedures,

1. In the initial stage, the algorithm considers each data point as a cluster and selects a proximity matrix to measure the distance between clusters. For the proximity matrix, four distance functions are available: single linkage (min), average linkage, full linkage, and ward (max). Single linkage indicates that the distance between two clusters is defined as the smallest distance between two points in the first cluster. Full linkage uses a maximum of two data points to connect two clusters. Average linkage computes the distance between two clusters by averaging all data points from the first cluster. Ward utilises the sum of squares to compute distance between locations.

2. To determine the closest pair of clusters, it calculates the similarity (distance) between each cluster.

3. Then, based on the distance function, clusters that are similar are merged to form a single cluster.

4. Steps 2 and 3 are repeated iteratively until all data points are merged into a single cluster.

Hierarchical clustering typically involves constructing a single tree of clusters, where each node represents a cluster and each data point begins as a tree leaf. The tree's origin is the final cluster containing all data points

Divisive Clustering

a "top-down" technique in which all observations begin in a single cluster and are divided recursively as one descends the hierarchy. It is the reverse of the agglomerative approach. Starting with the root node (cluster), each node creates the cluster (leaf) on its own. Bottom-up approaches make clustering decisions based on local patterns without taking the global distribution into account initially. These early decisions are irreversible. When making top-level partitioning decisions, thorough information about the global distribution is helpful for top-down clustering [52].

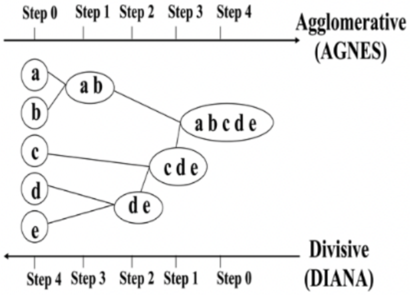


Figure 2.18: Representation of agglomerative and divisive approach [33]

Hierarchical clustering output

Generally, the results of hierarchical clustering are displayed in the form of a dendrogram. In hierarchical clustering, there are no assumptions about the number of clusters during dendrogram construction. After constructing the dendrogram, this structure is sliced horizontally. All of the subsequent child branches generated below the horizontal cut represent an individual cluster at the highest level and specify the membership of each data sample within that cluster.

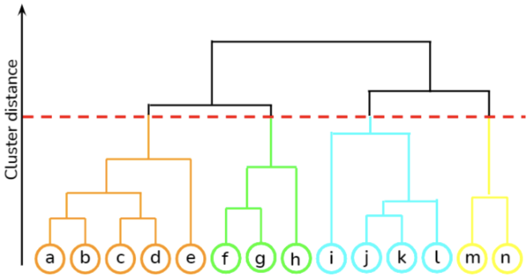


Figure 2.19: Dendrogram with clusters marked [34]

2.4.2 Gaussian Mixture Model (GMM)

A Gaussian Mixture Model (GMM) is a parametric probability density function that is represented numerically as the weighted sum of Gaussian component densities [37]. GMM parameters are estimated from training data using the iterative Expectation-Maximisation (EM) approach, allowing the model to learn the sub populations automatically. Since sub population assignment is not known, this constitutes a form of unsupervised learning.

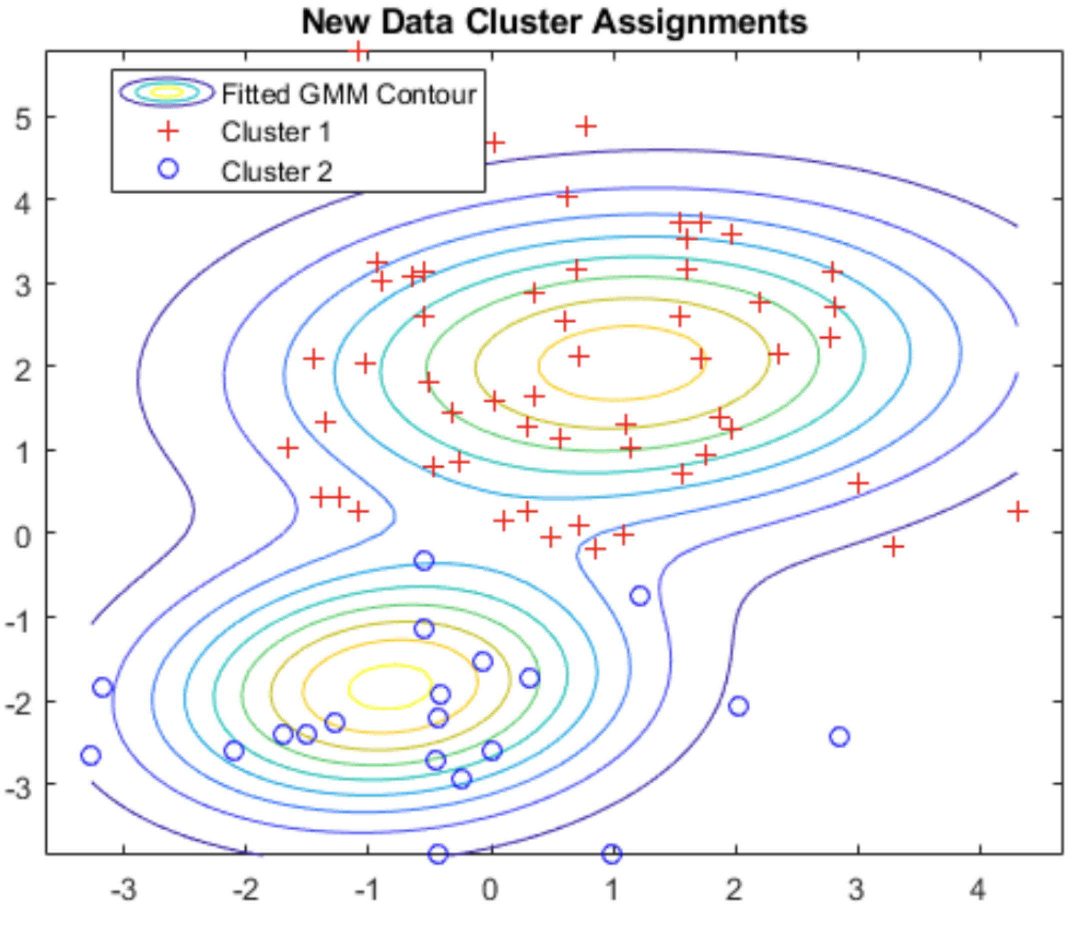


Figure 2.20: Mixture components with data points visualisation [35]

This algorithm is based on mixture models, they are probabilistic models for representing the presence of sub populations within an overall population that do not require that an observed data set identify the sub population to which an individual observation belongs. In statistics, a mixture model is a model that represents the presence of sub populations within an overall population without requiring that an observed data set identify the sub population to which an individual observation belongs. a mixture model corresponds to the mixture distribution, which reflects the probability distribution of observations over the whole population.

Gaussian distribution

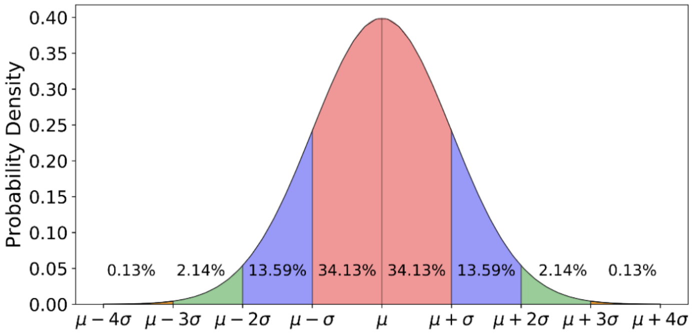


Figure 2.21: Gaussian Distribution (Normal distribution)

Gaussian distribution, also known as the normal distribution, is a probability distribution that is symmetric around the mean, indicating that data near the mean occur more often than data distant from the mean. A normal distribution will show as a bell curve on a graph. A normal distribution is a probability distribution that is used to explain events with a default state and cumulative potential departures from that state. The key attribute that may be observed is that the mean, median, and mode all have comparable values, which results in a symmetric distribution. defining X as variables observe, μ distribution's mean or expectation, σ standard deviation and σ^2 distribution variance. As equation 3, the general expression for $f(x)$ probability density function.

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{1}{2}(\frac{x-\mu}{\sigma})^2} \tag{2.16}$$

Gaussian distribution is a version of the standard normal distribution that been stretched by a factor of (standard deviation) and translated with (mean). The mean defines the position of the curve's apex. By increasing and reducing the mean, the curve will shift to the right and left. The standard deviation determined the curve's width. Increased standard deviation results in a wider curve.

For multivariate Gaussian Distribution, the parameter is denoted d Gaussian distribution of a vector dimensions, x as input vector in d length, μ as dimensional vector of the distribution mean. Σ are dxd size covariances matrix and $|\Sigma|$ are the determinant of the covariance matrix.

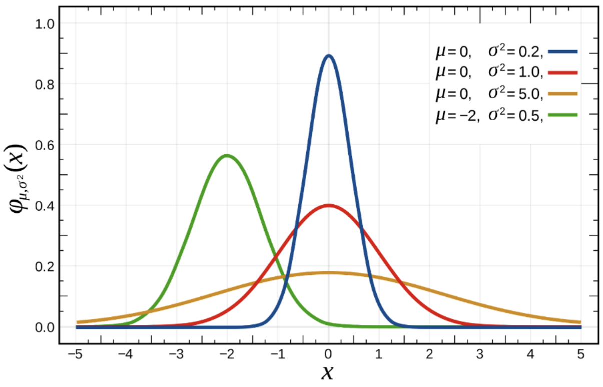


Figure 2.22: Gaussian distribution curve in difference mean and variance

So, we could be expressed as the probability density function

$$N(x|\mu,\Sigma) = \frac{1}{(2\Pi)^{d/2}\sqrt{|\Sigma|}} exp(-\frac{1}{2}(x-\mu)^T(x-\mu)\Sigma^{-1}(x-\mu)) \tag{2.17}$$

For a dataset with d features, we would have a mixture of k Gaussian distributions (where k is equivalent to the number of clusters), each having a certain mean vector and variance matrix. Interpolating over causes the Gaussian to shift in the d-dimensional (hyper)plane, while modifying the matrix causes the Gaussian to change shape.

Gaussain Mixture Model

A Gaussian mixture model is a density model where we combine a finite Gaussian mixture number of K Gaussian distributions $N(x|\mu_k,\Sigma_k)$ model.

$$p(x|\theta) = \sum_{k=1}^{K} \pi_k N(x|\mu_k,\Sigma_k) \tag{2.18}$$

$$0 \leq \pi_k \leq 1, \sum_{k=1}^{K} \pi_k = 1$$

Where we defined $\theta := \{\mu_k,\Sigma_k,\pi_k : k = 1,...,K\}$ as the collection of all parameters of the model. This convex combination of Gaussian distribution give more flexibility for modelling complex densities than a simple Gaussian distribution.

Parameter Learning via Maximum Likelihood

The purpose is to use a GMM with K mixture components to find an acceptable approximation of this unknown distribution $p(x)$. Assume that given a dataset $X = x_1,...,x_N$ where x_n, $n = 1,...,N$ are drawn from Independent and identically distributed (i.i.d) of an unknown distribution $p(x)$. The parameter of the GMM are the K-means μ_k, the covariances Σ_k, and the mixture weigh Π_k. As a result, we describe all these free parameter in $\theta := \{\mu_k,\Sigma_k,\Pi_k : k = 1,...,K\}$.

To get a maximum likelihood (ML) of the model parameters, We begin by putting down the likelihood, or the training data's predicted distribution given the parameters. We take use of our i.i.d. assumption, which results in factorised likelihood.

$$p(X|\theta) = \prod_{n=1}^{N} p(x_n|\theta), \; p(x_n|\theta) = \sum_{k=1}^{K} \pi_k N(x|\mu_k, \Sigma_k), \tag{2.19}$$

$$log\ p(X|\theta) = \sum_{n=1}^{N} logp(x_n|\theta) = \sum_{n=1}^{N} log \sum_{k=1}^{K} \Pi_k N(x|\mu_k, \Sigma_k) =: L \tag{2.20}$$

Where $p(x_n|\theta)$ is Gaussian mixture density likelihood term for each individual in equation 2.19. The log-likelihood is obtained as equation The objective of equation 2.20 is to identify the parameters θ^*_{ML} that maximises the log-likelihood L. However, it is not a "typical" approach for computing the log-likelihood gradient $dL/d\theta$. We are unable to find a closed-form solution. we can form the maximise the likelihood $p(X|\theta)$ of the data with regard to the model parameters by

$$\theta^* = argmax\ p(X|\theta) = argmax \prod_{i=1}^{N} p(x_i|\theta) \tag{2.21}$$

However, the results from normal parameters estimate show a straightforward iterative procedure for solving the parameter estimation issue using the EM algorithm's maximum likelihood approach would be helpful to solve a complexity in process. Expectation-Maximisation (EM) algorithm for GMMs can be used to identify appropriate model parameters θ_{ML},

Expectation-Maximisation (EM)

The Expectation-Maximisation (EM) method, a statistical technique for determining the optimal model parameters, was proposed by Dempster et al. [51] and is a general iterative scheme for learning parameters (maximum likelihood or MAP) in mixture models and, more generally, latent-variable models. EM is generally used when data has missing values. These unobserved variables are referred to as latent variables. In unsupervised learning problems, the objective or cluster number is set to be unknown. Due to these missing variables, it is difficult to find the appropriate model parameters. Determining the mean vector and covariance matrix would be straightforward if we knew which cluster corresponded to which data point. Since unknown values of the latent variables, Expectation-Maximisation attempts to identify the optimal values for these variables using the available data and then discovers the model parameters

These are the two main parts of the EM algorithm: the *E* Step, also known as the Expectation Step or Estimation Step, and the *M* Step, also known as the Maximisation Step.

- Estimate step:
  - μ_k, Σ_k and Π_k should be initialised with some random values, or with K means clustering or hierarchical clustering results.
  - Then, using the specified parameter values, estimate the latent variables values.
- Maximisation step:
  - Update the values of the parameters (μ_k, Σ_k and Π_k) determined using the Maximum Likelihood approach.

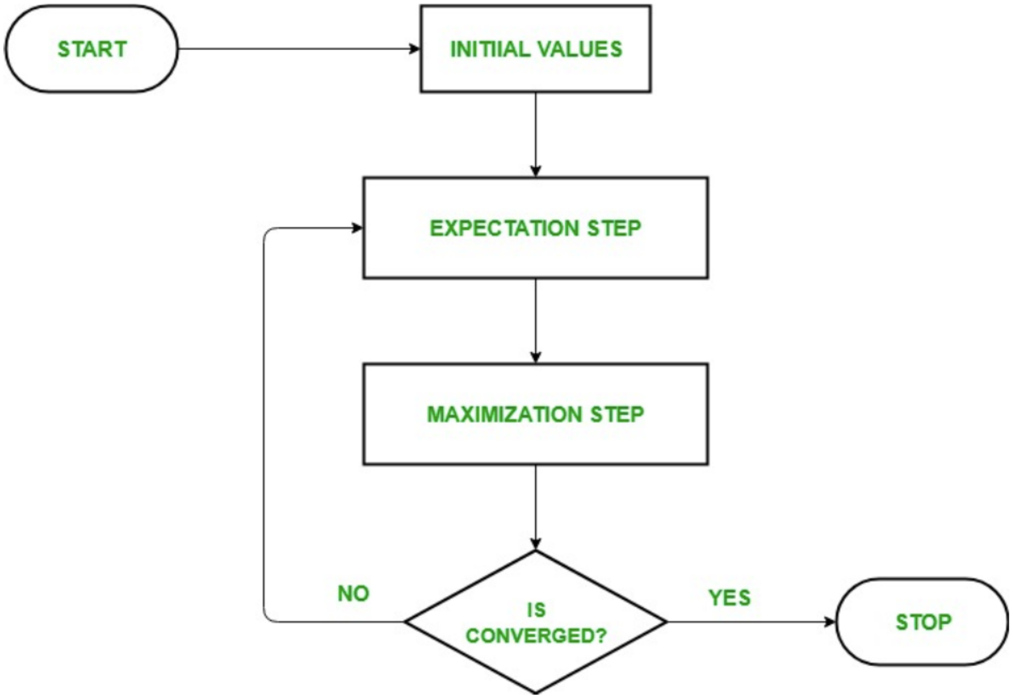


Figure 2.23: Expectation-Maximisation Algorithm [36]

2.5 Cluster Quality Assessment

Clustering is a type of unsupervised learning and a typical tool for statistical data analysis used in a variety of disciplines. The function of the assessment technique is to assess the quality of the data in certain aspects. Clustering does not need labels to execute the process of forming the clusters; only a collection of characteristics for each observation is involved. The objective is to generate clusters with similar observations grouped together and different observations maintained as far apart as possible. clustering algorithms split an input data set into a number of partitions, or clusters. For tasks where a target partition is established for testing purposes, a clustering solution is described as a mapping from each data point to its cluster assignment in both the target and hypothesised clustering. Unlike supervised learning methods, evaluating the success of a clustering algorithm is not as simple as counting the number of mistakes or the accuracy and recall. For instance, clustering may be performed on cancer samples with the expectation that samples in the same group represent the same subtype of cancer. Once the clustering has been executed, the quality of the result must be evaluated qualitatively. First, to ensure that groupings are relevant, and second, since a score could act as a substitute for assessing different models and finding the best one.

2.5.1 Homogeneity

To achieve homogeneity criterion, a clustering algorithm must assign to a single cluster only the data points that belong to a single class. In other words, the distribution of classes inside each cluster should be skewed toward a single class, or have zero entropy. Examining the conditional entropy of the class distribution given the specified clustering makes it possible to estimate how near a clustering is to this target. In the situation of full homogeneity, the value $H(C|K)$ equals 0.The size of the value depends on the size of the dataset and the distribution of class sizes. Therefore, instead of taking the conditional entropy in its raw form, The value is normalised by the highest decrease in entropy that the clustering information could give $H(C)$. we define homogeneity as equation 2.22, 2.23 and 2.24

$$h = 1 - \frac{H(C|K)}{H(C)}, \tag{2.22}$$

$$H(C|K) = 1 - \sum_{c=1}^{|C|} \sum_{k=1}^{|K|} \frac{a_{ck}}{N} \cdot log \frac{a_(ck)}{\sum_{k=1}^{|K|} a_{ck}} \tag{2.23}$$

$$H(C) = - \sum_{c=1}^{|C|} \frac{\sum_{k=1}^{|K|} a_{ck}}{n} \cdot log \frac{\sum_{k=1}^{|K|} a_(ck)}{n} \tag{2.24}$$

2.5.2 Completeness

Completeness and homogeneity are symmetrical. To meet the completeness criterion, a clustering must assign to a single cluster all data points that belong to a single class. To evaluate completeness, the distribution of cluster assignments within each class is examined. Each of these distributions will be totally skewed towards a single cluster in a clustering solution that is perfectly thorough. This degree of skew could be determined by calculating the conditional entropy, $H(K|C)$, of the proposed cluster distribution based on the class of the component data points. In the event of perfect completion, $H(K|C) = 0$. we define completeness as equation 2.25, 2.26 and 2.27

$$c = 1 - \frac{H(K|C)}{H(K)}, \tag{2.25}$$

$$H(K|C) = -\sum_{c=1}^{|C|}\sum_{k=1}^{|K|} \frac{a_{ck}}{N} \cdot log \frac{a_(ck)}{\sum_{k=1}^{|K|} a_{ck}} \tag{2.26}$$

$$H(K) = -\sum_{k=1}^{|K|} \frac{\sum_{c=1}^{|K|} a_{ck}}{n} \cdot log \frac{\sum_{c=1}^{|K|} a_(ck)}{n} \tag{2.27}$$

2.5.3 V-Measure

V-measure is an entropy-based metric that specifically quantifies how well the homogeneity and completeness conditions have been met. It was proposed by Andrew Rosenberg and Julia Hirschberg in 2007 [67]. V-measure is calculated as the harmonic mean of distinct homogeneity and completeness scores, identical to how accuracy and recall are often merged into F-measure [68]. As F-measure scores may be weighted, so too can V-measure scores be weighted to favour homogeneity or completeness.

After calculating the weighted harmonic mean of homogeneity and completeness, the V-measure of a clustering solution is computed. If it is larger than 1, completeness is given more weight in the computation, and if it is smaller than 1, homogeneity is weighted more heavily.

$$V = 2 \cdot \frac{(h \cdot c)}{h + c}, \tag{2.28}$$

There is also another widely used extrinsic method called Adjusted Rand Index (ARI). However, the limitation of the ARI, which highly relies on having accurate true class labels to accurately reflect the quality of a clustering solution, makes it inappropriate for use in situations where the true class labels are derived from patient-level annotations, as occurs in our second dataset (TCGA-COAD).

2.5.4 Silhouette

A simple graphical representation for partitioning approaches was proposed by Peter Rousseeuw in 1987 [69]. Each cluster is represented by a so-called silhouette based on a comparison of its closeness and distance. The silhouette illustrates which items are well-integrated within their cluster and which are positioned between clusters. By combining the silhouettes into a single plot, the full clustering is presented, providing for an assessment of the relative quality of the clusters and an overview of the data setup. The average silhouette width gives an assessment of the clustering's validity and may be used to determine the 'optimal' number of clusters [70].

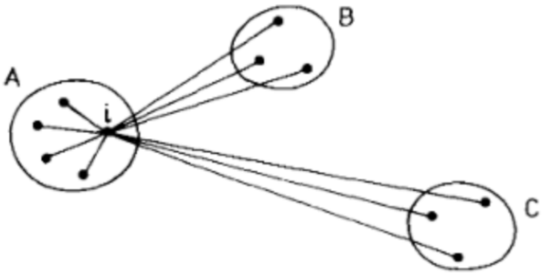


Figure 2.24: Illustration of the elements and its distant from each cluster when computation [69]

The silhouettes formed below are beneficial when the distances are on a ratio scale (as in the case of Euclidean distances) and when looking for clusters that are compact and well-defined. In practice, the definition uses average proximities, as in the case of group average linkage, which is known to function best in situations involving approximately spherical clusters. To generate silhouettes, we need just two things: the division we've got (through the use of a clustering approach) and a collection of all item distances. For any object i in the dataset, the number of $s(i)$ is the dissimilarities value from objects. And denote by A the cluster to which it has been assigned. Then we can compute $a(i)$ is the average dissimilarity of i to all other objects of A.and $d(i,C)$ is average dissimilarity of i to all objects of Cluster C.

The average length of all lines going from i to C. After computing $d(i,C)$ for all $C\neg A$, and then the smallest of those numbers is selected. It can denote by equation 2.29

$$b(i) = minimum_{C\neg A} d(i,C) \tag{2.29}$$

And we can write the simple formula for compute silhouette score as

$$s(i) = \frac{b(i) - a(i)}{max\{a(i), b(i)\}} \tag{2.30}$$

Chapter 3

Experimental Framework Design

In this chapter, The contents contain the overview of two colorectal cancer datasets and the experimental strategy used to create the clustering framework on sub-tissue tiles. The data section will discuss the selected WSI dataset with characteristics, metadata, and labels, how to divide the WSI into sub-tissue tiles. In the experiments setting section, we describe an experimental design for observing essential configurations of framework to obtain the high-quality clusters. This study also includes how to handle patient data into the sets of training, validation, and test, how representation of tissue tile is extracted, how transforming the original feature space of WSI sub-tile patches influences the cluster quality, and we examine the selected clustering algorithms to perform on the feature representation through quality cluster assessment techniques.

3.1 Datasets

3.1.1 100,000 histological images of human colorectal cancer and healthy tissue (100k-nct)

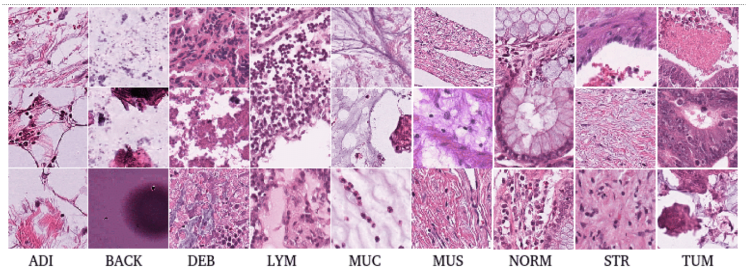


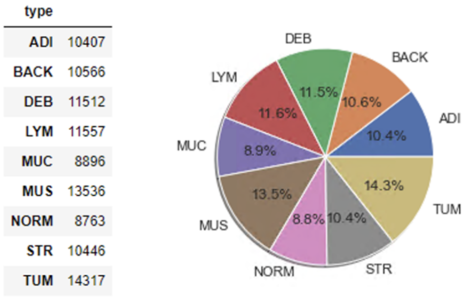
Figure 3.1: Samples from the NCT-CRC-HE-100k dataset [80]

This is a collection of 100,000 non-overlapping image patches extracted from hematoxylin and eosin stained histological images of human colorectal cancer and normal tissue. Each image is patched to 224x224 pixels (px), with a pixel size of 0.5 microns (MPP) and no colour normalisation was applied to these images. And it is also with the label of tissue types in table

Table 3.1: Tissue type description

Tissue type	Tissue type description
ADI	Adipose
BACK	background
DEB	debris
LYM	lymphocytes
MUC	mucus
MUS	smooth muscle
NORM	normal colon mucosa
STR	cancer-associated stroma
TUM	colorectal adenocarcinoma epithelium

3.1. In this study, the nct-100k dataset was chosen due to its richness of individual phenotype. Consequently, it will be potential to verify the hypothesis that our clustering framework can differentiate tissue visual characteristics without supervisory biases.



type	
ADI	10407
BACK	10566
DEB	11512
LYM	11557
MUC	8896
MUS	13536
NORM	8763
STR	10446
TUM	14317

Figure 3.2: Number of tissues type label on NCT-CRC-HE-100k

3.1.2 The cancer genome atlas colon adenocarcinoma (TCGA-COAD)

The data collection is part of the initiative to establish a scientific community engaged in linking cancer phenotypes to genetics by providing clinical pictures matched to participants from the cancer genome atlas [79]. As the diagnostic tissue images are captured by high-definition digital microscope and mapped with a patient clinical profile. The labels of which are derived from the patient level information. As with literature reviews, WSI examination was often performed by combining analytical results from smaller patches of equivalent size as opposed to examining the entire WSI. This dataset was used to verify the hypothesis that cluster assignment abundances of tissue tiles can reflect molecular or clinical patient profiles.

TCGA-COAD WSIs to sub-tile tissue patches

As the 100k-nct dataset was patched from the datasource, original images from the TCGA-COAD dataset are whole slide images that are cropped to match the 100k-nct dataset. The

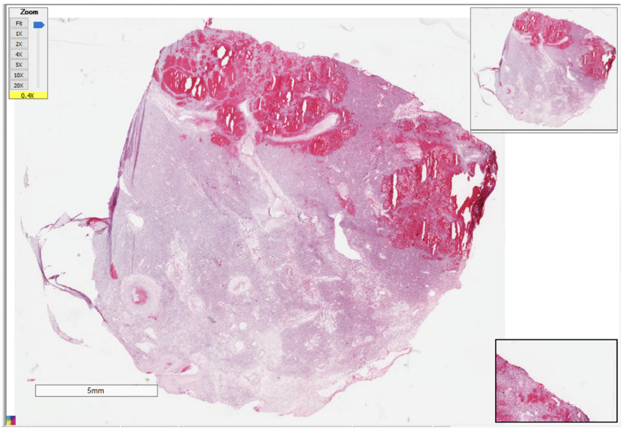


Figure 3.3: Whole slide image of TCGA-CZ-5467 case id

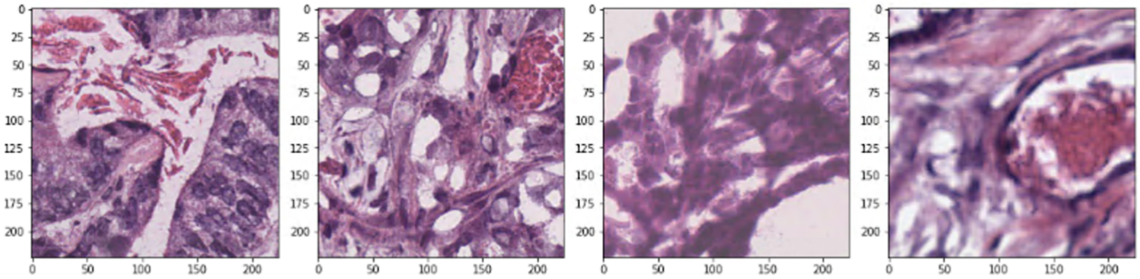


Figure 3.4: 224x244 pixel sub tile samples from WSI of TCGA-G4-6317 case id

TCGA-COAD consists of 459 WSIs with 20x magnification. This magnification level gives patches sufficiently large to capture a specific detail, such as nucleus details, without generating an excessive amount of patches after cropping. And WSIs are cropped with a 224x244 region of whole slide images. After that, 2,441,581 sub-tissue patches are produced.

TCGA-COAD External Annotation

WSIs annotations are from collaborating between K. Yuan Lab and UCL on detecting quiescence in colorectal tissues. From 57 unique patient quiescence labels annotated on each patient-level TCGA-COAD WSIs. Annotations for quiescence can represent the state of quiescence in colorectal tissues for each patient. 30 WSIs are annotated on label 0 and 27 on label 1. And unlabeled WSIs are considered to be undefined status, so we represent them as label 2. Consequently, each patient only produces one WSI. We can treat patient-level annotation as slide-level

Table 3.2: TCGA-COAD's WSIs quiescence labels description

quiescence label	label description
0	negative quiescence
1	positive quiescence
2	uncertain result

annotation. As shown in figure 28, 125,664 tiles are annotated on label 0, 185,028 tiles on label 1 and 2,130,889 tiles on label 2.

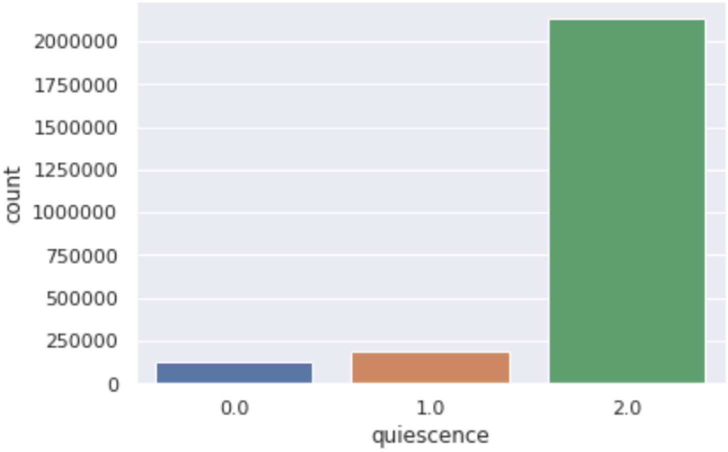


Figure 3.5: number of different quiescence labels

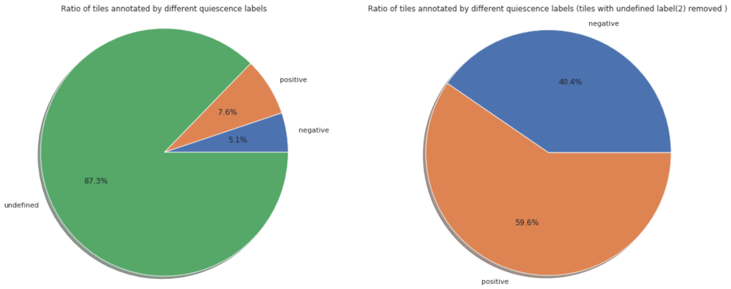


Figure 3.6: Ratio of tiles annotated by different quiescence labels and after filtered out undefined label

3.2 Experiment setting

In this section, we describe thoroughly how we extract deep features from various deep extractors such as pathologyGAN, ResNet50, InceptionV3 and VGG16, how to fitting features with dimension reduction techniques, PCA and UMAP, observing how to use the Silloutte coefficient to find optimal cluster configurations and how to subject high quality clusters to intrinsic measurements such as V-measure. Therefore, the above could all be defined in the following steps.

1. Sub-tissue patch representation extraction
2. Dimension Reduction or feature transformation
3. Splitting into train, validation, test
4. Clustering Configuration
5. High Quality Cluster Evaluation

3.2.1 Sub-tissue patch representation extraction

In this study, four different candidate methods of representation extraction were chosen to vectorise all tissue tiles for cluster analysis.

PathologyGAN Encoder

PathologyGAN Encoder is the first method considered to be the most potential feature extractor for tissue tiles with respect to the literature review and intra-team knowledge. For training configuration, the input image is of 224 x 244 size with 3 image channels. 150 dimensions of the attention network, 10,000 maximum number of instances for a bag and output of 200-d vector as the latent space from the model. It runs on a 1-e4 learning rate, 50 epochs, 10 folds. Due to Generative Adversaries Network requires large number of samples to achieve the equilibrium, PathologyGAN Encoder was train based on sub-tissue patch from TCGA-COAD.

ResNet50, InceptionV3 and VGG16

The other 3 methods, ResNet50, InceptionV3 and VGG16, are the toplist of deep architectures according to recent research. These 3 features are fitted by ImageNet and removing the top fully-connected layers of a deep classifier, the remaining parts work as a feature extractor. These methods are based on pre-trained weight from ImageNet dataset with 224 x 244 input size, and other default parameters on Pytorch. It runs on 4,096 batch-size. In ResNet50 and InceptionV3, we only selected the 2,048 sized outputs from avg_pool (GlobalAveragePooling) before the classification layer. Similarly, VGG16, 4,096 sized fully-connected layer output is selected. Finally, two colorectal datasets were inferenced to get feature space by these methods.

3.2.2 Dimension reduction or feature transformation

From the outputs of the representation extraction methods, high-dimensional features are extracted. the data become very sparse, and the analysis is plagued by the curse of dimensionality. It also creates problems in terms of time and space while training. Dimension reduction strategy is employed to turn original features into a more clusterable form.

Apart from clustering sake, visualisable feature space is preferred for interpretation, we select only the 3 dimensions as the number of components in these experiments. The new 3-d representation makes it easy enough to interpret the feature space of different feature extractor from how well their performance on two colorectal datasets. On the fitting data, we selected only 5% (122,079 samples) of the all slide of TCGA-COAD sub-tissue patch. and 100k-nct was selected all sample to fit estimators.

Principal components analysis (PCA)

First, Principal components analysis (PCA) is selected by its popularity over time and robustness to preserve important features. PCA estimators was used all default parameters except number of components(n_components) at 3.

Uniform manifold approximation and projection (UMAP)

This method uses manifold learning techniques and topological analysis concepts to tackle by reducing complexity in topological spaces and continuous geometry to basic combinatorics and counting. the UMAP was configured the size of local neighborhood (n_neighbors) at 30 , The effective minimum distance between embedded points(min_dist) at 0, number of components(n_components) at 3 and enable low memory configuration.

3.2.3 Splitting train, validation, test

In this study, Two colorectal datasets were divided into train and test data with proportions of 66.66 % and 33.33 % respectively. In 100k-nct, we simply and directly split the data by sub-tissue sample. However, All TCGA-COAD sub-tissues patches were annotated from patient level. To avoid the same patient WSIs sub-tissues contaminated into training and test set, we divide dataset by patient-level criteria.

For validation set, we employed 3-fold cross-validation strategy on training set to stabilise a machine learning model's skill on new data. The cross-validation can help estimate how selected cluster configuration e.g. number of cluster will perform on data not being seen during training. It's straightforward and gives a less biased or optimistic estimate of model skill.

As Figure 31, we can see the ratio of tiles annotated by different quiescence labels without label 2. In the training set, 46.90% of tiles are annotated on label 0 and 52.10% of tiles on label 1. And In merging validation set and test set, 30.70% of tiles are annotated on label 0 and 69.30% of tiles on label 1.

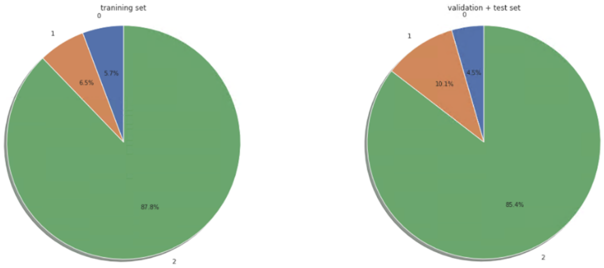


Figure 3.7: Ratio of tiles annotated by different quiescence labels

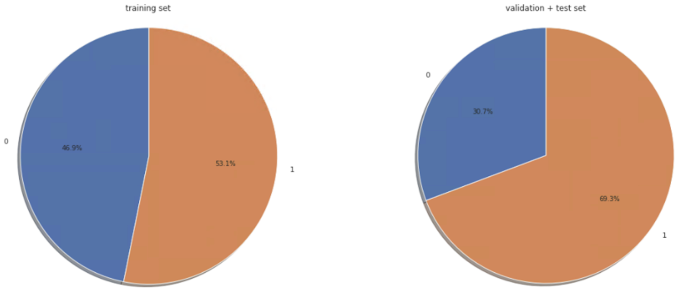


Figure 3.8: Ratio of tiles annotated by different quiescence labels without label 2

3.2.4 Clustering configuration

In order to cluster the features from different features, two clustering concepts are examined: prototype-based clustering and hierarchical-based clustering. Firstly, Gaussian Mixture Model (GMM), the one for clustering for prototype-based clustering, is a probabilistic model that assumes all the data points come from a mix of a finite number of Gaussian distributions with unknown parameters. One way to think of mixture models is as an extension of k-means clustering that takes into account the data's covariance structure as well as the centres of the latent Gaussian. And the another one, Agglomerative hierarchical clustering is selected to be the one for hierarchical-based clustering. Each observation begins in its own cluster, which is then gradually combined to produce hierarchical clustering using a bottom-up approach. In this study, Agglomerative hierarchical clustering is used ward linkage criteria for merge strategy. It is a variance-minimising strategy by sum of squared differences within each cluster. And we also employ K- nearest neighbours after clustering results from Agglomerative hierarchical clustering.

Both algorithms are experimented by 2 to 50 numbers of clusters. In the training session, we decided to downsampling training size according to the capability of processing memory on the lab workstation. Due to the large number of sub-tile tissues from TCGA-COAD, training sample were kept only 5% on each fold randomly. In each validating variable, optimal number of cluster is selected based on maximal silhouette score.

3.2.5 Cluster quality evaluation

From the two colorectal cancer datasets with annotations, it can be applied by extrinsic methods that measure the correlation of resulted clusters to group truth. If the ground truth is unavailable, we can evaluate the quality of a clustering based on how well the clusters are separated using intrinsic methods. Ground truth can be considered as supervision in the form of "cluster labels." In consequence, extrinsic methods are also known as supervised methods, while intrinsic methods are unsupervised methods [5].

In order to find the optimal cluster number, Silhouette coefficient was selected. It is intrinsic methods to find measure of how similar an object is to its own cluster (cohesion) compared to other clusters (separation) by using the mean intra-cluster distance and the mean nearest-cluster distance for each sample with unsupervised manner.

For extrinsic methods, V-measure is considered to be the one that is solving problems such as dependent on clustering algorithm or dataset, problem of matching of evaluated portion of data and aspect of completeness and homogeneity within clusters.

Chapter 4

Result and Discussion

In this research, we aim to develop a clustering framework that yields high quality clusters of sub-tissue to be claimed as phenotype clusters The clusters are expected to recognise tissue visual characteristics without supervisory biases and cluster assignment abundances of tissue tiles is expected to reflect patient molecular or clinical profiles. In chapter 3, we discussed the two colorectal datasets and how they are splitted into tissue patches. In experimental setting section, we describe an experimental design for observing a multitude of framework configurations on the high-quality clusters. This chapter describes the experimental outcomes and manifests optimal unsupervised clustering framework with respect to ability to capture biological visual characteristic. Interpretation and explanation on promising experimental findings will be discussed.

4.1 Reduced Dimension Feature Space Visualisation

The two colorectal datasets were passed to generate the feature by different representation extractors, PathologyGAN, RESNET50, InceptionV3 and VGG16. Each representation was then applied with dimension reduction techniques, PCA and UMAP which transformed the representations into 3 dimensions. Depicted in figure 4.1, three dimensional UMAP feature space showed distinguishing areas of tissue types than that of PCA. Speaking of PCA, Intra-distances are small among datapoints in the same tissue type. Notwithstanding, distances between cluster of tissue type are significantly smaller than that is by UMAP.

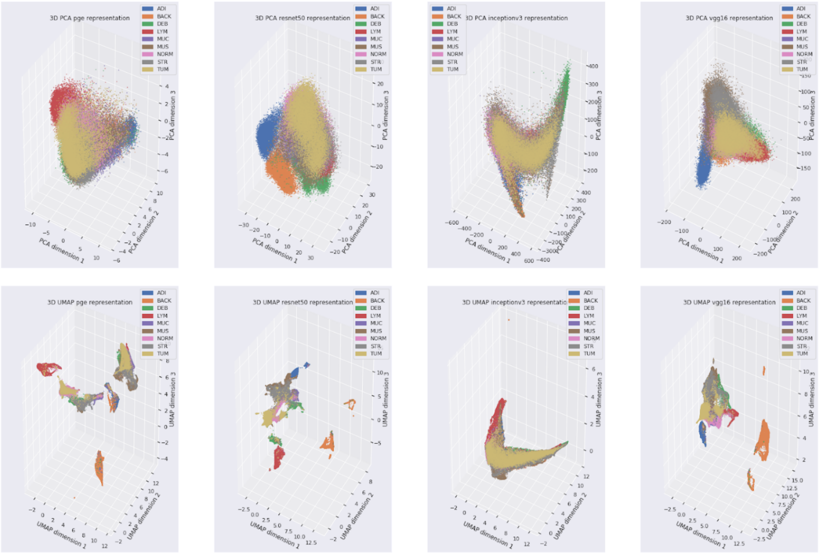


Figure 4.1: 3D PCA (top) and 3D UMAP (below) representation in NCT-100k and labelled with tissue type from PathologyGAN, ResNet50, InceptionV3 and VGG16 respectively.

In each transformed feature space, TCGA-COAD sub-tissue patches were labelled with quiescence label as shown in figure 4.2. Only tissues with negative and positive quiescence are illustrated since there is no pattern perceived when samples with uncertain quiescence included. PCA algorithm quite under-performed in the entire candidates of representation about separating particular quiescence-related clusters.

4.2 Optimal number of cluster suggested silhouette score

After we get feature vectors from each image representation extractors and each of them is reduced in dimension by two dimensionality reduction techniques, those defined training sets were to train on by Gaussian Mixture Model(GMM) and Hierarchical-based clustering(HC). Both algorithms were experimented by varying numbers of clusters from 2 to 50.

4.2.1 Silhouette score on 100k-nct

Firstly, those features sets of 100k-nct images were trained with both clustering algorithm. As figure 4.3, Calculating and visualizing the silhouette score according to clustering the images on original, 3d-PCA, and 3D-UMAP feature spaces. The line colours are blue, orange, and green respective. Overall, the silhouette score for the feature space processed by UMAP is considerably higher than that of the original and PCA.

Figure 4.4 illustrates all the highest silhouette scores by each image representation. PathologyGAN on 3D-UMAP yield the highest silhouette score at 0.68. and follow by resnet50, inceptionV3 and vgg16. and their score are 0.65, 0.49 and 0.61 respectively.

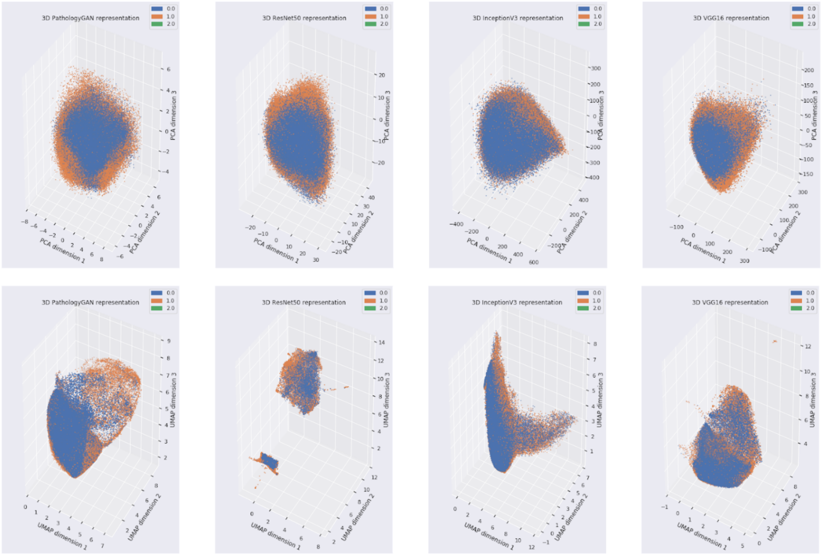


Figure 4.2: 3D PCA (top) and 3D UMAP (below) representation in TCGA-COAD and labelled with quiescence label from PathologyGAN, RESNET50, InceptionV3 and VGG16 respectively.

As shown in figure 4.5, number of clusters suggested by the Silhouette score from all image representation methods on 100k-nct. The number of clusters which produced the highest scores is implied as the optimal number of cluster for a specific representation. number of clusters from 3 to 12 is indicated for all grouped characteristics. Restnet50 suggests a greater number of clusters than other features, which is relatively close to the number of ground-truth classes.

For more references, figure 4.6 presents silhouette score from GMM and HC on 100k-nct on 3D UMAP is explored. The star symbols denoted the optimal number of clusters based on the yield silhouette score with the highest value in each representation technique.

4.2.2 Silhouette score on TCGA-COAD

Another colorectal dataset, both clustering algorithms were employed to train clustering on 4 different image representations as it was done on 100k-nct. As shown in figure 4.7, silhouette scores were computed on the clustering with PCA and UMAP feature spaces. and its line colours are blue and orange respectively. We decided to skip the training on original features because of computational complexity in multi-million samples with high dimension. It is out of the capacity of lab workstation. Moreover, low quality of clustering on 100k-nct which has been shown and will be shown later in this section demonstrates no reason for further investigation on any original feature spaces

In this dataset, clustering by HC on 3d-UMAP features produced significantly higher silhouette scores than 3D-PCA regardless of image representation. However, the performance of GMM on reduced feature spaces by UMAP and PCA are comparable. UMAP-reduced features yield better silhouette score than PCA in small number of clusters such as 2 to 10 for patholo-

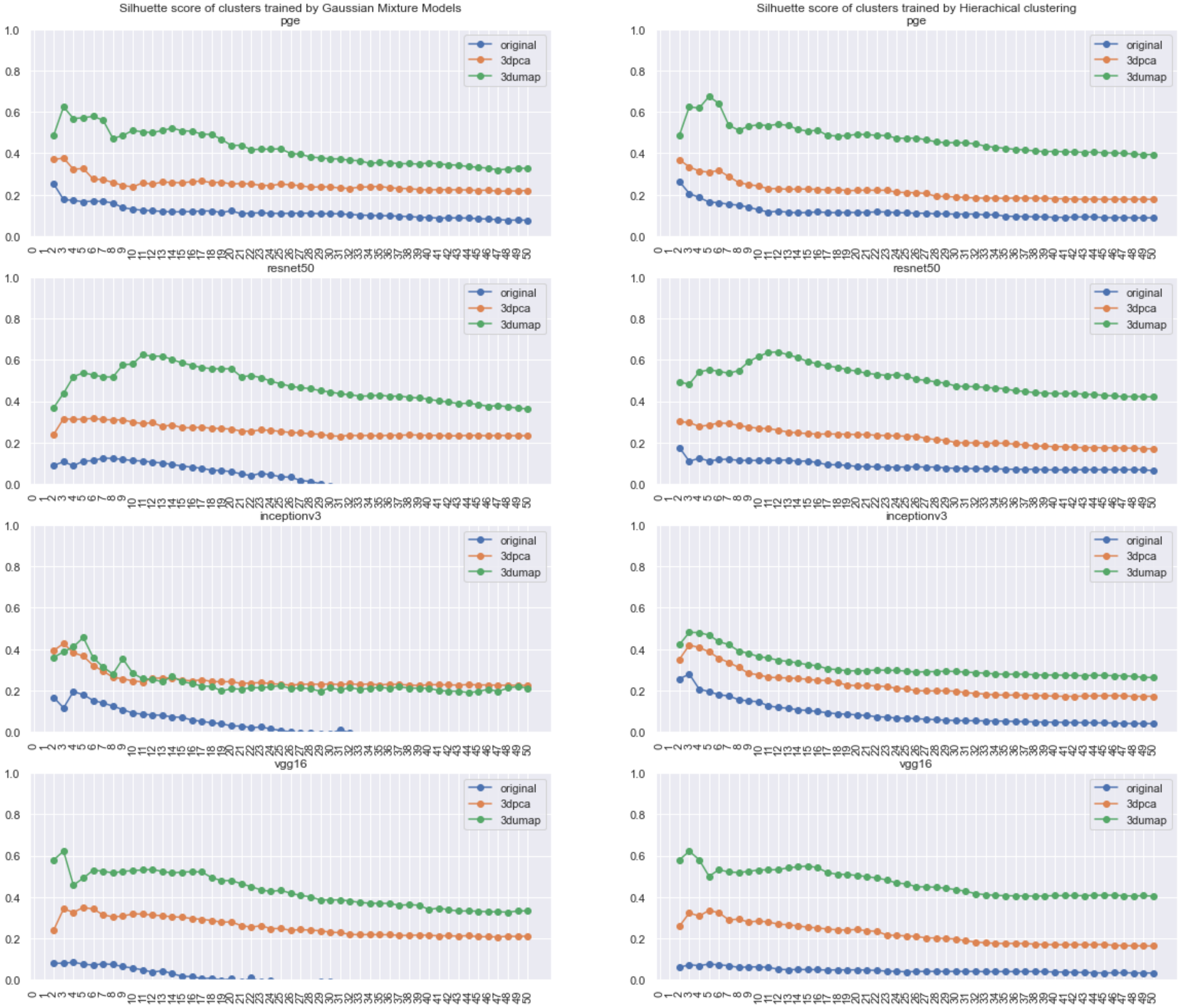


Figure 4.3: Silhouette score of 100k-nct trained by GMM and HC.

gyGAN, 2 to 3 for Resnet50, 2 to 9 for InceptiveV3 and 2 to 7 for VGG16 while PCA-reduced features yield higher in relatively greater number of clusters.

Figure 4.8 reveals the highest silhouette scores of clustering the TCGA-COAD dataset on 4 different image representations. Resnet50 which was dimensionality-reduced by 3D-UMAP has the highest silhouette score, at 0.89 in both clustering algorithms. Regardless of representation and clustering algorithm, it is noticeable that UMAP gains higher quality clusters on the basis of internal statistical properties. compared to PCA.

From figure 4.9. As overall representations dimensional-reduced by UMAP produced the higher silhouette score than PCA, The silhouette score suggested the number of clusters between 2 to 5 among all image representations.

In this setting, silhouette scoring tends to suggest the lower number of cluster in which 2

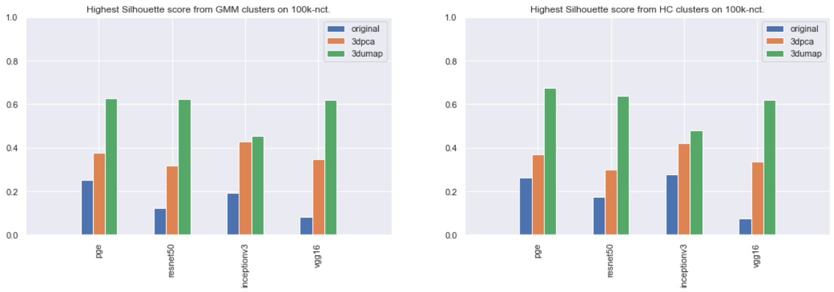


Figure 4.4: Highest Silhouette score from GMM and HC clusters on 100k-nct.

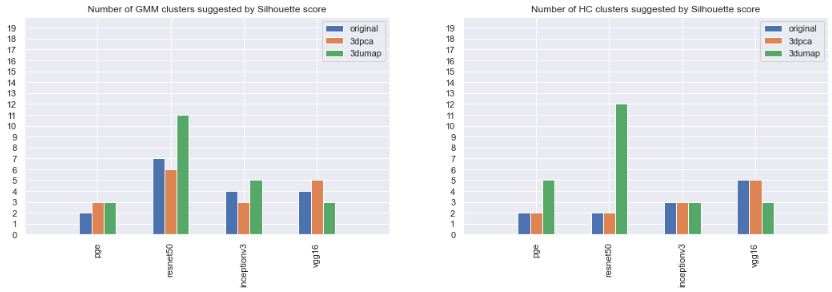


Figure 4.5: Number of clusters suggested by Silhouette score on 100k-nct.

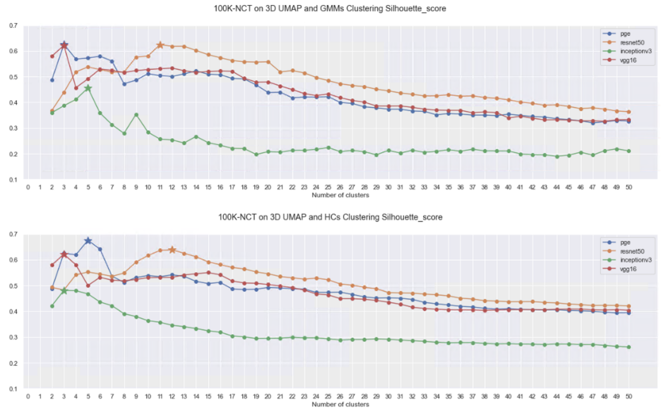


Figure 4.6: Suggest number of cluster by Silhouette score in GMM and HC on 100k-nct on 3D UMAP

number of cluster is the majority. The number lies in to the number of classes, the quiescence status, derived from patient WSIs.

As depicted in figure 4.10, silhouette scores of clustering by GMM and HC on TCGA-COAD are visualised. The star symbols denoted the optimal number of clusters based on the yielded silhouette score with the highest value in each representation.

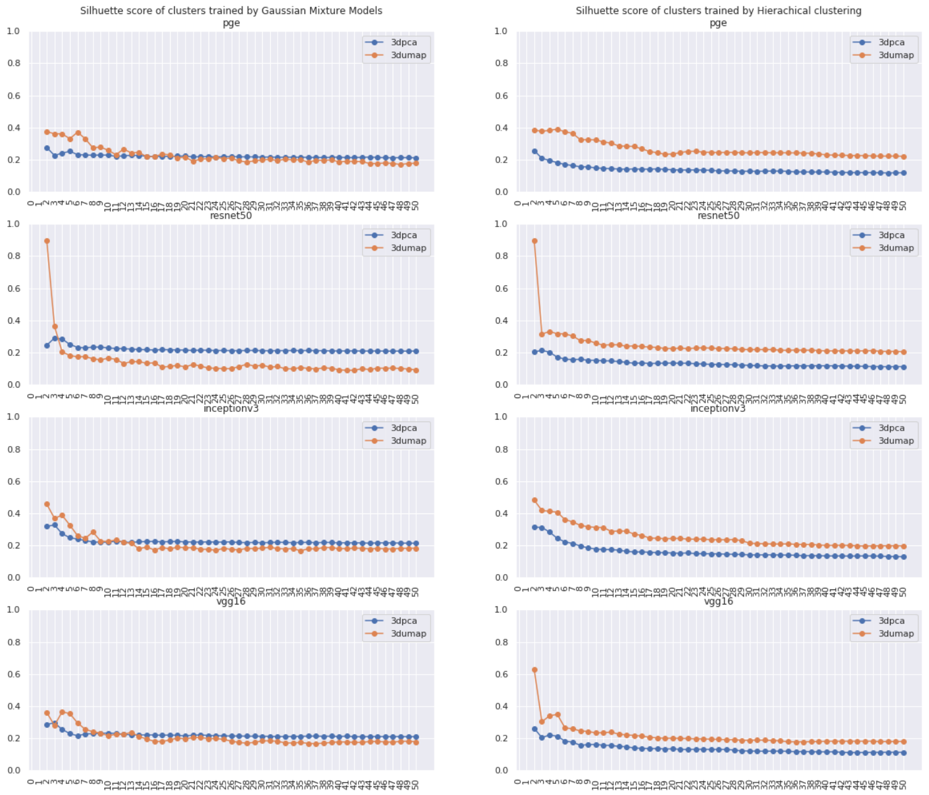


Figure 4.7: Silhouette score of clusters trained by Gaussian Mixture Models and Hierarchical clustering on TCGA-COAD.

4.3 Cluster Quality Evaluation

As two colorectal cancer datasets have pre-defined annotations, cluster evaluation can be conducted by extrinsic measures to evaluate correlation between clustered groups and their designated ground-truth. For extrinsic methods, V-measure is considered to be an inclusive score which takes both homogeneity and completeness into consideration. In this section, each dataset will be examined as follows.

4.3.1 100k-nct cluster quality evaluation

Firstly, how density of samples in each feature space is correlated with data annotation, which is tissue type, was explored. Three extrinsic scores are employed to evaluate the overall performance, which the suggested number of clusters are disregarded. Box plots in Figure 4.11 and 4.12 demonstrate that UMAP (green boxes) is at the top among three forms of dimension transformation. As the result presents overall performance, one conclusion can be drawn from is that UMAP gains more cluster-able representation no matter what is the original representation and what clustering configurations will be.

We further dig down into the quality of clusters of which number of groups suggested by the highest silhouette score. Figure 4.13 reveals results from the same three measures of extrinsic

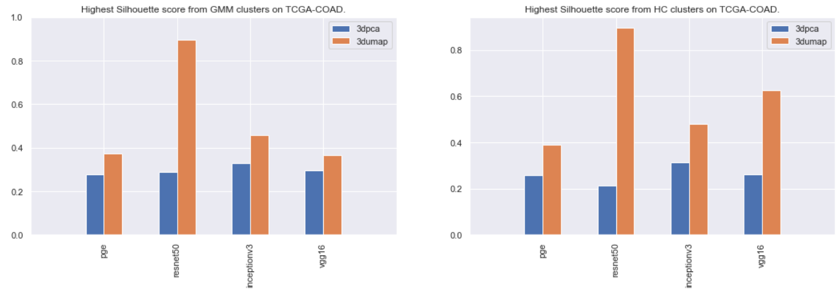


Figure 4.8: Highest Silhouette score from GMM and HC clusters on TCGA-COAD.

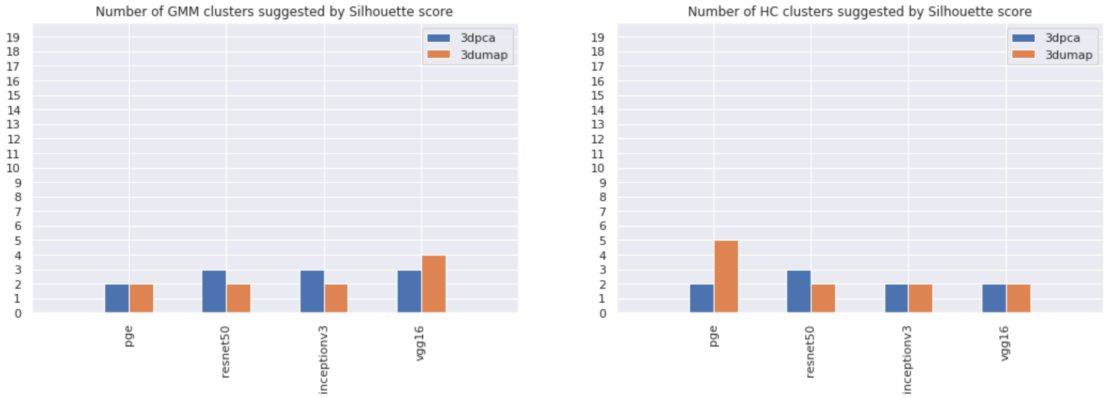


Figure 4.9: Number of clusters suggested by Silhouette score on TGCA-COAD.

scoring including homogeneity, completeness and V-measure. However, only the optimal number of clusters is investigated. According to the bar charts, UMAP still outperforms other forms of representation transformation.

For GMM clustering, optimal-k Restnet 50 gave the most highly homogeneous clusters while the VGG16 counterpart was at the top in basis of complete clusters. On the other hand inceptionV3 is among the lowest in both aspects. Despite different assumptions are behind the the two clustering algorithms, HC generates the similar pattern that GMM does. Except the original feature space of Resnet50 which is scored considerably high in completeness but relatively low in homogeneity. However, referring to the suggested silhouette reported in the previous section in figure 4.5, silhouette score of HC on the Resnet50's original feature is maximised at low number of clusters, which is a common configuration that yields high completeness but low homogeneity.

As V-measure is regarded as the most balanced scoring techniques to consider cluster quality, in addition to the very bottom of the mentioned chart, figure 4.14 summarise the comparison base on V-measure alone. It is perceived that once UMAP is employed to uncover the underlying manifold of 100k-nct dataset on a selected representation, optimal-k Resnet50 significantly outperform the other representations in which results produced by both GMM and HC are comparable. Considering all representation transformed by UMAP, PathologyGAN is only one representation that leads HC to win GMM.

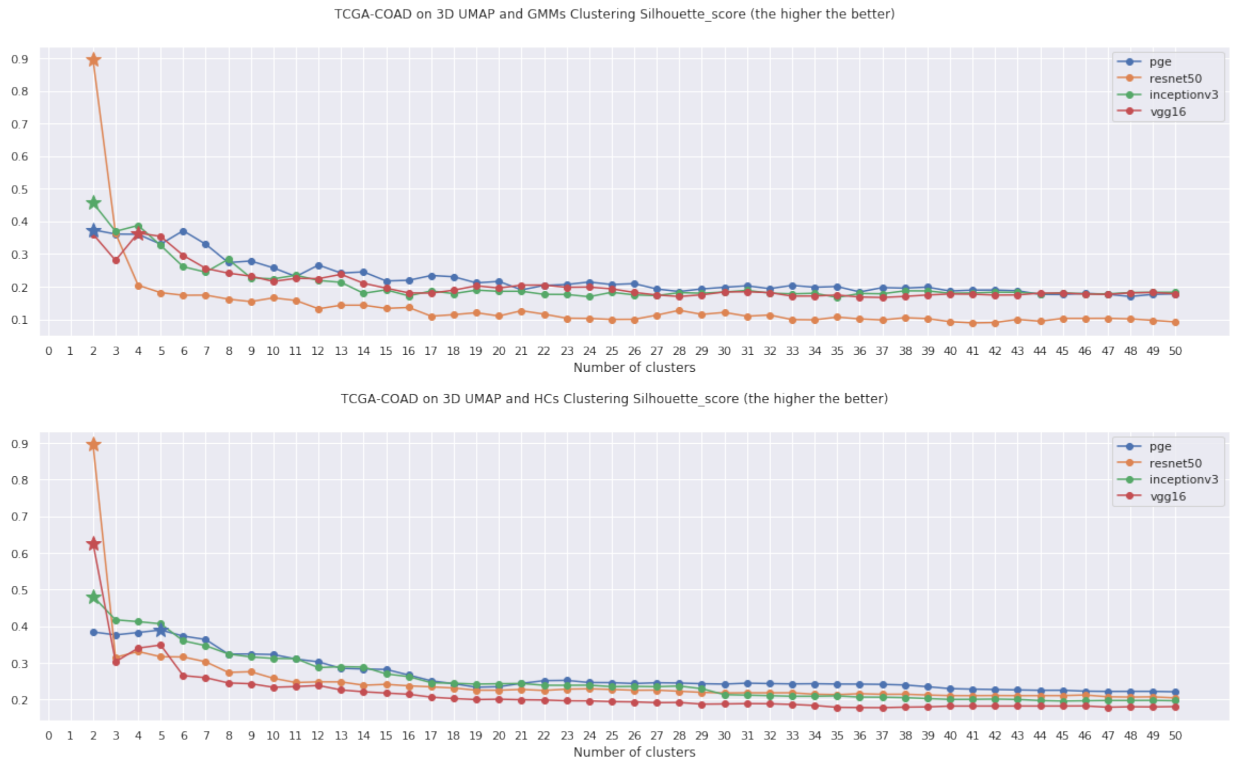


Figure 4.10: Suggest number of cluster by Silhouette score in Gaussian Mixture Models (GMM) and Hierarchical clustering (HC) on TCGA-COAD on 3D UMAP

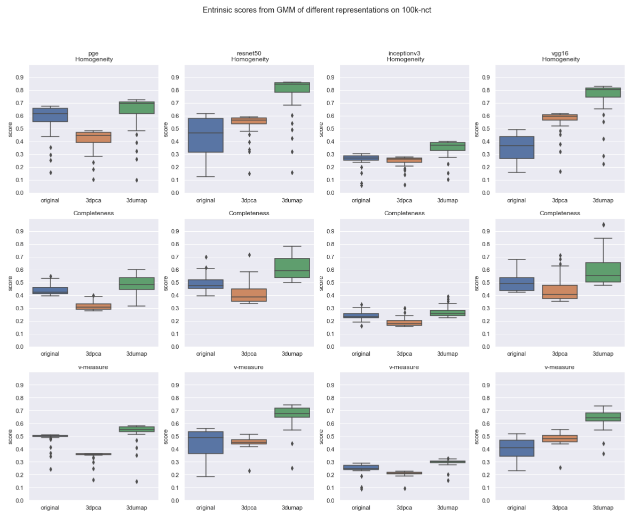


Figure 4.11: Extrinsic scores from GMM of different representations on 100k-nct

4.3.2 TCGA-COAD cluster quality evaluation

Assessing cluster quality is incredibly challenging if no available annotation is directly representing data points in sample-by-sample basis. TCGA-COAD dataset is annotated in the patient level. It is undeniable that high extrinsic score is not to be expected. As shown in figure 4.15. both clustering algorithms produced the relatively low V-measures of all clustering configurations. Surprisingly, PathologyGAN representation which did not win in the 100k-nct clustering

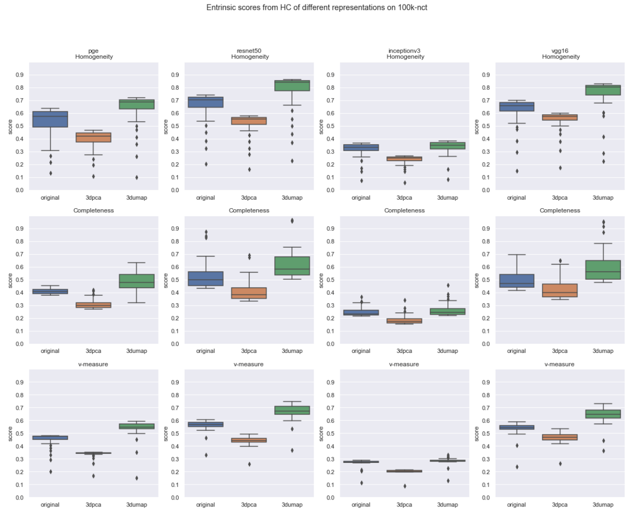


Figure 4.12: Extrinsic scores from HC of different representations on 100k-nct

benchmark reveals a trend of discovering some characteristics while v-measures of the remaining setting are close to zero. However, PathologyGAN itself would not not obtain the wining place. It still required the UMAP transformation to help achieve the improvement margin.

To be more precise, we again investigated the extrinsic performance of clustering of which number of clusters suggested by maximal silhouette scores. The optimal clusters of each configuration are illustrated in figure 4.16. All three extrinsic scores even strengthen the previous finding. Even if number of cluster is selected by intrinsic properties without label influence, PathologyGANs remains outperform the other method. However, that of ResNet50 in which silhouette score stood out in figure 4.7 failed to capture this molecular property at all.

Thus, silhouette score can not be claimed to guaranty the best representation for a specific clustering task but it is still an effective number-of-cluster selector. UMAP is by far a practical transformer collaborating well any clustering paradigm in identifying phenotypic clusters without supervisory biases.

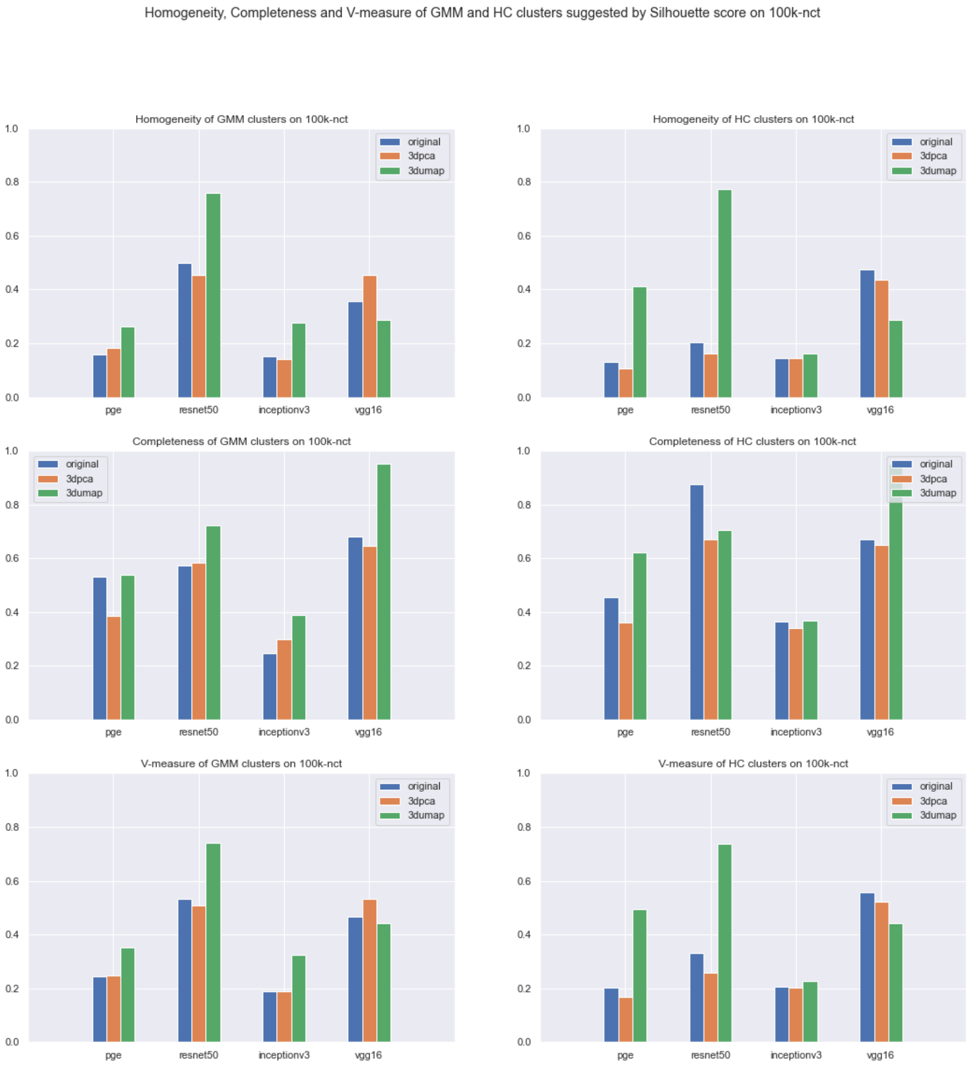
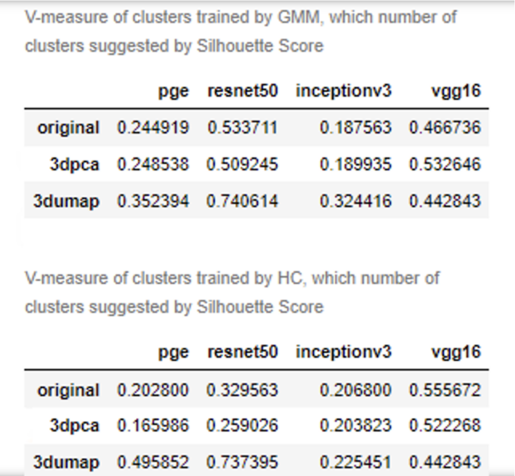


Figure 4.13: Homogeneity, Completeness and V-measure of GMM and HC clusters suggested by Silhouette score on 100k-nct

4.4 Further interpretation

In this study, two colorectal datasets were introduced as a bench-marking problem in tissue patch clustering. As a result, four main independent components which can influence were to examine on this issue including (i) clustering algorithm, (ii) dimension reduction or feature transformation, and (iii) the selected representation. Here, each aspect will be emphasised by information shown in table 4.1



V-measure of clusters trained by GMM, which number of clusters suggested by Silhouette Score

	pge	resnet50	inceptionv3	vgg16
original	0.244919	0.533711	0.187563	0.466736
3dpca	0.248538	0.509245	0.189935	0.532646
3dumap	0.352394	0.740614	0.324416	0.442843

V-measure of clusters trained by HC, which number of clusters suggested by Silhouette Score

	pge	resnet50	inceptionv3	vgg16
original	0.202800	0.329563	0.206800	0.555672
3dpca	0.165986	0.259026	0.203823	0.522268
3dumap	0.495852	0.737395	0.225451	0.442843

Figure 4.14: V-measure of clusters trained by GMM and HC, which number of clusters suggested by Silhouette Score

4.4.1 Discussion on clustering algorithm

As mention in Chapter 2, the two representatives of clustering technique, partitioning-based and hierarchical-based clustering. Although results produced by both GMM and HC are always comparative, there is a to-some-degree noticeable figures which can be supported by fundamental theories of clustering.

GMM as the former category seems to regard the prior-defined Gaussian distribution of data while inspecting data in given dimension space. In 100k-nct benchmak, where no representation extractors were ever trained upon, GMM is slightly better in finding collaboration from rich characteristics extracted by Resnet50. As Resnet50 was trained by one of the biggest and diverged image dataset, ImageNet, it is guaranteed to embed overwhelming varieties of visual feature, which were relevant or vice-versa to tissue images. Result in 4.1 shows that GMM obtained the highest V-measure at 0.741 whereas HC's was little far behind at 0.737.

In TCGA-COAD, extrinsic score can not be as high as expected as annotation derived from patient level. Regardless of feature transformation, both HC and GMM produced equivalent v-measure score, 0.022. the score is incredibly low but it is the highest among all settings.

Although there was a slightly different figure by the two different clustering paradigms, the margin is not significantly big enough to justify the most appropriate one for tissue patch clustering.

4.4.2 Dimension reduction or feature transformation

In term of feature space, each representation was transformed by 3 methods namely original space (no transformation at al), PCA, UMAP for exploring how global and local structures encoded by either linear interpolation properties or as complex as manifold learning can help improve cluster quality. Moreover, in TCGA-COAD dataset, computational constrains limited

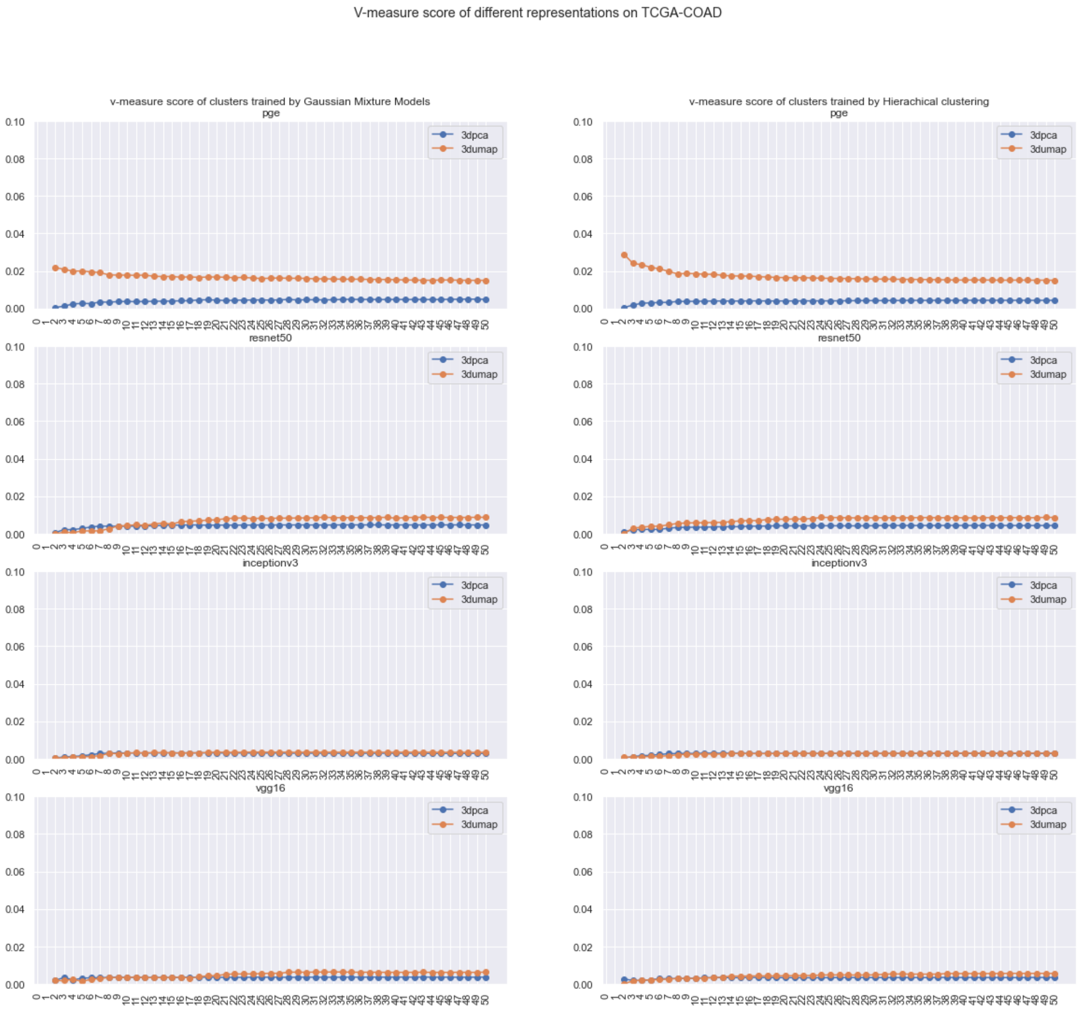


Figure 4.15: V-measure score of different representations on TCGA-COAD

our exploration on clustering such as a big dataset.

Base on evidence shown in 4.1, it is undoubtedly claimed that UMAP yielded considerably better overall performance for all clustering configuration. It is implied that finding of the study [40] is supported. Discovering underlying manifold of dataset is required in order for clustering algorithms to be fully functional in capturing phenotypic properties of tissue image.

4.4.3 Image representation

Four feature extractors were explored in this study namely PathologyGAN, Resnet50, InceptionV3, VGG16. Based on two benchmark datasets, InceptionV3 struggled to achieve any promising findings neither in identifying tissue type nor differentiating quiescent status. It is clearly seen that the clever idea about growing parallel-processing-friendly network depth which solved a complex classification (ImageNet) can not be always transferable to clustering in different domains. Clustering tissue image on InceptionV3 representation yielded poor results in every configuration.

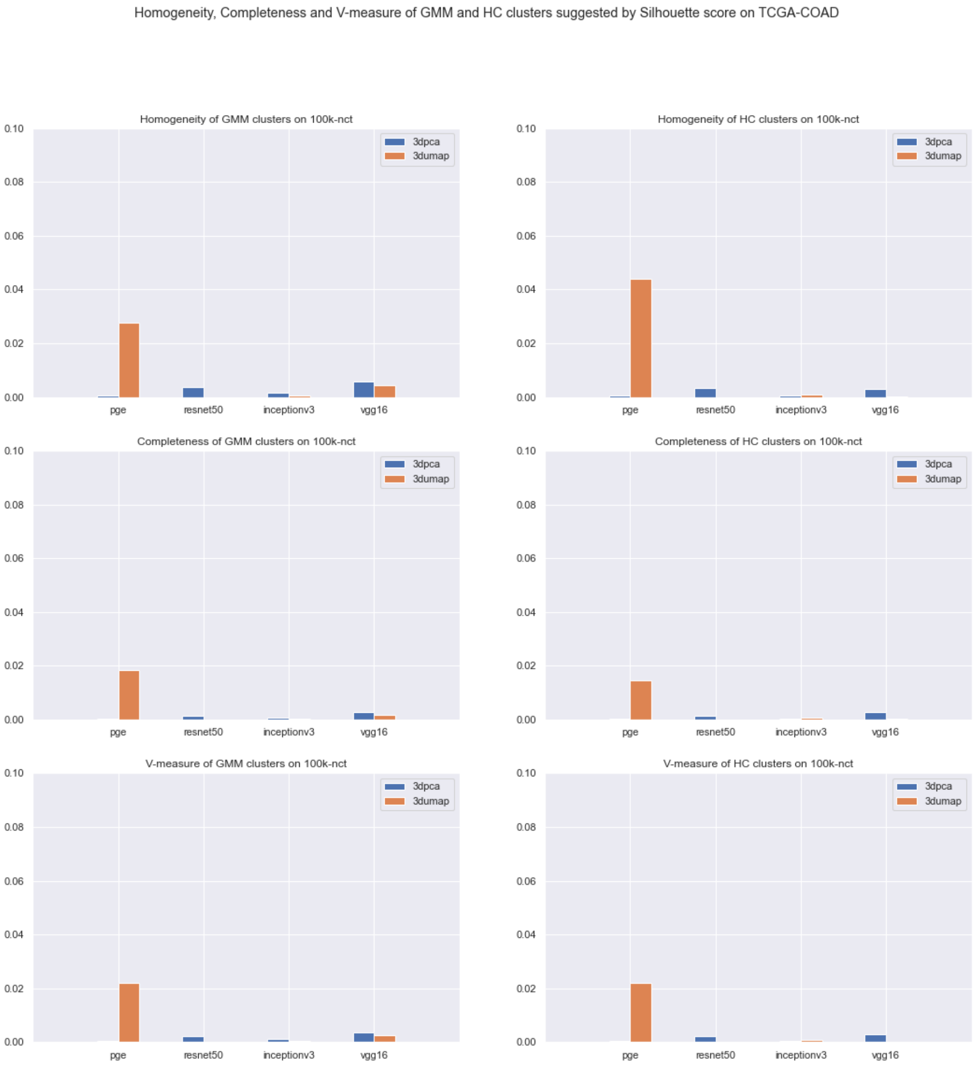


Figure 4.16: Homogeneity, Completeness and V-measure of GMM and HC clusters suggested by Silhouette score on TCGA-COAD

Resnet50 and VGG16 of which architectures were developed in tiny different concepts. They both performed far better than InceptionV3, which was designed for low computational capability. Overall, Resnet50 will be considered as a better candidate of transferable ImageNet's knowledge to tackle tissue type clustering. However, they both failed to capture quiescence-related visual patterns as reported V-measure is nearly zero.

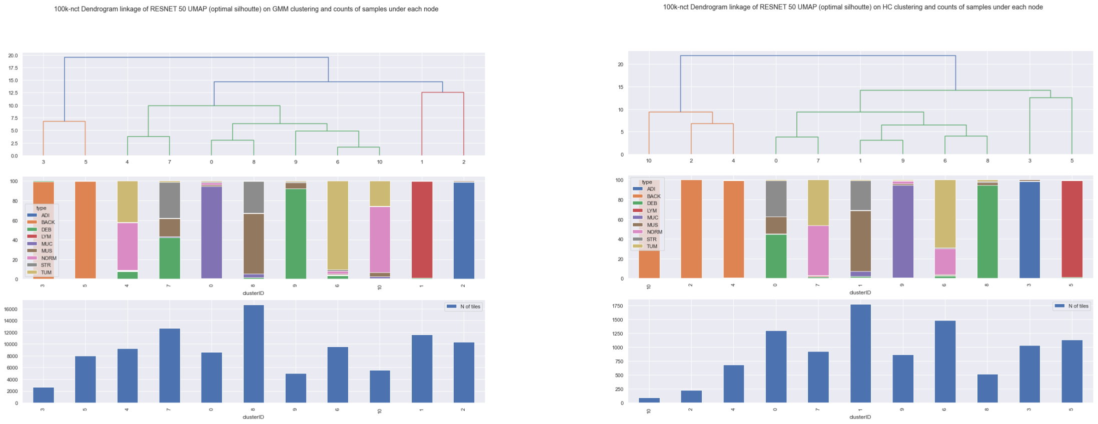
Another feature extractor in this study is PathologyGAN which was trained on TCGA-COAD in an unsupervised generative adversarial manner. It was considered the most potential representation because of domain relevance and an absence of supervisory biases. However, PathologyGAN seems to be under performance when transferred to apply in 100k-nct dataset, compared to Resnet50, which was trained on a dataset of more divered samples. In TCGA-COAD benchmark, other ImageNet-based pre-trained models were complete unsuccessful in capturing quiescence. In contrast, PathologyGAN by far obtained the highest extrinsic performance of capturing quiescence-related visual patterns.

Table 4.1: Summarised experimental result

Dataset	Clustering Algotithm	Dimension Reduction	Feature Extractor	Highest Sihoutte Score	Optimal no. of clusters	Homogeneity	Completeness	V-measure
100K-NCT	Gaussian Mixture Model (GMM)	Original	pathologyGAN	0.252	2	0.159	0.532	0.245
			resnet50	0.125	7	0.500	0.573	0.534
			inceptionv3	0.194	4	0.151	0.247	0.188
			vgg16	0.084	4	0.358	0.679	0.467
		3D PCA	pathologyGAN	0.377	3	0.184	0.384	0.249
			resnet50	0.317	6	0.453	0.582	0.509
			inceptionv3	0.427	3	0.139	0.299	0.190
			vgg16	0.349	5	0.453	0.646	0.533
		3D UMAP	pathologyGAN	0.625	3	0.262	0.537	0.352
			resnet50	0.624	11	0.758	0.724	0.741
			inceptionv3	0.456	5	0.278	0.390	0.324
			vgg16	0.621	3	0.288	0.953	0.443
	Hierarchical -based Clustering (HC)	Original	pathologyGAN	0.263	2	0.131	0.454	0.203
			resnet50	0.176	2	0.203	0.874	0.330
			inceptionv3	0.279	3	0.145	0.363	0.207
			vgg16	0.075	5	0.474	0.671	0.556
		3D PCA	pathologyGAN	0.369	2	0.108	0.360	0.166
			resnet50	0.301	2	0.161	0.671	0.259
			inceptionv3	0.419	3	0.146	0.341	0.204
			vgg16	0.335	5	0.437	0.648	0.522
		3D UMAP	pathologyGAN	0.675	5	0.412	0.622	0.496
			resnet50	0.638	12	0.773	0.705	0.737
			inceptionv3	0.481	3	0.163	0.368	0.225
			vgg16	0.621	3	0.288	0.953	0.443
TCGA-COAD	Gaussian Mixture Model (GMM)	3D PCA	pathologyGAN	0.276	2	0.001	>0.000	0.001
			resnet50	0.290	3	0.004	0.002	0.002
			inceptionv3	0.328	3	0.002	0.001	0.001
			vgg16	0.296	3	0.006	0.003	0.004
		3D UMAP	pathologyGAN	0.374	2	0.028	0.018	0.022
			resnet50	0.897	2	>0.000	>0.000	>0.000
			inceptionv3	0.457	2	>0.000	>0.000	>0.000
			vgg16	0.364	4	0.004	0.002	0.002
	Hierarchical -based Clustering (HC)	3D PCA	pathologyGAN	0.257	2	0.001	>0.000	0.001
			resnet50	0.213	3	0.003	0.002	0.002
			inceptionv3	0.313	2	>0.000	>0.000	>0.000
			vgg16	0.262	2	0.003	0.003	0.003
		3D UMAP	pathologyGAN	0.390	5	0.044	0.015	0.022
			resnet50	0.898	2	>0.000	>0.000	>0.000
			inceptionv3	0.482	2	0.001	0.001	0.001
			vgg16	0.627	2	>0.000	>0.000	>0.000

4.4.4 Explore what inside each and between cluster

To examine relationships inside the cluster, we visualize the recursive merging of two clusters' centroids based on their linkage distance in optimal number of clusters suggested by silhouette score. As shown in Figure 4.17a, RESNET 3D UMAP with GMM is yielded highest silhouette score by 11 number of clusters. We can observe the high prevalence of particular tissue types inside clusters. The proportion of cluster members belonging to the 3 and 4 cluster ids entirely common on BACK subtiles. Moreover, the LYM sub-tiles and ADI sub-tiles have almost perfect cluster member percentages for cluster ids 1 and 2, respectively. each cluster has a distant relationship to an adjacent cluster. Furthermore, 0, 6 and 9 cluster ids are dominated by MUC, TUM and DEB tissue types in high proportion respectively. Not only dominated one tissue type cluster, but there are also the combination of tissue types within a cluster. 4 and 10 cluster ids are



(a) GMM

(b) HC

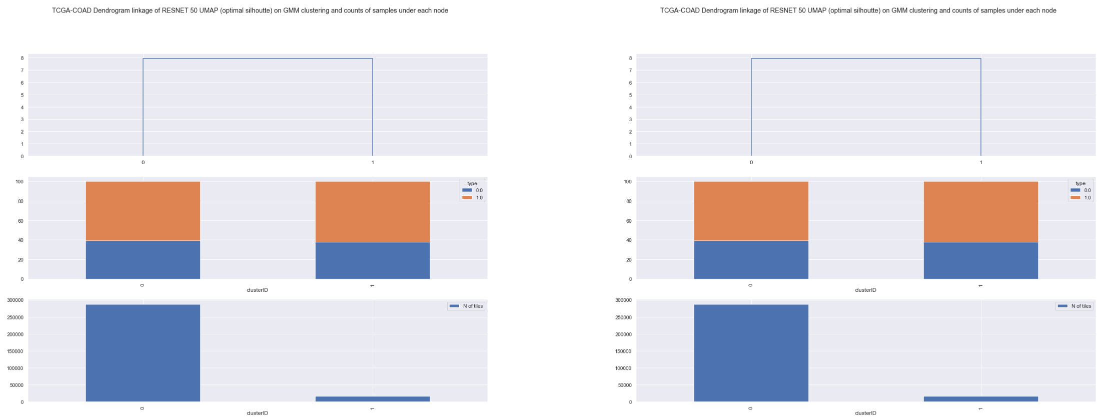
Figure 4.17: 100k-nct Dendrogram linkage of RESNET 50 UMAP from (optimal silhoutte) GMM clustering and counts of samples under each node

share the similar cluster membership between TUM and NORM tissues types. Most of cluster members in 7 and 8 cluster ids share STR and MUS tissue type in their cluster as well. Not only that, 8 cluster ids is also the cluster with the maximum number of sub-tissues allocated by GMM with RESTNET 50 under UMAP dimension reduction, based on the sample count for each cluster.

From previous viewpoint with GMM, there is also visible similarly in Figure 4.17b that show linkage cluster of RESNET 50 UMAP with HC clustering in dendrogram perspective from some random sample from 100k-nct sub tissues. 11 number of clusters also yield highest silhouette score. the 2, 4 and 10 cluster ids show abundance of the cluster members belonging to BACK sub tiles. They also share the similar linkage of hierarchical structure. The other pure clusters from their tissue type's cluster membership are 3, 5, 8 and 9 cluster ids that contain almost unique of ADI, LYM, DEB and MUC tissue types. In addition, there exist mixtures of tissue types inside a cluster. 6 and 7 cluster ids are share the similar cluster membership between TUM and NORM tissues types. The majority of cluster members in cluster ids 0 and 1 have STR and MUS tissue types, although there is a relatively large number of DEB tissue type inside cluster id 0.

Figures 4.18a and 4.18b of TCGA-COAD illustrate dendrograms of RESNET 50 UMAP with the highest silhouette score on GMM and HC clustering. In both algorithms, the number of clusters that produced the highest silhouette score is 2. When examining individual clusters, the percentage of positive and negative quiescence subtissue labels is nearly equivalent in both. And it correlate with preceding part, the v-measure score of TCGA-COAD essentially reflected zero in RESTNET representation. Moreover, when we investigate the sample counts under each node, we see that the number of samples between the two clusters varies significantly.

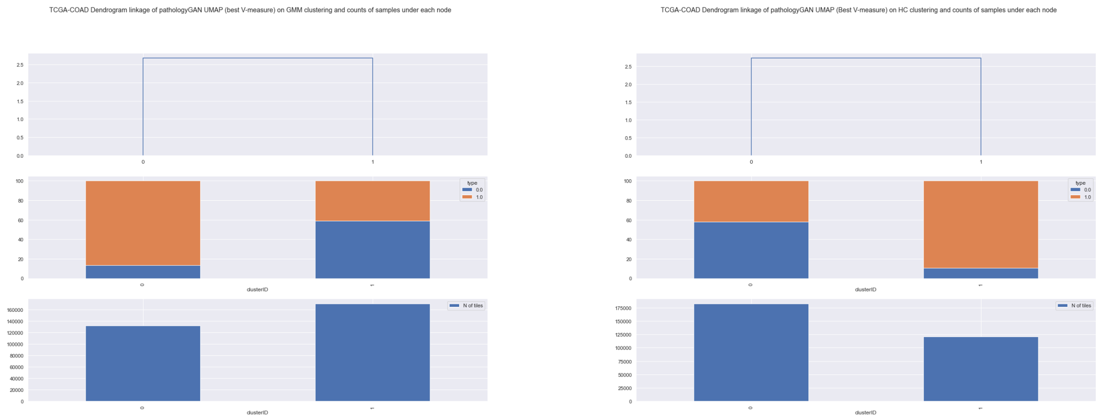
Achieving higher silhouette under image representation does not ensure achieving in extrinsic evaluation. Therefore, we examine the best v-measure score produced by the experiment



(a) GMM

(b) HC

Figure 4.18: TCGA-COAD Dendrogram linkage of RESNET 50 UMAP (optimal silhoutte) on GMM and HC clustering and counts of samples under each node



(a) GMM

(b) HC

Figure 4.19: TCGA-COAD Dendrogram linkage of pathologyGAN UMAP (best V-measure) on GMM clustering and counts of samples under each node

to determine its cluster qualities. When we investigate pathologyGAN UMAP with clustering results that provide the best v-measure score on TCGA-COAD. Comparing it to 3D UMAP RESTNET 50 to figure 4.19 that showed the percentage of cluster members, the quiescence state label separate better even its very low silhouette sore. Positive quiescence cluster membership is considerably dominant in 0 cluster ids and 1 HC cluster ids in GMM. And PathologyGAN potentially balance the number of samples in each cluster. UMAP can be effective to unfolding latent feature in pathologyGAN.

To analyse the cluster assignments on the WSI of a patient to determine how well the cluster assignments can characterize the WSI characteristics of a patient with derived quiescence. Utilising 3D UMAP pathologyGAN on GMM to produce Figure 4.20. The blue overlay represents 0 cluster ids while the red overlay represents 1 cluster ids. When comparing positive and negative quiescence patient WSI, there is no meaningful difference between the cluster assignment. As experimental result, the extrinsic score cannot be as high as the annotation produced from the

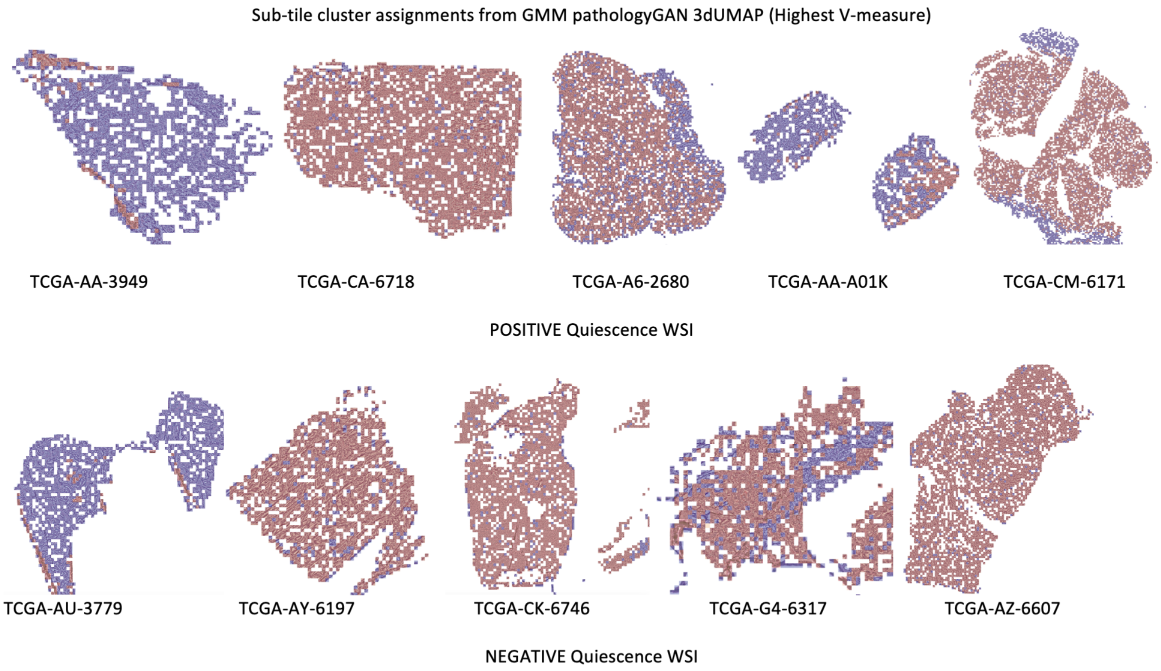


Figure 4.20: Sub-tile cluster assignments from GMM pathologyGAN 3dUMAP (Highest V-measure)

patient level, as described before. so, their cluster assignments can not capture its charesteristics of patient quienscen state as well.

Chapter 5

Conclusion

5.1 Summary of thesis contributions and experiments

This research presents a clustering approach to identifying phenotype of tissue slides. The cluster setting was designed to work on tissue tile, patched from Whole Slide Image (WSI). The ultimate goal of this study is to find a generic approach to obtaining 'high-quality clusters' of tissue patches with respect to phenotypic classes or perhaps representing phenotype of WSI.

The experiment was conducted in 4 stages with two colorectal cancer datasets, 100k-nct with annotated tissue types and TCGA-COAD with patient quiescence. (i) Representation of tissue tile: 4 different representation extractors were categorised into 2 groups. ImageNet-based Pre-trained CNNs including Resnet50, InceptionV3, VGG16 and an unsupervised generative model PathologyGAN were trained on TCGA-COAD dataset. each instance of tissue patches were feature-extracted for further modelling. (ii) dimension reduction or feature transformation: each representation of data is processed into 3 different forms of feature transformation including original feature, 3D-PCA feature and 3D-UMAP feature to figure out which approach of discovering underlying clusterable structure. (iii) cluster configuration: partition-based clustering and hierarchical-based clustering were explored in varying number of clusters (iv) how optimal clusters were selected and interpreted : the optimal number of cluster was suggested by silhouette score of cross-validation. Cluster visualisation and extrinsic measure e.g. V-measure were employed to evaluate cluster quality in real-life purpose.

5.2 Response to research questions

Four research questions mentioned in Chapter 1 will be responded here. First, main components to be together constitute a clustering framework including feature representation, feature transformation for improving clusterablity, clustering algorithm, intrinsic measure to be as configuration e.g. number of cluster identifier. According to the study, these components were important success factors of identifying tissue type in 100k-nct data without any influence of ground-truth.

Second, quiescence is the only molecular property being explored in this study. No v-measure sounds enough to say that solid pattern is captured and the association between cluster memberships and molecular profile is obligated to further investigate. However, the configuration with maximal v-measure was revealed in Dendrogram that one of the clusters dominated by tiles obtained from patient with positive quiescence. Third, silhouette score is an effective measurement to decide cluster configuration especially number of clusters. Notwithstanding, it is yet to be concluded that the best candidate of representation can be reliant on silhouette score. Forth, Restnet50 seems to be robust on data of which domain is irrelevant to source domain, from ImageNet to colorectal cancer dataset (100k-nct). However, in case PathollogyGAN has a chance to be trained in an unsupervised adversarial manner on the working dataset, TCGA-COAD, it potentially solves a complex problem lying in the dataset. Finally, regarding the study, manifold learning such as UMAP plays an important role in preparing a more clusterable representation. It helps improve significant performance of clustering based on extrinsic scores. The scoring considered both direct label e.g. tissue type and patient's derived label e.g. quiescence as the ground-truth.

5.3 Limitations and suggested future research works

There are three main areas which have not been fully achieved under this study.

1. The relationship between tile assignments of this phenotypic clusters and classification performance of patient clinical profile was not given enough attention in this study. More patient metadata related to their clinical and molecular status could be investigated based on clustering perspectives.

2. Although silhouette score is incredibly effective in identifying clustering hyper-parameters, no measure has been founded yet in selecting the most appropriate feature extractor without the suggestion of ground-truth. a generic cluster quality measurement could be an attention grabbing research topic in this area.

3. Phenotypic clusters are supposed to be proven the applicability across patient cohorts with only minor fine-tuning. Mechanisms which bring this ability to the clustering framework should be experimented and discussed towards a fully automate histopathology-based cancer diagnosis.

Bibliography

[1] C. Kelly, A. Karthikesalingam, M. Suleyman, G. Corrado and D. King, *Key challenges for delivering clinical impact with artificial intelligence.* BMC medicine 17.1 (2019): 1-9

[2] M. Chen and H. Zhao, *Next-generation sequencing in liquid biopsy: cancer screening and early detection.* Human genomics 13.1 (2019): 1-10.

[3] K. Bera, K. A. Schalper, D. L. Rimm, V. Velcheti and A. Madabhushi, *Artificial intelligence in digital pathology—new tools for diagnosis and precision oncology.* Nature reviews Clinical oncology 16.11 (2019): 703-715.

[4] T. J. Fuchs and J. M. Buhmann, *Computational pathology: challenges and promises for tissue analysis.* Computerized Medical Imaging and Graphics 35.7-8 (2011): 515-530.

[5] C. Xie, H. Muhammad, C. M. Vanderbilt, R. Caso, D. Yarlagadda, G. Campanella and T J. Fuchs, *Beyond classification: Whole slide tissue histopathology analysis by end-to-end part learning.* Medical Imaging with Deep Learning. PMLR, 2020.

[6] M. Dash and H. Liu, *Feature selection for clustering.* Pacific-Asia Conference on knowledge discovery and data mining. Springer, Berlin, Heidelberg, 2000.

[7] R. W. Sembiring, J. M. Zain and A. Embong, *Dimension reduction of health data clustering.* arXiv preprint arXiv:1110.3569 (2011).

[8] P. W. Hamilton et al, *Digital pathology and image analysis in tissue biomarker research.* Methods 70.1 (2014): 59-73

[9] P. D. Caie et al, *Novel histopathologic feature identified through image analysis augments stage II colorectal cancer clinical reporting.* Oncotarget 7.28 (2016): 44381.

[10] C. Janiesch, P. Z. Christian and K. Heinrich. *Machine learning and deep learning.* Electronic Markets 31.3 (2021): 685-695.

[11] J. Deng, *A large-scale hierarchical image database.* Proceeding of IEEE Computer Vision and Pattern Recognition (2009).

[12] M. A. Morid, A. Borjali and G. D. Fiol, *A scoping review of transfer learning research on medical image analysis using ImageNet.* Journal of Computers in biology and medicine (2021).

[13] L. Alzubaidi et al, *Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions.* Journal of Big Data 8.1 (2021): 1-74.

[14] A. Janowczyk and A. Madabhushi, *Deep learning for digital pathology image analysis: A comprehensive tutorial with selected use cases.* Journal of pathology informatics 7 (2016).

[15] B. E. Bejnordi et al, *Diagnostic assessment of deep learning algorithms for detection of lymph node metastases in women with breast cancer.* Jama 318.22 (2017): 2199-2210.

[16] BACH ICIAR, *Grand Challenge on Breast Cancer Histology Images. 2018.* (2018)

[17] O. Ronneberger, P. Fischer and T. Brox, *U-net: Convolutional networks for biomedical image segmentation.* International Conference on Medical image computing and computer-assisted intervention. Springer, Cham, 2015

[18] S. Graham et el, *Hover-net: Simultaneous segmentation and classification of nuclei in multi-tissue histology images.* Medical Image Analysis 58 (2019): 101563.

[19] K. Yao, N. D. Rochman and S. X. Sun, *Cell type classification and unsupervised morphological phenotyping from low-resolution images using deep learning.* Scientific reports 9.1 (2019): 1-13.

[20] R. Wetteland et al, *Multiscale deep neural networks for multiclass tissue classification of histological whole-slide images.* arXiv preprint arXiv:1909.01178 (2019).

[21] A. Stenzinger et al, *Artificial intelligence and pathology: From principles to practice and future applications in histomorphology and molecular profiling.* Seminars in cancer biology. Academic Press, 2021.

[22] K. Ikromjanov et al, *Multi-class Classification of Histopathology Images using Fine-Tuning Techniques of Transfer Learning.* Journal of Korea Multimedia Society 24.7 (2021): 849-859.

[23] K. Medjaher el at, *Feature extraction and evaluation for Health Assessment and Failure prognostics.* Proceedings of First European Conference of the Prognostics and Health Management Society, PHM-E'12. Anibal Bregon, Abhinav Saxena, 2012.

[24] Y. Bengio, A. C. Courville, and P. Vincent, *Unsupervised feature learning and deep learning: A review and new perspectives.* CoRR, abs/1206.5538 1 (2012): 2012.

[25] J. Pardede et al, *Implementation of transfer learning using VGG16 on fruit ripeness detection.* International Journal of Intelligent Systems And Applications 13.2. (2021).

[26] A. Sagar and D. Jacob, *On using transfer learning for plant disease detection.* bioRxiv (2021): 2020-05.

[27] G. J. Chowdary et al, *Face mask detection using transfer learning of inceptionv3.* International Conference on Big Data Analytics. Springer, Cham, 2020.

[28] V. Arora et al, *Transfer learning-based approach for detecting COVID-19 ailment in lung CT scan.* Computers in Biology and Medicine 135 (2021): 104575.

[29] K. J. Holyoak, *Parallel distributed processing: explorations in the microstructure of cognition.* Science 236 (1987): 992-997.

[30] D. P. Kingma and M. Welling, *Auto-encoding variational bayes.* arXiv preprint arXiv:1312.6114 (2013).

[31] I. Guyon et al, *Unsupervised and transfer learning challenge.*The 2011 International Joint Conference on Neural Networks. IEEE, 2011.

[32] A. C. Quiros, R. Murray-Smith and K. Yuan, *Learning a low dimensional manifold of real cancer tissue with PathologyGAN.* arXiv preprint arXiv:2004.06517 (2020).

[33] A. K. Mann and N. Kaur, *Review Paper on Clustering Techniques.* Global Journal of Computer Science and Technology. 13.5 (2013) [Online]:

[34] P. Prasad , *Hierarchical clustering explained* [Online]. 2021. retrieved from : https://towardsdatascience.com/hierarchical-clustering-explained-e59b13846da8

[35] C. Maklin , *Gaussian Mixture Models Clustering Algorithm Explained* [Online]. 2021. retrieved from : https://towardsdatascience.com/gaussian-mixture-models-d13a5e915c8e

[36] R. Bhadauria , *ML | Expectation-Maximization Algorithm* [Online]. 2019. retrieved from : https://www.geeksforgeeks.org/ml-expectation-maximization-algorithm

[37] D. A. Reynolds , *Gaussian mixture models* Encyclopedia of biometrics 741.659-663 (2009).

[38] V. Molchanov and L. Linsen, *Overcoming the Curse of Dimensionality When Clustering Multivariate Volume Data.* VISIGRAPP (2018).

[39] P. Prabhu and N. Anbazhagan, *Improving the performance of k-means clustering for high dimensional data set.* International journal on computer science and engineering (2011).

[40] R. McConville et al, *N2d:(not too) deep clustering via clustering the local manifold of an autoencoded embedding.* 2020 25th International Conference on Pattern Recognition (ICPR). IEEE, 2021.

[41] M. Halkidi and M. Vazirgiannis, *Clustering validity assessment: Finding the optimal partitioning of a data set.* Proceedings 2001 IEEE international conference on data mining. IEEE, 2001.

[42] J. Shuja et al, *Resource efficient geo-textual hierarchical clustering framework for social iot applications.* IEEE Sensors Journal 21.22 (2021): 25114-25122.

[43] F. S. Hoseini, S. Rahrovani and M. H. Chehreghani, *A generic framework for clustering vehicle motion trajectories.* arXiv preprint arXiv:2009.12443 (2020).

[44] M. Ciortan and M. Defrance, *Contrastive self-supervised clustering of scRNA-seq data.* BMC bioinformatics 22.1 (2021): 1-27.

[45] E. Amigó, J. Gonzalo, J. Artiles and F. Verdejo, *A comparison of extrinsic clustering evaluation metrics based on formal constraints.* Information retrieval 12.4 (2009): 461-486.

[46] J. Hämäläinen, S. Jauhiainen and T. Kärkkäinen, *Comparison of Internal Clustering Validation Indices for Prototype-Based Clustering.* Algorithms 2017, 10, 105. https://doi.org/10.3390/a10030105

[47] L. Rokach, O. Maimon (2006). *Clustering methods.* Data Mining and Knowledge Discovery Handbook Springer. pp. 321–352. ISBN 978-0-387-25465-4.

[48] J. Yadav and M. Sharma, *A Review of K-mean Algorithm.* Int. J. Eng. Trends Technol 4.7 (2013): 2972-2976. APA

[49] K. P. Murphy, *Machine Learning: A Probabilistic Perspective* MIT Press, Cambridge, Mass, (2012)

[50] M. Deisenroth, A. Faisal, C. Ong, (2020) *Mathematics for Machine Learning* Cambridge University Press.

[51] A. P. Dempster, N. M. Laird and D. B. Rubin *Maximum Likelihood from Incomplete Data via the EM Algorithm* Journal of the Royal Statistical Society, 39(1), 1–38. 1977

[52] C. D. Manning, P. Raghavan and H. Schütze, (2008) *Introduction to Information Retrieval*Cambridge University Press.

[53] I. Goodfellow et al, *Generative Adversarial Nets* Proceedings of the International Conference on Neural Information Processing Systems (NIPS 2014). pp. 2672–2680. 2014

[54] A. C. Quiros, N. Coudray, A. Yeaton, W. Sunhem, R. Murray-Smith, A. Tsirigos and K. Yuan, *Adversarial learning of cancer tissue representations* arXiv preprint arXiv:2108.02223.

[55] A. Brock, J. Donahue, and K. Simonyan, *Large scale GAN training for high-delity natural image synthesis* In International Conference on Learning Representations, 2019.

[56] A. Jolicoeur-Martineau, *The relativistic discriminator: a key element missing from standard GAN* In International Conference on Learning Representations, 2019.

[57] T. Karras, S. Laine, and T. Aila, *A style-based generator architecture for generative adversarial networks* 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Jun 2019. doi: 10.1109/cvpr.2019.00453

[58] K. He, X. Zhang, S. Ren and J. Sun, *Deep residual learning for image recognition* 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Jun 2016. doi: 10.1109/cvpr.2016.90.

[59] F. Rousseau, L. Drumetz and R. Fablet, *Residual Networks as Flows of Diffeomorphisms* Journal of Mathematical Imaging and Vision. 62. 2020 10.1007/s10851-019-00890-3.

[60] K. Simonyan and A. Zisserman, *Very deep convolutional networks for large-scale image recognition* In ICLR, 2015.

[61] T. Vo, T. Nguyen and C. T. Le *Race Recognition Using Deep Convolutional Neural Networks* Symmetry, 10.11, 2018. doi: 10.3390/sym10110564

[62] L. Khuyen, *An overview of VGG16 and NiN models* [Online], 2021. retrieved from https://medium.com/mlearning-ai/an-overview-of-vgg16-and-nin-models-96e4bf398484.

[63] V. Sze, Y. H. Chen, T. J. Yang and J. S. Emer, *Efficient Processing of Deep Neural Networks: A Tutorial and Survey.* arXiv 2017, arXiv:1703.09039.

[64] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. *Going deeper with convolutions* In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 1–9, 2015.

[65] A. Bandodker, *Inception V1 Architecture Explained* [Online]. 2020. retrieved from https://medium.com/@abheerchrome/inception-v1-architecture-explained-454b2eb66baf

[66] S. Zorgui el at, *A Convolutional Neural Network for Lentigo Diagnosis.* In International Conference on Smart Homes and Health Telematics. Springer, Cham, 2020. p. 89-99

[67] A. Rosenberg and J. Hirschberg , *V-Measure: A Conditional Entropy-basedExternal Cluster Evaluation Measure* In Proceedings of the 2007 Joint Confer-ence on Empirical Methods in Natural Language Processing and ComputationalNatural Language Learning (EMNLP-CoNLL), pages 410–420, 2007.

[68] C. J. Van Rijsbergen. *Information Retrieval* 2nd edition.Dept. of Computer Science, University of Glasgow. 1979.

[69] P. J. Rousseeuw. *Silhouettes: a Graphical Aid to the Interpretation and Validation of Cluster Analysis* Computational and Applied Mathematics. 20: 53–65. doi:10.1016/0377-0427(87)90125-7

[70] K. R. Shahapure and C. Nicholas. *Cluster Quality Analysis Using Silhouette Score* 2020 IEEE 7th International Conference on Data Science and Advanced Analytics (DSAA), 2020, pp. 747-748, doi: 10.1109/DSAA49011.2020.00096.

[71] B. Cheng, L. Zhuo and J. Zhang, *Comparative Study on Dimensionality Reduction in Large-Scale Image Retrieval* 2013 IEEE International Symposium on Multimedia, 2013, pp. 445-450, doi: 10.1109/ISM.2013.86.

[72] R. Aziz, C. K. Verma and N. Srivastava, *Dimension reduction methods for microarray data: a review* AIMS Bioengineering, 4(2), 179-197.

[73] S. Ayesha, M. K. Hanif and R. Talib, *Overview and comparative study of dimensionality reduction techniques for high dimensional data.* Information Fusion, Vol 59, 44-58, 2020. https://doi.org/10.1016/j.inffus.2020.01.005.

[74] K. Pearson , *On Lines and Planes of Closest Fit to Systems of Points in Space* Philosophical Magazine. 2 (11): 559–572. 1901. doi:10.1080/14786440109462720.

[75] L. McInnes , J. Healy and J. Melville, *Umap: Uniform manifold approximation and projection for dimension reduction* arXiv preprint arXiv:1802.03426.

[76] M. Berger, *Riemannian Geometry During the Second Half of the Twentieth Century* University Lecture Series, vol. 17, Rhode Island: American Mathematical Society. 2000

[77] L. McInnes, *How UMAP works* [Online]. 2018. https://umap-learn.readthedocs.io/en/latest/

[78] J. Kather el al. *100,000 histological images of human colorectal cancer and healthy tissue (v0.1) [Dataset]*.Zenodo.2018. https://doi.org/10.5281/zenodo.1214456

[79] S. Kirk et al, *Radiology Data from The Cancer Genome Atlas Colon Adenocarcinoma [TCGA-COAD] collection.* The Cancer Imaging Archive. http://doi.org/10.7937/K9/TCIA.2016.HJJHBOXZ

[80] H. Nima et al. *Deep multi-resolution dictionary learning for histopathology image analysis*. arXiv preprint arXiv:2104.00669, 2021.

[81] M. Macenko et al, *A method for normalising histology slides for quantitative analysis* 2009 IEEE International Symposium on Biomedical Imaging: From Nano to Macro, 2009, pp. 1107-1110, doi: 10.1109/ISBI.2009.5193250.

[82] J. Han et el. *Cluster Analysis: Basic Concepts and Methods*. In The Morgan Kaufmann Series in Data Management Systems, Data Mining (Third Edition). 443-495. 2012.ISBN 9780123814791