Technical University of Denmark

DTU compute

Triple negative breast cancer

A data-driven learning of clinical subtypes

by

Thomas Leicht Jensen

TNBC
Master's Thesis (30 ECTS)
February, 2020
DTU supervisor: Professor, Anders Bjornholm Dahl
Visiopharm supervisor: Industrial PhD fellow, Jeppe Thagaard

Preface and acknowledgements

Preface: This thesis is submitted and is part of partial fulfilment of the requirements for the Master of Science in Engineering (Biomedical Engineering) programme at the Technical University of Denmark (Danmarks Tekniske Universitet, DTU). The thesis began during the autumn semester of 2019 and is made in collaboration between the Technical University of Denmark and Visiopharm A/S. Visiopharm produces image analysis software for histopathology and uses AI and deep learning to research new technologies in digital augmented pathology.
This thesis works with histopathology from tumor biopsies and throughout this thesis terms are used on two hierarchical levels. On the higher level, a patient is from a data-context represented by a whole slide image. A patient slide and a patient refers to the same high-level description. The lower hierarchical level refers to the images, referred to as tiles, extracted from the whole slide. In this context images and tiles refer to the same lower-level data representation. The term tile has, is to the farthest extend used as the preferable low-level term, but image and tile may be used interchangeably.

Acknowledgements: I would like to thank my supervisors Professor Anders Bjorholm Dahl and Industrial PhD fellow Jeppe Thagaard for their guidance throughout the process of working with this thesis. An additional thanks goes to Morten Hannemose, DTU, for his valuable and highly qualified inputs to the discussions and various aspects in this thesis. I would also like to thank Simon Haastrup and Lulu Wong for their invaluable feedback proof reading this thesis.

List of terms

AE: Autoencoder
ACC: Accuracy (unsupervised clustering accuracy)
AJIVE: Angle based Joint and Individual Variation Explained
AMI: Adjusted mutual information
ANN: Artificial neural network
ARI: Adjusted Rand Index
BC: Breasat cancer
BRCA I, II: Tumor suppressor genes
CAE: Convolutional autoencoder
CHI: Calinski-Harabasz index
CI: Confidence interval (95% CI default)
CPH: Cox proportional hazard
CR: Compression ratio
DALY: Disability-adjusted life year
DBI: Davies–Bouldin index
DCEC: Deep convolutional embedded clustering
EPV: Event per variable
FN: False negative
FP: False positive
HE: Hematoxylin & eosine
HER2: Human epidermal growth factor receptor 2
HR: Hazard ratio
KL: Kullback–Leibler
Lehmann: 4 histological subtypes of TNBCs:

- **BL1**: Basal-like 1
- **BL2**: Basal-like 2
- **M**: Mesenchymal
- **LAR**: Luminal Androgen Receptor

LR: Learning rate
MI: Mutual information
ML: Maximum likelihood
MSE: Mean squared error
NMI: Normalized mutual information
PAM50: 50 genetic cancer test to tumor subtyping:

- **LumA**: Luminal A
- **LumB**: Luminal B

Her2: HER2 enriched

Basal-like: Basal-like

Normal-like: Normal-like

PCA: Principal component analysis
PCR: Pathological complete response
RCB: Residual cancer burden
RGB: Red green blue
RI: Rand index
ROI: Region of interest
SSE: Silhouette score
TNBC: Triple negative breast cancer
lrTNBC: late response TNBC
nrTNBC: no response TNBC
rrTNBC: rapid response TNBC

TCGA: The Cancer Genome Atlas
TIL: Tumor infiltrating lymphocyte
TP: True negative
TP: True positive
tsne: T-distributed Stochastic Neighbor Embedding
WSI: Whole slide image

Abstract: Triple negative breast cancer (TNBC) is a breast cancer subtype that lacks estrogen and progesterone receptors and HER2 overexpression, that accounts for about 16% of all breast cancer cases (5). TNBC is a heterogeneous group considered difficult to treat with a 62.5% 5-year overall survival compared to 80.8% for non-TNBCs (13).

The purpose of this thesis is to investigate unsupervised learning methods that cluster tumor histology for TNBCs and relate the clustering outcome to survival analysis. Two unsupervised clustering methods are investigated and compared: Naive K-means and Deep Convolutional Embedded Clustering (DCEC).

To assess the quantitative clustering capabilities for the two methods, two labeled datasets (MNIST and Tissues) are used to train and test Naive K-means and DCEC. MNIST consists of handwritten digits and Tissues consists of 4 tissue types: fat, lymphocytes, stroma and tumor. Upon evaluation of external metrics, DCEC proves to be a better clustering method compared to the simpler Naive K-means for both datasets.

Experiments by varying the number of clusters K and clustering weight γ for the Tissues dataset suggests a color dependency in which DCEC clusters the Tissues dataset. Consequently, the Tumor (n=98 patients) dataset is stain normalized using Macenko normalization (26) to account for stain variations.
A subset to the Tumor dataset with hyperparameters K=4 and γ=0.6 is used in cluster analysis. Clustering correlations to PAM50 genetics, Lehmann histology as well as tumor and overall disease staging are investigated. Results show little variation and no identifiable patterns appear stratifying to PAM50, Lehman or tumor and neoplasm disease stages.

The clustering outcome of the tumor tiles is used as a covariate to model survival analysis of the patients. The covariate is modelled in two ways: continuous and binary. The result from the continuous modelling shows no significant relation between survival outcome and the clusters, the binary result shows significantly better prognosis to patients positive for cluster 0. Plotting the survival curves of cluster 0 positive patients (n=92) against all other patients (n=6) also shows a significantly better prognosis to cluster 0 positive patients.

Extracting the most likely tiles from each cluster shows a tendency towards stain variation residuals from the patient groups - there is a possible confounding between clusters and stain residuals. Results from the survival analysis shows significantly better outcomes for patients positive to cluster 0, but the results should be interpreted cautiously because of a possible stain confounding distorting the analysis of the clustering outcome. Further work needs to be carried out improving the data pre-processing steps as well as implementing a clustering method that is robust against stain variations in order to cluster tumor tiles of histological relevance.

Resume: Triple negativ brystkræft (TNBC) er en undertype af brystkræft, som mangler østrogen- og progesteronreceptorer samt HER2-overekspression. 16% af alle brystkræft patienter er TNBC (5). TNBC er en heterogen undertype, som er vanskelig at behandle og har en lavere 5 års overlevesrate på 62.5% sammenlignet med 80.8% for ikke-TNBC (13).

Formålet med dette speciale er at undersøge unsupervised learning metoder til clustering af tumor histologi og relatere dette med overlevelses-analyse. To metoder for unsupervised clustering metoder undersøges og sammenlignes: Naive K-means og Deep Convolutional Embedded Clustering (DCEC).

To datasæt med labels (MNIST og Tissues) bruges til at træne og teste Naive K-means og DCEC i deres kvantitative egenskaber til at gruppere data. MNIST består af håndskrevne tal og Tissues af 4 vævstyper: fedt, lymfocytter, stroma og tumor. Ved evaluaring af external metrics for begge datasæt viser DCEC sig som en bedre metode til clustering sammenlignet med den simplere K-means.

Eksperimenter der varierer antallet af clusters K og clustering vægten γ viser resultater, der antyder at DCEC grupperer Tissues betinget af vævsfarvning (staining). Som resultat bliver Tumor datasættet (n=98 patienter) stain-normaliseret ved Macenko normalisering (26).
Et subsæt til Tumor datasættet med hyperparametre K=4 og $\gamma = 0.6$ anvendes til cluster-analyse. Clustering resultatet korreleres til PAM50 genetik, Lehmann histologiske undertyper samt tumor og neoplasme stadieinddeling. Resultatet viser ingen tydelige mønstre mellem disse subtyper og clusters fra Tumor datasættet.

Clustering resultatet bruges også som kovariat input til at modellere overlevelsesanalyse på et patient-niveau. Kovariaterne er modelleret på to måder: kontinuert og binært. Resultaterne fra den kontinuerte modellering viser ingen signifikante sammenhænge mellem clusters og overlevelse. Resultaterne fra den binære modellering viser en signifikant bedre overlevelse for cluster 0 positive patienter. Overlevelseskurver for cluster 0 positive patienter (n=92) mod cluster 0 negative patienter (n=6) viser en signifikant dårligere overlevelse for cluster 0 negative patienter.

Sampling af de mest sandsynlige tiles fra hver cluster viser en tendens mod en residual stain variation, som indikerer en sammenhæng mellem clusters og stain normaliseringen. Resultatet viser et signifikant bedre udfald for cluster 0 positive patienter, men resultatet skal analyseres med forsigtighed grundet den potentielle sammenhæng mellem clusters og stain-residualet. Fremtidigt arbejde skal forbedre databehandlingen såvel som implementeringen af en clustering metode, der er mere robust mod stain-variationer, så der kan grupperes på baggrund af histologiske ligheder.

Contents

List of Figures

List of Tables

1

Introduction

Breast cancer (BC) is the most common cancer to women. In 2016 BC had a yearly incidence of 1.7 million, claiming 535.000 deaths, and causing 14.9 million disability-adjusted life years (DALY) (12). Breast cancer is a heterogeneous disease because it can arise from different cells of origin and can present with different clinical phenotypes (32), (19). Triple negative breast cancer (TNBC) is a heterogeneous BC subtype that has been associated with higher tumor grade and metastasis (25).

The definite cancer diagnosis is made by a pathologist analyzing tumor biopsies. This process is time consuming, expensive and takes years of practise for a pathologist. With the advancement of increased computational power, computational pathology is an increasingly popular approach to uncover disease patterns using deep learning. This thesis seeks to use computational tools on TNBC whole slide images (WSI) to provide algorithms that can guide pathologist towards more appropriate and efficient conclusions, diagnoses, and therapeutic interventions for patients.

1.1 Research statement

The primary goal of this thesis is to investigate TNBC WSIs with the to identify morphological patterns in a data-driven approach. The goal is to implement an unsupervised framework for spatial clustering of tumor tiles from TNBC patients and relate the clustering to already existing methods for tumor subtyping. The thesis is carried out in parts by solving smaller objectives:

- Define TNBC from a clinical point of view and make a literature review of current work of breast cancer analysis using machine learning.
- Implement unsupervised clustering algorithms that cluster TNBC tiles by morphology.
- Evaluate clustering algorithms in their quantitative and qualitative capabilities to separate morphology.
- Relate clustering results to existing BC subtypes:
    1. PAM50 genetic subtypes

2. 4 Lehmann TNBC histological subtypes
3. Tumor size and overall neoplasm disease stages

- Use survival modelling to predict patient outcomes.

1.2 Thesis outline

Chapters are structured as follows:
Chapter 2 is a theory section presenting relevant clustering techniques and a literature review about current work carried out to identify BC subtypes.

Chapter 3 presents the three datasets used in this thesis.

Chapter 4 describes the used methods in this thesis including data processing, unsupervised learning methods and cluster analysis.

Chapter 5 presents the results obtained from the experiments and methods described in chapter 4.

Chapter 6 evaluates and discusses the used methods and obtained results.

Chapter 7 summarizes the key points from previous chapters and concludes the thesis.

1.3 Breast cancer from a clinical perspective

Patients benefit from earlier detection of breast cancer as it leads to better survival outcomes. At early stages, a tumor is less likely to have metastasized and is therefore easier to treat curatively. Biomarkers are biological markers that can be used as a measurable indicator of a person's physiological state or condition. Biomarkers may be useful as prognostic or predictive indicators in clinical disease, and may also suggest possible targets for novel therapies. Several studies have been carried out identifying biomarkers for BC and for subtypes of TNBC specifically (41).

For individuals diagnosed with cancer, further analyses of biomarkers, such as the presence of the human epidermal growth factor 2 (HER2) receptor or detection of the genetic BRCA-I and BRCA-II mutations, provides further insight into choice of treatment and clinical prognosis. A tissue biopsy is in the diagnostic procedure that can definitely determine if the suspected tissue sample is cancerous.

1.4 Triple negative breast cancer

TNBC is clinically defined as breast cancer that lacks expression of estrogen and progesterone receptors and HER2 overexpression. TNBC is common BC subtype where a study by Blows *et al.* (5) gathered data from 12 studies (n=10159) and identified 1645 (16%) TNBC patients. TNBC is associated with a large tumor size

and higher grade of dysplasia, which describes a high mitotic rate (25). Because TNBC lacks growth factor hormone receptors they are difficult to target and treat with anti-hormonal therapies. This partly explains the why TNBC patients have a lower 5-year survival rate compared to non-TNBC. In a 2018 study by Goncalves *et al.* (13) they investigated 447 BCs with 19.5% TNBCs and found the overall 5-year survival to be 62.1% and 80.8% for TNBC and non-TNBC respectively. They conclude that TNBCs exhibit more aggressive behaviour, more frequent recurrence and worse survival outcome compared to non-TNBC.

TNBC subtypes

Several studies have investigated the differences in BCs and TNBCs from different data inputs like genetics or histology. Two commonly used types are here presented from genetics (PAM50) and histology (Lehmann). PAM50 has 5 subtypes genetically, and Lehmann has 4 histological subtypes. A part goal in this thesis is to investigate whether unsupervised learning can identify cluster patterns correlating to PAM50 and Lehmann subtypes.

1.4.1 PAM50

PAM50 uses 50 gene expressions to subtype BCs in 5 classes. Perou *et al.* (31) made a classification system that with hierarchical clustering characterized gene expression variations in a set of 65 surgical specimens of human breast tumours. The specimens came from 42 different individuals, collectively representing 8,102 human genes. They identified four groups of samples were related to different molecular features of mammary epithelial biology. The work from Perou *et al.* has led the way to the PAM50 genetic assay analysis (Bernard *et al.*, (4)). PAM50 has 5 subtypes Luminal A (LumA), Luminal B (LumB), HER2-enriched (HER2-E), Basal-like, and Normal-like. This classification approach can be applied across all clinical subgroups of breast cancer. PAM50 will be used in this thesis as a genetic parameter. The purpose of interest is here to correlate histology with genetics to investigate whether these are related.

1.4.2 Lehmann classification

Lehmann *et al.* subtype TNBC slides to 4 histological classes by investigating the heterogeneity of TNBC tumors. They suggest 4 tumor-specific subtypes: Basal-like 1 (BL1), basal-like 2 (BL2), mesenchymal (M) and luminal androgen receptor (LAR). The BL1 subtype is characterized by elevated cell cycle number and gene expression responsive to DNA damage, the BL2 subtype is enriched in growth factor signaling and myoepithelial markers. The M subtype has elevated expression of genes involved in epithelial-mesenchymal-transition and growth factor pathways. The LAR subtype is characterized by luminal gene expression and is driven by the androgen receptor. The group used five publicly available chemotherapy breast cancer gene expression datasets. They retrospectively evaluated chemotherapy response of over 300 TNBC patients from pre-treatment biopsies subtyped using either the intrinsic PAM50 or their own histological subtyping. Lehmann *et al.* found that TNBC subtypes demonstrated significant differences in response to

neoadjuvant (before main treatment) chemotherapy: 41% of BL1 patients achieved a pathological complete response (pCR) compared to 18% for BL2 and 29% for LAR with 95% confidence intervals (CIs; [33, 51], [9, 28], [17, 41], respectively). Lehmann group conclude that the 4 TNBC patient stratification gives valuable insight in neoadjuvant chemotherapy response. They show that BL1 will receive greater benefit from standard neoadjuvant chemotherapy than patients with other TNBC subtypes such as BL2 and LAR. They suggest subtyping of TNBC tumors because it could provide significant value for the future clinical decision making. Lehmann's work is interesting because it provides a dictionary-based histology subtyping and the work in this thesis attempts to relate clustering outcomes to Lehmann histology types.

2

Theory

Numerous machine learning approaches have been used in medical research to extract image features by reducing the dimensionality of an image. Feature extraction is useful when input data is large and a lot of the content in the image is considered redundant. Deep learning has been a hot topic to extract features from images gaining momentum since 2012 with AlexNet image classification (20) on the ImageNet dataset (16). Deep learning models have the ability to automatically extract non-linear feature representations of data, which has made deep learning a popular method to unsupervised feature extraction.

A pathologist uses histological biopsy slides to identify and diagnose cancer. Pathological analysis of the tumor biopsy determines the definitive basis of diagnosis and treatment of breast cancer (42). Using tumor biopsies to diagnose cancer is a time-consuming process: A technician prepares tissue slides by slicing a tissue biopsy into thin histologic sections, and staining the slides with various dyes. This thesis only focuses on hematoxylin & eosin (HE) stained slides. Digitizing the sliced and stained specimen is the basis for WSIs.

Digitized WSIs are large images, as large as billions of pixels taking a lot of digital storage. The vast size of a tumor from a microscopic point of view gives rise to high tumor intra-heterogeneity. That is a tumor slide can have different genomic and biological expressions within a tumor giving rise to diverse microenvironments within the same slide. Furthermore tumors show high inter-heterogenity because of different genetic expression profiles between patients. Due to the vast quantity of information WSIs contain, feature extraction methods are useful to obtain more compact feature representations.
The extracted features can be clustered to analyze the patterns in the WSI. Clustering is the analysis of grouping a set of objects in a way that the groups have similarities more to each other than to those in other clusters. This thesis aims to analyze TNBC samples for patterns in inter and intra-heterogenity.
Section 2.1 presents clustering methods to identify TNBC patterns.

2.1 Clustering methods

Various feature extractions methods exist to reduce dimensionality of an input. With inspiration from Min *et al.* (27), this section presents clustering methods

that use autoencoders (AEs) or convolutional AEs (CAEs) to extract features. This section also presents clustering methods like the simple K-means clustering algorithm and more sophisticated network achitectures that jointly optimize a CAE and clustering loss.

2.1.1 Autoencoder

AE is a commonly used method for learning data representations in an unsupervised manner. It consists of two parts: an encoder and decoder. For a dataset of N samples (images) $\{x\}_{i=1}^{N}$ consider the i^{th} sample x_i. The encoder $f_\theta(\cdot)$ maps the i^{th} sample x_i to a more compact latent representation embedding z_i.

$$z_i = f_\theta(x_i) \tag{2.1}$$

Secondly the objective for the decoder $g_\phi(\cdot)$ is to reconstruct the original input image using the compact representation z_i.

$$\tilde{x}_i = g_\phi(z_i) \tag{2.2}$$

The output $(\tilde{x}_i)$ is optimized to match the input (x_i) by optimizing a loss function. A common and popular choice the mean squared error (MSE) loss function. The MSE loss function L_r is defined as:

$$\min_{\phi,\theta} L_r = \min \frac{1}{N} \sum_{i=1}^{N} ||x_i - \tilde{x}_i||^2 \tag{2.3}$$

where θ and ϕ denote the encoder and decoder weights respectively. CAE uses a series of stacked convolutions to encode and decode a sample while using MSE as a loss function.

2.1.2 K-means clustering

K-means is a commonly used algorithm for clustering. The method assigns cluster labels to similar groups of unlabeled data. K-means works by calculating K centroids (center of gravity points) and assigning labels to all samples with the smallest Euclidean distance to a centroid. Given a dataset of $\mathbf{x} = (x_{(1)}, ..., x_{(n)})$ d-dimensional points the objective is assign labels from the K pre-defined clusters. For $i = \{1, 2, .., n\}$ datapoints and $j = \{1, .., K\}$ clusters the K-means clustering algorithm works by alternating between two steps:

1. Randomly initialize cluster centroids $\mu_1, \mu_2, .., \mu_K,\ \mu_j \in R^d$.

2. Assign labels and update centroids iteratively until convergence. For all samples i assign cluster labels with the smallest Euclidean distance to a cluster centroid:

   $$c_i = \arg\min_j ||x^i - \mu_j||^2 \tag{2.4}$$

   for all K clusters update the mean of new centroids:

   $$\mu_j = \frac{\sum_{i=1}^{n}(c_i = j)x_i}{\sum_{i=1}^{n}(c_i = j)} \tag{2.5}$$

Repeat eq. 2.4 and 2.5 until the assignments no longer change.

Notice that the method is not guaranteed to find the optimum as the result could depend on how the clusters are initialized.
A simple clustering algorithm is to use a combination of CAE and K-means clustering by extracting latent representations with a CAE and assigning cluster labels using K-means. Fig 2.1 shows the simple K-means-based clustering method that from now on will be denoted as Naive K-means (CAE together with K-means clustering).

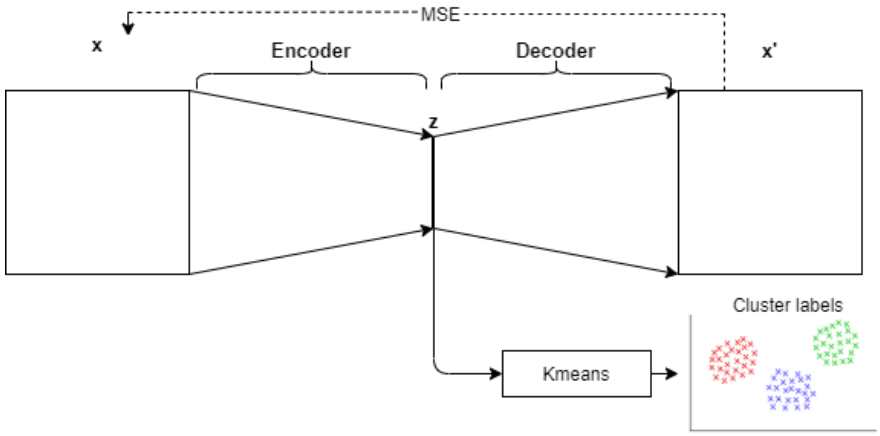


Figure 2.1: Simple CAE + K-means method. Pretrained CAE extracts image features, z, and K-means clusters the latent representation. The network uses the MSE loss function.

2.1.3 Deep clustering

Some deep learning networks combine AE feature learning and clustering into a unified framework directly clustering images in an end-to-end manner. This framework is called Deep Clustering.

AE based data clustering

Although a CAE can learn an effective latent representation and reconstruction, it might not cluster images with similar traits. One approach to cluster samples is to make a joint optimization between a clustering and reconstruction loss. In addition to the reconstruction loss, like in eq. 2.3, another objective optimizing a clustering loss is added. The idea a joint training between clustering and AE is what Song *et al.* (33) proposed. They define a new joint loss function:

$$\epsilon = \min_{\phi,\theta} \left(\frac{1}{N} \sum_{i=1}^{N} ||x_i - \tilde{x}_i||^2 - \gamma \sum_{i=1}^{N} ||z_i^t - c_i^*||^2 \right) \tag{2.6}$$

In eq. 2.6, the first term on the RHS represents the reconstruction loss, the second term is the clustering loss (γ is a clustering weight between 0 and 1), and N is the number of samples. The algorithm is set to a maximum number of iterations T. z_i^t is the latent representation at the t^{th} iteration and c_i^* is defined as:

$$c_i^* = \arg\min ||z_i^t - c_j^{t-1}||^2 \tag{2.7}$$

c_i^* is the closest cluster center for the i^{th} sample and c_j^{t-1} is the j^{th} cluster center computed at the $(t-1)^{th}$ iteration. The optimization is an iterative process where two components, the mapping function $f_\theta(\cdot)$ and the cluster centers c, are optimized in alternating steps. First, the mapping function $f_\theta(\cdot)$ is optimized while keeping the cluster centers c fixed. Secondly the the cluster centers are updated:

$$c_j^t = \frac{\sum_{x_i \in C_j^{t-1}} f^t(x_i)}{|C_j^{t-1}|} \tag{2.8}$$

Where C_j^{t-1} is the set of samples to the j^{th} cluster at $(t-1)^{th}$ iteration and $|C_j|$ is the number of samples in the cluster. The sample assignment computed in the last iteration is used to update the cluster centers of the current iteration. The sample assignments at the first iteration C^0 are initialized randomly.

Deep convolutional embedded clustering

Guo *et al.*. (14) suggest the Deep Convolutional Embedded Clustering (DCEC) method where CAE and a clustering metric are trained jointly. That is, a clustering oriented loss is directly built on embedded features to jointly perform feature refinement with the CAE and cluster assignment. Fig. 2.2 shows the DCEC network architecture.

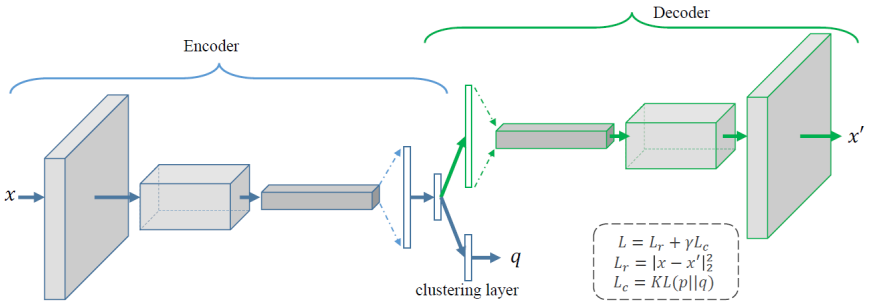


Figure 2.2: DCEC method. Soft labels q are assigned from the latent representation. The model is trained jointly for both L_r and L_c as seen in eq. 2.12. The figure is taken from (14) with the approval of the main author.

In contrast to Song *et al.* that trained the reconstruction and clustering losses in an alternating manner, this optimization is solved by mini-batch stochastic gradient descent and back-propagation at the same time. The process has cluster centers μ_j as trainable weights where each embedded point z_i is mapped to a soft label q_i by Student's t-distribution as defined by van der Maaten & Hinton (22):

$$q_{ij} = \frac{(1 + ||z_i - \mu_j||^2)^{-1}}{\sum_{j'}(1 + ||z_i - \mu_{j'}||^2)^{-1}} \tag{2.9}$$

where q_{ij} denotes soft label probabilities for all $i = \{1, .., N\}$ samples and $j = \{1, .., K\}$ clusters ($q_{ij} \in R^{N \times K}$). The numerator in eq. 2.9 maps z_i and μ_j by $(1+||z_i-\mu_j||^2)^{-1}$ that resembles the inverse square law making q_{ij} sensitive to z_i and μ_j with a small Euclidean distance and insensitive to larger Euclidean distances. The denominator $\sum_j'(1 + ||z_i - \mu_j||^2)^{-1}$ normalizes the i^{th} sample in q_{ij} relative to

sum of row elements in the i^{th} row. The i^{th} sample has a probability belonging to the j^{th} cluster. All cluster assignments are determined by the arg max (the cluster that sample i most likely belongs to). q_{ij} has a target distribution P:

$$p_{ij} = \frac{q_{ij}^2 / \sum_{i'} q_{i'j}}{\sum_{j'} (q_{ij'}^2 / \sum_{i'} q_{i'j'})} \tag{2.10}$$

p_{ij} is modelled accordingly to what Xie *et al.* (39) have done by normalizing with $\sum_{i'} q_{i'j}$ as cluster frequencies (summing the K columns). Having defined q_{ij} and q_{ij}, the DCEC loss function is defined by the clustering loss (L_c) with the Kullback-Leibler (KL) divergence and the reconstruction loss L_r:

$$\begin{aligned} L_c &= KL(P||Q) = \sum_i \sum_j p_{ij} log \frac{p_{ij}}{q_{ij}} \\ L_r &= \frac{1}{N} \sum_{i=1}^{N} ||x_i - \tilde{x}_i||^2 \end{aligned} \tag{2.11}$$

which defines the joint objective at which the algorithm is optimized towards:

$$L = L_r + \gamma L_c \tag{2.12}$$

2.2 Machine learning for TNBC and WSI, a literature review

The objective of this section is to give an overview to the challenges in WSI analysis and what research has been carried out by others to stratify TNBC patients. Researchers have worked with different machine learning methods and data types. The articles in this section serve to inspire aspects of this thesis and gives a general description to how machine learning can be used to analyze TNBC.

2.2.1 Whole slide imaging

The introduction of WSI scanners enables big-data analysis of histopathology as entire biopsies from patients can now be digitized. Together with increasingly powerful computational tools for machine learning algorithms, the idea of image analysis on WSIs has become feasible.
Although the computational tools have become more powerful, WSIs carry enormous amounts of data, with sizes up to $100,000 \times 100,000$ pixels in which each pixel has RGB channels.
Before machine learning can be used on the image data, it should be pre-processed. This includes removing irrelevant parts of the slide like background. The common procedure is subsequently to extract tiles that are smaller images from the WSI. Tile sizes range from $[32 \times 32]$ to $[10000 \times 10000]$ pixels (11) depending on the objective. The primary reason for extracting tiles instead of working on a slide-level is because of the vast memory requirements it would take for a computer to process a WSI directly. Tile extraction is a compromise made to analyze WSIs on a partial level. The individual tiles are used to aggregate information on a slide-level.

2.2.2 Multiomics

One way of analyzing TNBCs is through multiomics. Multiomics is an umbrella term where different -omic groups are combined to include more aspects than what one -omic group would have provided. Examples of typical -omics are genomics, transcriptomics, proteomics and metabolomics.

In a study by Chiu *et al.*. (9) genomic data was used on TNBC patients to characterize inter-tumor heterogeneity. The group clustered 137 TNBC patients from The Cancer Genome Atlas (TCGA) where they included data from gene, miRNA and copy number variation expressions. The group fused the three data types using the method similarity network fusion.
The study identified three patient clusters, where one of the clusters was enriched in the non-basal PAM50 subtype. This cluster exhibited more aggressive clinical features and had a distinct signature of oncogenic mutations. The study proposes this method as a new classification scheme for TNBC patient based on their -omics profile. This article describes an innovative way to cluster TNBCs and correlates the clustering outcomes to current well-established tumor subtypings, like Lehmann and PAM50. The 137 TNBC patients evaluated in this article are also the same subjects initially included in this thesis.

In a similar study, Zhang *et al.* (43) analyzed metastatic relapse on TNBC patients. Using multiomics data (transcriptome, copy number alterations, and mutations from 171 cancer-related genes), Zhang *et al.*. analyzed 453 primary TNBCs from three cohorts. With the objective of investigating metastatic relapse, they labeled each patient to either rapid (rrTNBc), late (lrTNBC), or no relapse groups (nrTNBC, no relapse/death with at least 5 years of follow-up).
Five different machine learning methods are used to predict the three outcomes. With 70% training and 30% test proportion, 10-fold cross validation was performed and the models were evaluated using receiver operator characteristic (ROC) curves averaged from the 10-fold cross validations. Results show PAM50 differences between lrTNBC and rrTNBC. The group differences primarily reflect differences in luminal features, with lrTNBCs being more likely to be non-basal (primarily LumA or B). 40% of the Lehmann LAR subtype tumors in the cohort had late relapse. In conclusion they propose the use of multiomics as a predictive model to identify patients as high risk for rapid relapse by identifying clinical and genomic features that can be used to rrTNBC.

The studies by Chiu *et al.* and Zhang *et al.* give insights to how others have worked with relating PAM50 and Lehmann subtypes to TNBC. With inspiration from this, a similar goal for this thesis is to correlate the clustering outcomes from extracted tumor tiles to PAM50 and Lehmann subtypes.

2.2.3 Machine learning on tumor WSI

This section presents selected articles that have used supervised and unsupervised learning on BC and TNBC slides.

Xie *et al.* (40) analyzed histopathological images of BC via supervised and unsupervised deep convolutional neural networks. The dataset consisted of malignant and benign tumors, both subtyped into four clinical subtypes.

InceptionV3 (35) and InceptionResNetV2 (34) deep learning architectures were adapted and used in binary and multi-class classifications. Transfer learning was used by including pretrained weights from ImageNet. Xie *et al.* adapted the network parameters by retraining and varying the last fully connected layer to either a binary (benign or malignant) or 8 class problem (including subclasses of benign or malignant).

By comparing results from the two different network architectures the study concluded that InceptionResNetV2 achieved the best results in the supervised classification. Achieving this, they turned focus towards unsupervised, data-driven learning. This was done to investigate and develop alternatives to the sometimes difficult, time-consuming, and expensive work required in supervised learning by assigning labels to samples.

The study extracted features of each image in a 1,536-dimensional feature vector and reduced it to a 2-dimensional vector by encoding layers. The 2-dimensional vector was used to represent the histopathological images in a low-dimensional feature space and K-means was used to cluster the images. To quantify the clustering results they used internal metrics (Silhouette Score, SSE) and external metrics (clustering accuracy (ACC), adjusted rand index (ARI) and adjusted mutual information (AMI)). Results from the clustering metrics suggest that the optimal number of clusters is 2. Xie *et al.*. concluded that the adapted network architecture is better at extracting deep-level features than the original network from InceptionResNetV2. They suggest future work in finding new ways that can improve the clustering accuracy.

Xie *et al.* work with both supervised and unsupervised learning on BCs. Although the patient cohort is not solely TNBCs, this study gives an insight to how they choose to quantify their clustering results with both internal and external metrics.

Prediction of residual cancer burden in triple negative breast cancer

In a study by Naylor *et al.*. (29), WSIs from TNBC were analyzed predict the residual cancer burden (RCB) score to measure projected treatment efciency based on a biopsy taken before the treatment. Using the RCB score, the group investigated how patients reacted differently to treatments given their histological profile. Data consisted of 122 histopathological slides at 40x magnification, of which 56 were annotated as pathological complete response (PCR) and 66 as RCB stage I-III. The study uses both manual and automatic feature extraction methods. For the automatic feature extraction, they down-sample the WSIs to tile sizes of 224x224 and encode each tile to a feature vector using a pre-trained 152-layer ResNet. They cluster the encoded feature vectors into k clusters, and represent the distribution

of tiles in each of the clusters for each slide. To minimize data size, they reduce the dimensionality of each tile with principal component analysis (PCA) to 50 features. Then they sub-sample the images by either randomly drawing a fixed number of feature vectors or using cluster-based down sampling. The cluster-based down sampling works by clustering all feature vectors from one patient into $n_i/40$ (n_i the number samples in a feature bag) clusters and then sampling feature vectors from each cluster.
After sampling the feature vectors, the authors cluster the encoding vectors into k clusters and represent the percentage of tiles in each cluster on a slide. To classify each patient they use random forest by setting k=4. Fig. 2.3 shows the framework predicting PCR or RCB using the feature extraction method from 152-ResNet:

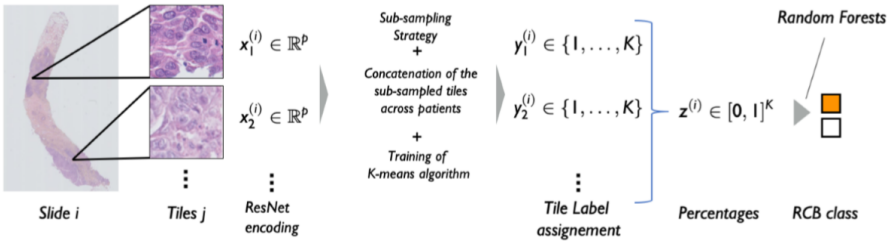


Figure 2.3: Binary prediction of pathological complete response (pCR) or residual cancer burden (RCB) from the 152-ResNet encoding. Figure is the same as seen in (29).

This study shows that encoding tiles using 152-ResNet is meaningful and that image augmentations can improve encoding performance.

2.2.4 Joint and individual analyses of TNBC using genominc and histology

A novel approach to stratifying BCs is by fusing histopathology and genomics. Examples of fusing genomics with histopathology are given by Carmichael *et al.*. (7) and Ash *et al.*. (3).
Carmichael *et al.*. seek to improve interpretability from convolutional neural network (CNN) while also including information from genetics. They do so by using the Angle-based joint and individual variation explained (AJIVE) method to investigate the relations between genetics and histopathology. The study analyzes n=1191 breast cancer patients with both genetics and WSIs, where the two data types are fused. AJIVE is a statistical feature extraction and dimensionality reduction algorithm for multi-block data. AJIVE returns three sets of components when applied to two data blocks (genetics and histopathology): joint block, block one (pathology), and block two (genetic).

The three components represent distributions from the AJIVE algorithm, reflecting stratified characteristics of the components together and separately. This gives the authors an insight to joint genetic and histological patterns. The distribution ends (positive and negative) were investigated to identify interpretable, distinct traits. For example in the first, joint component authors found two patterns: the negative scoring end was characterized by dense tumor infiltrating lymphocytes (TILs) and the positive end consisted of dense, high nuclear grade malignant cells reflecting

poorly-differentiated, high-grade tumors. When analyzing the PAM50 molecular subtypes for the joint compartment, they found that basal-like tumors tended to be more aggressive, with a high tumor grade and proliferation score.

Carmichael *et al.* introduce AJIVE as an exploratory window, making it possible for interpretable analysis on image and genetics simultaneously. However, their method requires future research to evaluate whether the learned features can be reproduced and validated by either pathologists or currently accepted automated computer vision systems. This article brings an interesting approach by fusing datasets to gain a deeper, more nuanced clinical insight, especially if the results are reproducible.

2.3 Survival analysis

This sections seeks to introduce the basic terms to survival analysis. With clustered patient tiles, it is of clinical relevance to relate morphological patterns to survival prognosis. This is done by including a response variable (how much time has passed since the beginning of the study until the end point). Patients participating in a study have two end points. Either a patient dies and the time from origin to death time is measured. The death of a patient is denoted an event from now. Alternatively a patient that has not died until follow-up has reached the end of the subject's participation. The patients still alive are typically called right censored events. Two methods are powerful when analyzing survival data: the survival function S(t) and the hazard function h(t). These functions will be used to study the impact the clustering outcomes have as factors to survival.

2.3.1 Survival and hazard function

The survival function S(t) is used to measure the fraction of patients living for a certain amount of time after an arbitrary start time. Assume a time T that denotes the survival time: T is the duration from start (here start is the diagnosis of cancer) until death. T is assumed to have a probability density function f(T) and a probability distribution function:

$$F(t) = P(T < t) = \int_0^t f(u)du \tag{2.13}$$

The survival function S(t) denotes the probability that a patient survives past T and is defined as:

$$S(t) = P(T \geq t) = 1 - F(t) \tag{2.14}$$

The hazard function h(t) is related to the survivor function and is used to express the the risk an event happens at some time t. It is defined as a probability of dying per unit time (by dividing by the time interval ΔT):

$$h(t) = \lim_{\Delta t \to 0} \frac{P(t \leq T < t + \Delta t | T \geq t)}{\Delta t} \tag{2.15}$$

where $h(t)\Delta T$ is the probability that a patient dies in the interval $(t, t + \Delta t)$. In this thesis t is a time in units of years. From probability theory the conditional probability for the numerator in eq. 2.15 can be rewritten:

$$P(t \leq T < t + \Delta t | T \geq t) = \frac{P(t \leq T < t + \Delta t)}{P(T \geq t)} \tag{2.16}$$

This expression can be rewritten in terms of F(.) from eq. 2.13:

$$\frac{F(t + \Delta t) - F(t)}{S(t)} \tag{2.17}$$

Inserting eq. 2.17 as the substitute for the numerator in eq. 2.15 gives:

$$h(t) = \lim_{\Delta t \to 0} \frac{F(t + \Delta t) - F(t)}{\Delta t} \frac{1}{S(t)} \tag{2.18}$$

where $\lim_{\Delta T \to 0} \frac{F(t+\Delta t)-F(t)}{\Delta t}$ is the derivative definition of F(t). The relation between h(t), S(t) and f(t) is therefore:

$$h(t) = \frac{f(t)}{S(t)} = -\frac{d}{dt} log[S(t)] \tag{2.19}$$

The cumulative hazard function H(t) is defined as the sum of the individual hazard rates h(t):

$$H(t) = \int_0^t h(u)du \tag{2.20}$$

in other words h(t) is the derivative to the cumulative hazard function. Given the relation between between h(t) and S(t), S(t) and H(t) are therefore also related:

$$S(t) = e^{-H(t)} \tag{2.21}$$

which follows:

$$H(t) = -log[S(t)] \tag{2.22}$$

The mathematical relation between hazard and survival functions forms the basis from which David Cox modelled the relation between the hazard rate and a set of covariates.

2.3.2 Cox proportional hazard regression

Cox modelled the hazard rate as a dependent to some covariates $\mathbf{x} \in R^{NxK}$. $\mathbf{x}$ is a N by K matrix, where each row corresponds to a patient with K covariates. For the i^{th} patient in $\mathbf{x}$, each column in this thesis is represented by a cluster covariate for the K clusters. The hazard function given the covariate data for the i^{th} patient is defined as:

$$h(t|x_i) = h_0(t)e^{(\beta_1 x_1 + ... + \beta_K x_K)} = h_0(t)e^{(\beta \mathbf{x})} \tag{2.23}$$

where $\mathbf{x} = [x_1, x_2, .., x_K]$ are the cluster covariates to the i^{th} patient, $\beta = [\beta_1, .., \beta_K]$ are the covariate coefficients that are estimated by regression and $h_0(t)$ is the baseline hazard rate when assuming all covariates $\mathbf{x}$ to be zero. The hazard $h(t)$ represents the relative risk of a patient dying at any given time t, given the patient has survived up to time t. The linear components $[\beta_1 x_1 + ... + \beta_p x_K]$ are also

called the prognostic index. The proportional hazard for patient can therefore be expressed given the prognostic index:

$$h_i(t) = h_0(t)e^{(\beta_1 x_1 + ... + \beta_K x_K)} \tag{2.24}$$

which can also be expressed from the prognostic index only:

$$\begin{aligned} \frac{h_i(t)}{h_0(t)} &= e^{(\beta_1 x_1 + ... + \beta_K x_K)} \\ log\left(\frac{h_i(t)}{h_0(t)}\right) &= \beta_1 x_1 + ... + \beta_K x_K \end{aligned} \tag{2.25}$$

The regression coefficients can be interpreted as the adjusted-for risk: how much does the relative risk change for one unit increase in x_i.
The right hand sides of eq. 2.25 have no terms depending on time. This reflects an important assumption Cox made to the cox proportional hazard: the relative risk is constant for all time values t, hence the name proportional hazards. The modelling uses survival data and some prognostic measurement x as a covariate. x can be both a continuous value (like a concentration) and discrete (like smoker yes/no).

For an intuitive understanding to cox modelling, consider the comparison between two participant groups (smoker and non-smoker) in terms of their expected hazards. Imagine a binary covariate $\mathbf{x}$ where $\mathbf{x} = 0$ denotes non-smokers and $\mathbf{x} = 1$ smokers. The hazard ratio (HR) between the groups (i=smoker, j=non-smoker) is:

$$HR = \frac{h_i(t)}{h_j(t)} = \frac{h_0(t)e^{(\beta_1 x_i)}}{h_0(t)e^{(\beta_1 x_j)}} = \frac{e^{(\beta_1 x_i)}}{e^{(\beta_1 x_j)}} = e^{(\beta_1(1-0))} = e^{\beta_1} \tag{2.26}$$

$e^{(\beta_1)}$ is the hazard ratio for the binary covariate to smoking yes/no. Note that HR and β are converted with the log-transform. HRs will be used to analyze relative risks patient groups have being associated with certain clusters. The risks for HR and β are analyzed as follows:

- HR$\approx$1 ($\beta \approx 0$): predictor does not effect survival
- $HR > 1$ ($\beta > 0$): increased risk
- $HR < 1$ ($\beta < 0$): reduced risk

The Cox Proportional Hazard is a semi-parametric model. Although the regression parameters β are known, the distribution of the outcome remains unknown and the baseline hazard function is not assumed to take any shape or form. The objective is to estimate the best coefficients β . An example is described by Breslow (6). The best coefficients are estimated by the optimizing the likelihood function for β. The estimates are based on the events taking place. The partial likelihood of patient i dying at time Y_i is defined:

$$L_i(\beta) = \frac{h(Y_i|\mathbf{x_i})}{\sum_{j:Y_j \geq Y_i} h(Y_i|\mathbf{x_j})} = \frac{h_0(Y_i)\theta_i}{\sum_{j:Y_j \geq Y_i} h_0(Y_i)\theta_j} = \frac{\theta_i}{\sum_{j:Y_j \geq Y_i} \theta_j} \tag{2.27}$$

where $\theta_j = e^{(\beta x_j)}$ and $\theta_i = e^{(\beta x_i)}$. The denominator in eq. 2.27 sums up all the patients j where an event has not occurred before time Y_i. Notice how the likelihood

is a function without the baseline hazard $h_0(Y_i)$. The partial likelihood is a score for which $0 < L_i(\beta) \leq 1$. Assuming that the subjects are independent and identically distributed the joint probability of all events (all events are indicated by $C_i = 1$) taking place can be multiplied:

$$L(\beta) = \prod_{i:C_i=1} L_i(\beta) \tag{2.28}$$

which with log transformation gives the log partial likelihood:

$$l(\beta) = \sum_{i:C_i=1} \left(\beta \mathbf{x_i} - log \sum_{j:Y_j \geq Y_i} \theta_j \right) \tag{2.29}$$

The best fit for β is estimated with the partial log likelihood in eq. 2.29. The optimization process is done by optimizing the derivatives $l'(\beta)$ and $l''(\beta)$ to β. The partial likelihood can be maximized using numeric algorithms like the Newton-Raphson algorithm which is used in this thesis by using the Lifelines library in Python.

2.3.3 Sample size

From a statistical point of view, the certainty to how correct the β estimates are depends on the number of events in data. The more events, the more reliable the estimated coefficients are. Particularly, the events per variable (EPV) defines the ratio of number of events per estimated variable. A general rule of thumb says 10 EPV is a good estimate stated by Peduzzi *et al.*. (30).

$$N_{subjects} = \frac{10/\#variables}{p_{event}} \tag{2.30}$$

here 2.30 estimates the patient size $N_{subjects}$ that are required given the fraction of events in the data is p_{event} and the number of variables that are to be estimated. Peduzzi's study focuses on a cardiac trial of 673 patients with 252 events (in which death is considered an event, $p_{event} = 252/673 = 0.37$) and 7 variables (EPV=252/7=36). Using simulations, they vary EPV = 2,5, 10, 15, 20 and 25 and assess the uncertainty of the coefficient estimates. They suggest caution about situations in which $EPV < 10$: *"The results should be cautiously interpreted in studies having fewer than 10 events per variable analyzed."*

Another study by Vittinghof *et al.* (38) agrees with Peduzzi *et al.*. that studies with $EPV < 10$ should be interpreted with caution. For Cox modelling, they find that instances with 5-9 EPV were not severely problematic and were usually comparable with 10-16 EPV. Authors in this study do not align with the systematic discounting of models with 5-9 EPV and conclude that the rule of thumb of 10 EPV is too conservative.

A $100(1-\alpha)$ confidence interval (CI) for HR can be found by exponentiating the CI limits for β. The CI for β is defined by $\hat{\beta} \pm z_{\alpha/2} s.e.(\hat{\beta})$. Here $\hat{\beta}$ is the estimated coefficient and $z_{\alpha/2}$ is the standard normal distribution, $\alpha = 0.05$ as standard. If the CI for β does not include zero (or CI does not include 1 for hazard

ratio), the coefficient is statistically significant. Consider the simple example described by David Collett ((10), p. 69-72) where he describes CIs by modelling one binary covariate X for women with breast cancer. x denotes either positive (x=1) or negative staining (x=0). The proportional hazard model for the i^{th} person at time t is about estimating one parameter β as follows:

$$h_i(t) = h_0(t)e^{\beta x_i} \tag{2.31}$$

where the baseline hazard $h_0(t)$ is for negatively stained tumors. From maximum likelihood estimation, $\hat{\beta} = 0.908, s.e.(\hat{\beta}) = 0.501$. The estimated hazard ratio for the positive group is therefore $e^{0.908} = 2.48$ (larger than $e^0 = 1$ for the stain-negative group). Therefore the women who have positively stained tumors will have a greater risk of death at any given time t than women with negatively stained tumors. Knowing the $s.e.(\hat{\beta}) = 0.501$ for the $\hat{\beta}$, the 95%CI for β is now:

$$CI(\beta) = [\hat{\beta} - 1.96 \cdot 0.501, \hat{\beta} + 1.96 \cdot 0.501] = [-0.074, 1.89] \tag{2.32}$$

Similarly the hazard ratio CI is found by exponentiating the results from eq. 2.32:

$$CI(HR) = [e^{-0.074}, e^{1.89}] = [0.93, 6.62] \tag{2.33}$$

From CI(HR) in eq. 2.33, there is not significant evidence supporting the idea that the negatively and positively stained groups have different survival trends because the CI for HR covers the possibilityy that $HR = 1$.

3

Data

Three datasets are used. One dataset is MNIST that consists of handwritten digits 0-9. The other two datasets are extracted from The Cancer Genome Atlas (TCGA) breast cancers (see (18)). From TCGA two independent datasets Tissues and Tumor are extracted. For the methods mentioned in section 2.1 (Naive K-means and DCEC), MNIST and Tissues will are exploratory datasets to get acquainted with the methods. Table 3.1 shows the overview of the three datasets.

Dataset	# images	Classes	Dimension
MNIST	70,000	10	784
Tissues	9,836	4	49,152
Tumor	426,642	NA	49,152

Table 3.1: Details of the three datasets. Note that no classes exist for Tumor.

3.1 Tissues and Tumor datasets

The datasets Tissues and Tumor are derived from TCGA. Of the 1098 breast cancers in TCGA, 98 are included in this study. For inclusion and exclusion criteria, see section 4.3.2. Tile sizes for both datasets $[3 \times 128 \times 128]$ at 20X magnification.

3.1.1 Tissues subset

The Tissues dataset consists of labeled tiles sampled from WSIs. It consists of small train and test partitions as seen in table 3.2.

Tissues	Fat	Lymphocytes	Stroma	Tumor
Train	1564	1535	1568	3180
Test	963	274	392	360

Table 3.2: The Tissues dataset for train and test partitions.

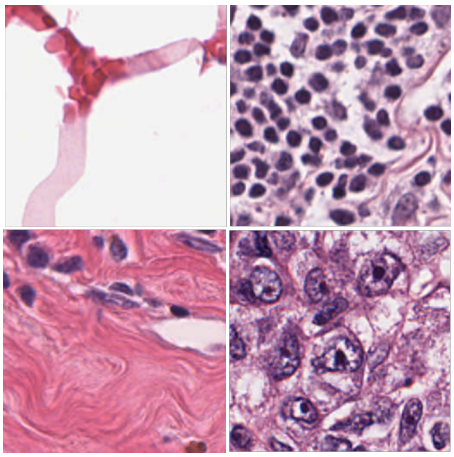


Figure 3.1: Representative images of tissue types: top left: fat, top right: lymphocytes, bottom left: stroma, bottom right: tumor.

Figure 3.1 shows the 4 tissue types for this dataset. Fat is characterized by its white and bright character and almost looks like the background in the WSIs. Lymphocytes typically appear as small, densely packed nuclei. The nuclei appear dark purple from the hematoxylin & eosine (HE) staining making for small and well-defined cells. Stroma is characterized by a strong pink color and consists primarily of connective tissue with a structural or connective function. The morphology of tumors varies more because of large intra- and inter-heterogenity. In general tumors are identified in WSIs as tissue that looks irregular and not like normally functioning tissue. Notice how the tumor has larger and less well-defined nuclei and the tissue looks irregular with less tissue structure.

3.1.2 Tumor subset

The Tumor dataset consists of 426,642 tiles extracted from predicted tumor regions from the 98 WSIs. The classification of tumor regions in the WSIs is described in section 4.3.2. Each patient has a genetic marker (PAM50), histological type (Lehman 4 class), tumor stage and neoplasm stage. Table 3.3 shows the subtypes on a patient level.

PAM50	Basal	Her2	LumA	LumB	Normal
	77	6	10	2	3
Lehman	BL1	BL2	LAR	M	
	34	13	19	32	
Tumor stage	I	II	III	IV	Other
	19	69	6	3	2
Neoplasm disease stage	I	II	III	IV	Other
	12	67	13	2	4

Table 3.3: Number of subjects for each subtype: PAM50 (genetic), Lehman (histology), tumor stage (tumor size score) and neoplasm disease stage (overall disease progression score).

Of the 98 patients, 19 patients died, with an average death time of 3.7 years, and 79 patients were still alive at 2.8 years follow-up time. The 79 surviving patients are also called right censored objects. Right censoring means a patient is still alive at some follow-up time, but it is unknown by how much lives on after that.

3.2 MNIST

The MNIST dataset (Lecun *et al.*. (23)) consists of 70,000 handwritten digits (0 to 9) of size $[28 \times 28 \times 1]$. 60000 have been used for training and 10000 have been used for testing.

4

Methods

4.1 Method overview

This section describes the methods and approaches described in the theory section (chapter 2). Figure 4.1 shows a simplified overview of the methods.

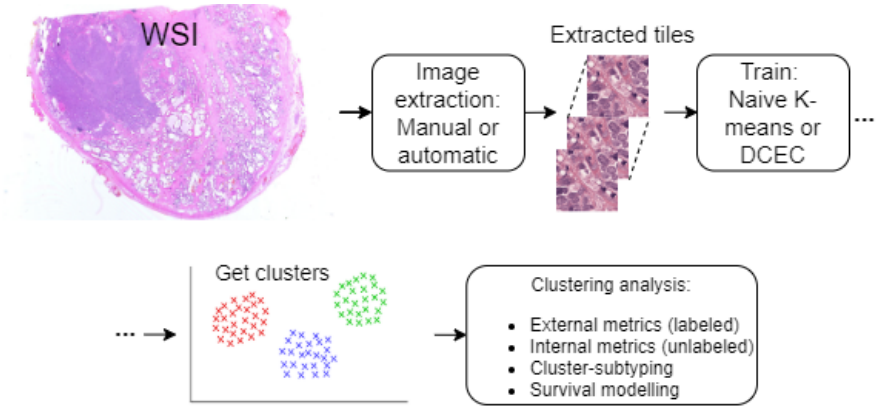


Figure 4.1: Method overview for WSI data analysis. WSIs are extracted from ROIs marked manually (Tissues dataset) or automatically (Tumor dataset). The tiles are clustered using either Naive K-means or DCEC. Cluster analysis is done on the assigned clusters.

4.2 Establishing a method comparison baseline

A baseline study to compare the clustering methods Naive K-means and DCEC is done to determine the best clustering method. The baseline evaluates the clustering methods by analyzing the two datasets MNIST and Tissues. The best method is used to cluster the automatically annotated Tumor dataset.

4.3 Data extraction

4.3.1 Tissues

Region of interest (ROI) are marked manually for each tissue type and extracted subsequently. The tissue types are extracted from a total of 8 patients with tile

size [3 × 128 × 128] at 20X. Figure 4.2 shows the manually annotated ROIs and tissue types.

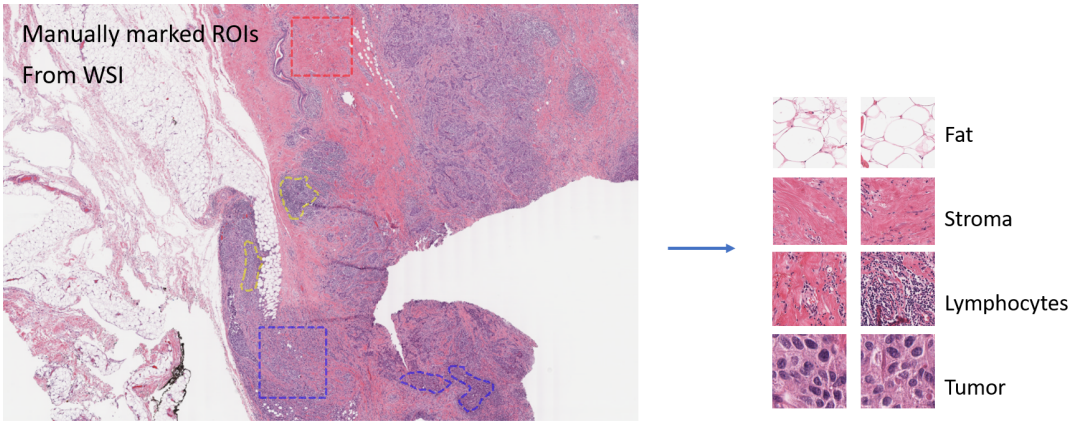


Figure 4.2: Manually marked ROIs to extract tissue types from. The blue dotted square in WSI shows how a tumor ROI is marked and tumor tiles are extracted.

4.3.2 Tumor

From TCGA 180 TNBCs were classified as TNBC by Lehmann *et al.*. (24). The metadata used in this thesis comes from Chiu *et al.*. (9). Chiu *et al.*. include 137 TNBC primary tumors with high tumor purity estimations.
In this thesis 30 WSIs that were not infiltrating ductal or medullary carcinomas were removed. This was done to have WSIs with a more uniform histology type. 9 WSIs were furthermore removed because the WSIs had too little tumor tissue for data extraction. A total of 98 HE stained WSIs are included. For the patient flow chart see figure 4.3.

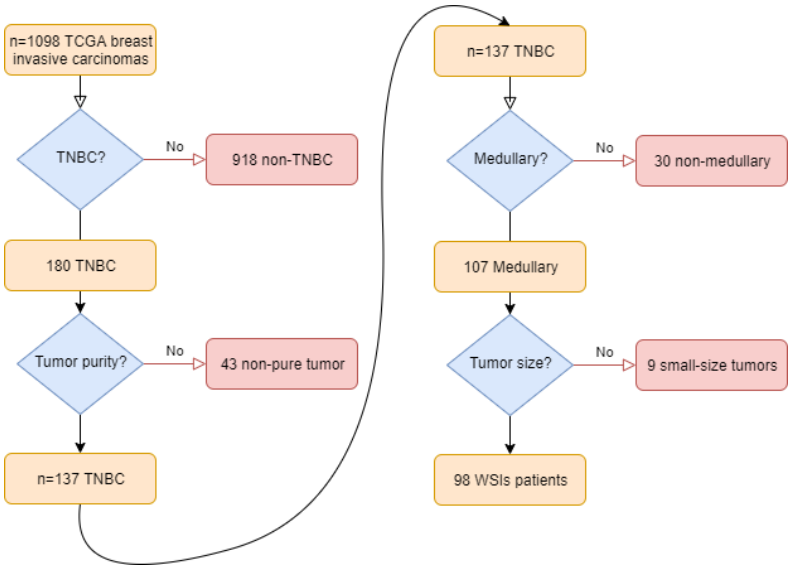


Figure 4.3: Patient flow chart of participants. Left half shows inclusion criteria for Lehman et al.. (n=180) and Chiu et al.. (n=137). Right half shows inclusion criteria in this study for histology consistency (n=107) and tumor size criteria (n=98).

The tumor ROIs were automatically annotated using Visiopharm's software. The software predicted ROIs in the WSIs to either tumor or background using the

DeepLabv3+ network (8). The network was pretrained on 8 manually annotated tumor/non-tumor WSIs. Figure 4.4 shows the square sub-regions of a WSI with manually annotated tumor/non-tumor regions.

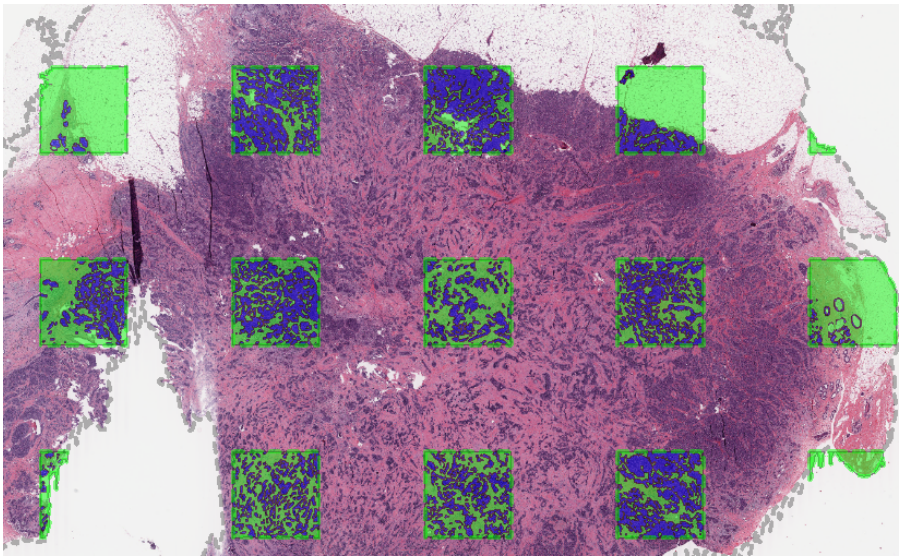


Figure 4.4: Manually annotated WSI with tumor (blue) and non-tumor (green). The blue-green squares are where the annotations were marked.

All 98 WSIs were predicted (tumor, non-tumor) using the pretrained tumor detection app. To account for potentially false tumor predictions, tumor ROIs with an area less than 10 tiles were set to background. This is similar to the work done by Muhammad *et al.*, although their tile size is $[224 \times 224]$ at 20X. WSIs in this thesis have a default 40X pixel:size resolution by:

$$40X :\sim 4\, pixels/\mu m \rightarrow 20X :\sim 2\, pixels/\mu m$$

with 2 $pixels/\mu m$ at 20X a tile width of 128 pixels corresponds to $\frac{128\, pixels}{2\, pixels/\mu m} = 64\mu m$. The area of a $[128 \times 128]$ tile is therefore:

$$Tile\ area = 64\mu m \cdot 64\mu m = 4096\mu m^2$$

and similarly the minimal tumor ROI threshold area is:

$$Tumor\ area = 10\, tiles \cdot 4096\mu m^2 = 40960\mu m^2$$

This approach requires a minimum tumor ROI area size of $A = 10 \cdot 4096\mu m^2 = 40960\mu m^2$ for not being set to background. After thresholding the ROIs by area, the edges of the tumor ROIs are reduced by the width of one tile. This processing step is also in line with Muhammad *et al.* and is done to minimize the extend of tiles with partial background (non-tumor). To give an example of the outcome of this method assume a circular tumor ROI with $A_1 = 40960\mu m^2$ and radius r_1:

$$r_1 = \sqrt{\frac{A_1}{\pi}} = \sqrt{\frac{40960\mu m^2}{3.14}} = 114\mu m$$

Edge erosion gives a new tumor ROI with radius r_2 and area A_2:

$$r_2 = 114\mu m - 64\mu m = 50\mu m$$
$$A_2 = \pi(r_2)^2 = 3.14 \cdot (50\mu m)^2 = 7850\mu m^2$$

A small tumor cell has radius $r = 10\mu m$ (see (1)). Assume that a tumor cell covers a square with sides $20\mu m + 2\mu m = 22\mu m$. Here each cell has a diameter of $20\mu m$ and extracellular matrix of $2\mu m$. Each tumor cell therefore covers an area of $A_{cell} = 22\mu m \cdot 22\mu m = 484\mu m^2$. This gives a rough approximate of up to $4096\mu m^2/144\mu m^2 \sim 8$ tumor cells/tile and $7850\mu m^2/484\mu m^2 \sim 16$ tumor cells in the eroded cicular tumor ROI. This approximate estimation shows how locally oriented the tiles are by covering up to 8 tightly packed tumor cells/tile. The tumor ROIs are extracted like in section 4.3.1 to tile size $[3 \times 128 \times 128]$ at 20X. This tumor selection method is the reason why 9 patient have been removed from this thesis (see figure 4.3).

Stain normalization

The tumor tiles are stain normalized like Macenko *et al.* (26) to account for over- and underexposed HE stains between slides as well as scanner-to-scanner variations. Macenko normalization works by scaling each image to its respective stain outliers through image transformations.
Each tile I with RGB channels is log-transformed relative to $I_0 = 240$. The optical density (OD) matrix is the log transformation from the normalized RGB image $I_{norm} = I/240$:

$$OD = -log_{10}(I/240) \tag{4.1}$$

OD is a RGB intensity matrix. $OD < 0.15 = \beta_{thres}$ is removed, to remove pixels that have almost no stain (low OD). The study notices that the RGB intensity relation between hematoxylin and eosin is geodesic (curved) in the RGB space, but linear after OD-transformation. They project the OD transformed pixels onto the geodesic to determine the stain vectors that explain the curvature. Stain vectors are in general supposed to give a normalized representation of the color of each stain for an image. The stain vectors for this purpose are hematoxylin and eosin.

Macenko et al. calculate the plane that the stain vectors cover by finding the two largest singular values from singular value decomposition (SVD) of the OD transformed pixels. The OD transformed pixels are projected onto the plane that the stain vectors form. The angles between the first SVD direction and all other points (pixels) is calculated. The angles with respect to the SVD direction give a distribution of angles and the 1^{st} and 99^{th} percentiles respectively are used as extremes to back-transform the images. Figure 4.5 shows the images before and after stain normalization. This type of normalization uses no target image as a reference to normalize towards since each image is normalized according to its own quantiles. The parameters $I_{norm} = 240, \beta_{thres} = 0.15$ are the same used by Macenko *et al.*.

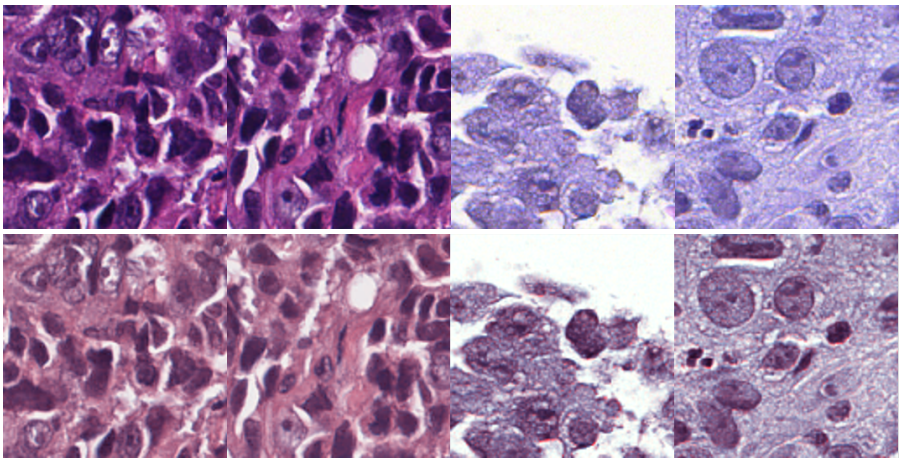


Figure 4.5: Macenko stain normalization. Selected tiles before (top row) and after (bottom) stain normalization. Notice how over-exposed tiles (two tiles top left) are reduced in intensity and how under-exposed tiles (two tiles top right) are corrected for their faint staining.

4.4 Metrics for model evaluation

Different metrics have been used to quantify the model performances. If ground truth labels are available, class predictions and ground truth can be evaluated against each other. This is called external metrics. If ground truth labels are not available, the model itself can be evaluated by how it clusters data. This is called internal metrics. The included external metrics are: normalized mutual information (NMI), adjusted rand index (ARI) and accuracy (ACC). The included internal metrics are: Calinski-Harabasz Index (CHI), Silhouette coefficient (SSE) and Davies–Bouldin Index (DBI). Several internal and external metrics are used to get a more detailed insight to how the models cluster tiles. Table 4.1 gives an overview of the metrics and how to analyze them.

Metrics			
External	NMI 0: no mutual information, 1: perfect correlation	ARI 0: random labeling, 1: perfect labeling	ACC Unsupervised clustering accuracy
Internal	CHI Higher CHI: more distinct clusters	SSE between -1 and 1. Closer to 1: better internal	DBI Lower DBI: more distinct clusters

Table 4.1: Overview of clustering metrics for labeled data (external) and unlabeled data (internal).

4.4.1 External metrics

External metrics quantify how well the assigned cluster labels correspond to the ground truth labels. External metrics have been used to analyse the datasets

MNIST and Tissues. More than one metric is used because the metrics reflect different aspects of the cluster distributions. For instance NMI quantifies the mutual information between the cluster and ground truth labels while ACC solely reflects the clustering accuracy.

NMI

NMI measures the agreement between assigned and ground truth labels. Let Y be the ground truth label vector and C cluster label vector. NMI is defined as:

$$NMI = \frac{2 \cdot I(Y, C)}{H(Y) + H(C)} \tag{4.2}$$

where *I(Y, C)* is the mutual information (MI) between the class and cluster labels and *H(.)* is the entropy for Y and C respectively. From information theory, MI is a measure of the mutual dependence between two variables (Y and C) and is related by the conditional entropy:

$$MI = I(Y, C) = H(Y) - H(Y|C) \tag{4.3}$$

where $H(Y)$ is the entropy for the class labels Y and $H(Y|C)$ is the conditional entropy for Y given C. MI reflects how the entropy reduces if the cluster labels are known. NMI is a score between 0 and 1, where NMI=0 means no MI and NMI=1 means perfect correlation between ground truth labels and assigned labels.

ARI

The Rand index (RI) is a similarity measure between assigned and ground truth labels. RI is a percentage measure of correct decisions and is calculated:

$$RI = \frac{TP + TN}{TP + FP + FN + TN} \tag{4.4}$$

where TP is the number of true positives, TN is true negatives, FP is false positives, and FN is false negatives. ARI is a variant to RI adjusting for the possibility of grouping by chance. $ARI \sim 0$ indicates random label assignments and $ARI = 1$ denotes perfect 1:1 label-prediction score. For an example to how ARI is calculated, see appendix A.4.

ACC

Unsupervised clustering accuracy is a method to identify the best one-to-one mapping between cluster and ground truth labels. It is similar to classification accuracy, but differs as this metric uses a mapping function m to identify the best mapping between the labels. This mapping is necessary because the unsupervised algorithm may use a different label than the actual ground truth to represent a cluster. From a mathematical point ACC is defined as:

$$ACC = \max_{m \in M} \frac{\sum_{i=1}^{n} 1\{l_i = m(c_i)\}}{n} \tag{4.5}$$

where l_i is the ground truth label, c_i is the cluster label, n is the number of samples, M is a mapping function with all the possible mappings between clusters and labels and $1\{\}$ is the indicator function returning 1 if $l_i = m(c_i)$. The best mapping is found using the Hungarian Algorithm (21).

4.4.2 Internal metrics

With no ground truth labels, internal metrics quantify how the clusters are separated from each other and how densely the clusters within the same groups are gathered. Internal metrics have been used to quantify the cluster dispersions for the Tumor dataset. Although the metrics might be based on similar principles (quantifying the within and inter-cluster separations), the ranges are different. For instance higher CHI indicates a better internal metric and CHI is maximized in a non-normalized manner. On the contrary DBI is a normalized metric, where a lower DBI indicates a better internal score. A review study comparing metrics by Arbelaitz et al. (2) shows that CHI, SSE and DBI are some of the best internal cluster validation tools.

CHI

Calinski-Harabasz Index (CHI) is expressed as a ratio of between-cluster variance and the overall within-cluster variance

$$CHI = \frac{Tr(B_k)/(k-1)}{Tr(W_k)/(N-k)} \tag{4.6}$$

where k denotes the number of clusters and N is the total number of subjects in the data. $Tr(B_k)$ is the trace of the matrix describing the between-groups dispersion and $Tr(W_k)$ is the trace of the within-cluster dispersion matrix:

$$B_k = \sum_{q=1}^{k} n_q (c_q - c_E)(c_q - c_E)^T \tag{4.7}$$

$$W_k = \sum_{q=1}^{k} \sum_{x \in C_q} (x - c_q)(x - c_q)^T \tag{4.8}$$

where C_q is the samples in cluster q, c_q is the centroid for cluster q, c_E is the center point of all the data from the N samples, and n_q is the number of samples in cluster q. Well-separated clustering solutions yield high CHI values.

SSE

The silhouette coefficient (SSE) is a normalized score between -1 for poor clustering and 1 for the best score. Let $a(i)$ denote the mean distance between sample i ($i \in C_i$, sample i is in cluster C_i) and all other data points within the same cluster:

$$a(i) = \frac{1}{|C_i| - 1} \sum_{j \in C_i, i \neq j} d(i,j) \tag{4.9}$$

where $d(i,j)$ is the distance between two points normalized to the cluster length by $\frac{1}{|C_i|-1}$ which is the distance measure for cluster C_i. a(i) is small for tightly bound clusters as $d(i,j)$ will have small distances. The dissimilarity $b(i)$ is defined as the mean distance between a point i in cluster C_i and all points in another cluster where $C_k \neq C_i$:

$$b(i) = \min_{k \neq i} \frac{1}{|C_k|} \sum_{j \in C_k} d(i,j) \tag{4.10}$$

In other words $b(i)$ is the smallest mean distance from i to all points to the closest neighbour cluster that i is not in. The silhouette score for one point i is now the normalized difference between $a(i)$ and $b(i)$:

$$s(i) = \frac{b(i) - a(i)}{\max[a(i), b(i)]} \tag{4.11}$$

DBI

Davies-Bouldin Index (DBI) is the average similarity between each cluster C_i, $i = \{1, .., K\}$ and its most similar neighbour cluster C_j. The similarity is defined as:

$$R_{ij} = \frac{s_i + s_j}{d_{ij}} \tag{4.12}$$

where s_i is the average distance between each point in cluster i and the centroid for C_i (similarly for s_j in cluster C_j) and d_{ij} is the distance between cluster centroids i, j. DBI is defined as the average similarities:

$$DBI = \frac{1}{K} \sum_{i=1}^{K} \max_{i \neq j} R_{ij} \tag{4.13}$$

An objectively better internal metric for DBI is given by a lower DBI.

4.5 DCEC training

To get a better understanding of the parameters in DCEC, two separate experiments are made by varying the number of clusters K, and the clustering weight γ. The experiments are carried out in two steps: One where only K is varied, and another where only γ is varied. Although the experiment parameters for MNIST and Tissues are different, the DCEC optimization algorithm is the same:

1. Pretrain CAE for x epochs.

2. Initialize K cluster centroids with K-means on the latent representation.

3. Assign γ value and update CAE weights, centroids, and distributions Q and P by iterating the steps:
   1) Update CAE weights and cluster centroids with optimizer from eq. 2.12
   2) Update soft labels Q and target distribution P every T^{th} batch.
   1) and 2) are repeated for up to x epochs (may vary between experiments) or until the prediction changes between updates vary by less than $\delta = 0.001$ (less than 0.1 % change between target updates).

Training is done using PyTorch. The number of pretraining epochs vary depending on the dataset and is trained until convergence. The models are trained using Adam optimization both for the pretraining and joint optimization. Weight decay is set to 0.1 for both pretraining and joint optimization every 20^{th} and 200^{th} epoch. The baseline experiment comparing DCEC to Naive K-means has $\delta = 0$.

4.5.1 MNIST

The network architecture for MNIST is the same as used by (14). Figure 4.6 shows the network architecture used for MNIST.

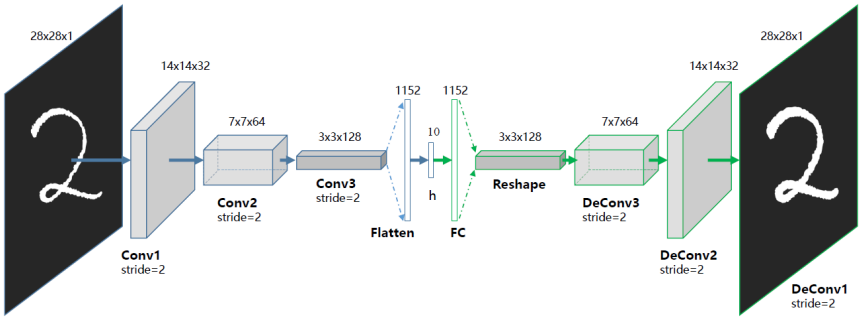


Figure 4.6: Network architecture for MNIST. An input image of size $[28, 28, 1]$ *is compressed to a 10-dimensional latent represenation between the fully connected, flattened layers on the encoder and decoder side* ($1152 = 3 \cdot 3 \cdot 128$)*. Figure is from DCEC article (14) with the author's permission.*

The encoder in figure 4.6 consists of three convolutional layers all with stride 2 and with the notation $Conv_{filter}^{kernel\ size}$: $Conv1_{32}^{5} \rightarrow Conv2_{64}^{5} \rightarrow Conv3_{128}^{3} \rightarrow FC_{10}$. The decoder is a mirror of the encoder. All layers except the input, output and embedding layers are activated by the standard ReLU to obtain non-linear mapping functions. With an input dimension of $28 \cdot 28 \cdot 1 = 784$ and a latent representation of $z \in R^{10}$ the compression ratio (CR) is $CR = 10/784 = 0.013$, or 1.3 %.

4.5.2 Tissues and Tumor

The overall network architecture for the Tissues and Tumor datasets is the same as in figure 4.6 but the image dimenstions and fully connected layer differ. The tiles were augmented by random horizontal flips to make the CAE more robust against spatial orientations in the tiles. Batch normalization in this network did not improve the model performance and was therefore dropped from the architecture. The linear layers on each side of the latent representation are significantly larger. Note how the CR however is similar to that of MNIST. For the Tissues and Tumor dataset, $CR = 512/(3 \cdot 128 \cdot 128) = 512/49152 = 0.010$, or 1%. Figure 4.7 shows the network architecture for the Tissues and Tumor datasets.

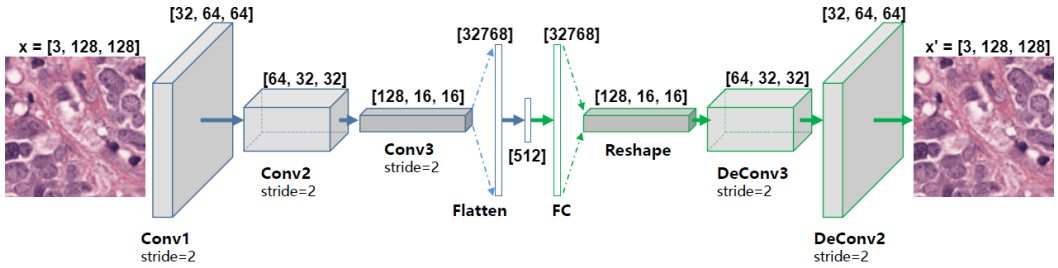


Figure 4.7: DCEC architecture for Tissues and Tumor datasets.

Sampling the Tumor dataset

The Tumor dataset consists of n=426,642 tiles (table 3.1). The number of extracted tiles from each WSI varies a lot. The lowest and highest number of tiles extracted from a WSI is 247 and 30125 respectively. All tiles are used to pretrain the CAE. The Tumor dataset converged after fewer epochs and took longer because of the larger dataset. To save on computational time and to cluster comparable number of tiles, up to 400 tiles from each WSI are extracted when training the joint optimization. Note this is only for the Tumor dataset. Four WSIs with less than 400 tiles (247, 276, 301 and 332 tiles) were included. As a result clustering is done on $94 \cdot 400 + 247 + 276 + 301 + 332 = 38756$ tumor tiles. Sampling a subset from a library is similar to Muhammad *et al.* (28). Here they sample a subset of 100,000 tumor tiles from their tumor library.

4.6 DCEC experiment setup

Two types of experiments are carried out: One where the numbers of clusters K are varied and another where the γ weights are varied. The idea is that gaining insights to the model characteristics will provide a better foundation for modelling the Tumor dataset appropriately.

The hyperparameters in section 4.6.2 and 4.6.3 are designed to be as consistent as possible between the experiments and datasets. Note how $T = 100$ for all experiments (the distributions q_{ij} and p_{ij} are updated every 100 batches). MNIST has $60,000/256 = 235$ batches and Tissues $7847/32 = 245$ batches. This approach is consistent between the datasets because q_{ij} and p_{ij} is updated two times during training of an epoch. Experiments in appendix A.3, table A.8 show that the value T is mostly a matter of computational time, lower T gives higher training time. Empirically the LR for the joint optimization was the most sensitive hyperparameter, for Tissues and Tumor dataset. Empirically LR=0.0005 is the best LR, as LR=0.001 makes the training for Tissues and Tumor unstable and LR=0.0001 is too low for the network to converge.

4.6.1 Dimensionality reduction

The latent representations are reduced to 2D so they can be visualized. For the datasets that means dimensionality reduction by:

$$MNIST : R^{10} \rightarrow R^2$$
$$Tissues/Tumor : R^{512} \rightarrow R^2$$

The two methods used for dimensionality reductions are: principal component analysis (PCA) and T-distributed Stochastic Neighbor Embedding (tsne). PCA uses orthogonal transformation to transform a set of correlated variables into a set of linearly uncorrelated variables, principal components. Reducing to two dimensions keeps the two principal components that explain the most variance in two directions.

Tsne is a computational expensive non-linear dimensionality reduction method that models the distribution of the high-dimensional features and optimizes the distributions to minimize KL divergence. Note that there is no motivation in comparing the two dimensionality reductions as they are only used for visualization tools.

4.6.2 Vary number of clusters K

MNIST

For this experiment all is kept constant, except for the number of clusters $K = \{5, 8, 10, 12, 16\}$. Fixed parameters for experiment:

Batch size	LR clust	LR pretrain	pretrain epochs	z dim	T	γ	δ
256	0.0005	0.001	100	10	100	0.1	0.001

Tissues

The values of K is varied increasing with $K = \{4, 8, 12, 20\}$ clusters to investigate if tissue subtypes appear with an increasing K. Fixed parameters for experiment:

Batch size	LR clust	LR pretrain	pretrain epochs	z dim	T	γ	δ
32	0.0005	0.001	100	512	100	0.1	0.001

4.6.3 Vary clustering weight γ

MNIST

For this experiment γ is varied and all else is kept constant. The experiment include $\gamma = \{0.1, 0.4, 0.6, 1\}$. The experiments are trained with the same pretrained network and all experiments are trained in the joint optimization for 50 epochs with

$\delta = 0$. $\delta = 0$ is chosen to investigate how the model converges once $\delta < 0.001$. Fixed parameters for experiment:

Batch size	LR clust	clust epochs	LR pre	pre epochs	z dim	T	K	δ
256	0.0005	50	0.001	100	10	100	10	0

Tissues

For these experiments $\gamma = \{0.1, 0.4, 0.6, 1\}$. The experiments with Tissues were less stable compared to the experiments of varying γ for MNIST and therefore $\delta = 0.001$ was chosen as a more conservative convergence criterion. Fixed parameters for this experiment:

Batch size	LR clust	LR pretrain	pretrain epochs	z dim	T	K	δ
32	0.0005	0.001	100	512	100	4	0.001

An additional Tissues experiment with $\delta = 0, \gamma = 0.1$ was done as seen in figure 5.2 to the baseline method comparison.

4.7 Cluster analysis

4.7.1 Cluster distributions

The cluster distributions are correlated to PAM50, Lehman histology, tumor stage and an overall tumor disease stage scoring (neoplasm).
This is done to investigate whether there are histological or PAM50 genetics that are distinguishable within the clusters. The hypothesis for tumor stage and overall disease stage is that tumors at earlier stages (a less progressed tumor) will contribute to a better overall survival for that cluster as it will have lower probability of metastatic spread and death.
The subtypes are defined on a slide-level. If a histology slide is defined as Lehman type BL1, all tiles from that slide are categorized BL1. Tumor stage III and IV are merged and so is neoplasm disease stage III and IV.

4.7.2 Cox regression

Survival information from a slide-level is used to model the cox regression. Patient information includes survival outcome (dead, or alive at follow up) and the time of follow up. The cox regressions are modelled using the Lifelines package in Python. Figure 4.8 describes the steps in the survival analysis.

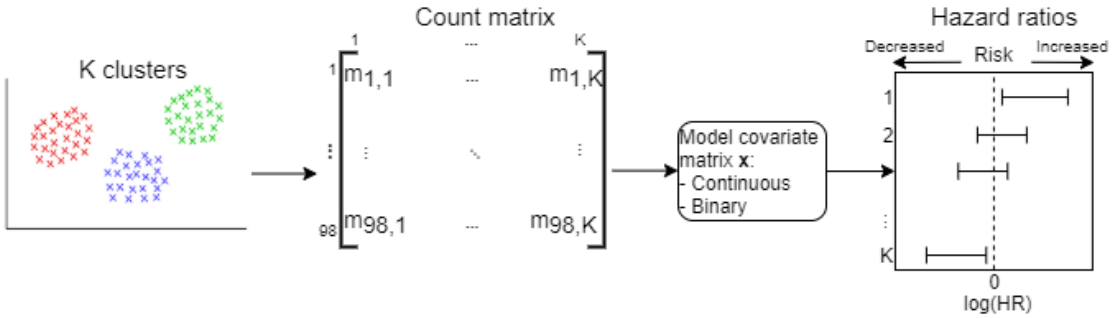


Figure 4.8: Survival modelling. From the cluster assignments, the count matrix $\mathbf{m}$ *is used to model the covariate* $\boldsymbol{x}$ *as either continuous or binary variables to estimate HRs for the clusters.*

The covariate $\mathbf{x}$ is modelled for all the patients from the count matrix $\mathbf{m}$. The count matrix denotes the number of tiles that are assigned to the K=4 clusters for all 98 patients, $\mathbf{m} \in R^{(98\times4)}$. $\mathbf{x}$ is used to model both continuous and binary covariates to investigate different approaches to Cox proportional hazard modelling. The continuous covariate is L1 normalized which divides each patient row by its summed cluster frequencies returning a covariate with values between 0 and 1. This gives equal weights across patients. Because of high collinearity between the continuous covariates a $penalizer = 0.1$ is added when fitting the cox regression. Fitting the regression in the Lifelines package that adds a penalizer scaling the sizes of the β coefficients during regression.

The binary covariate is modelled empirically so a cluster needs an activation of more than 4 tiles from a slide exists in a cluster before the cluster is considered positive (1 is assigned) and alternatively negative (0 is assigned). The idea for implementing an activation is to avoid a covariate matrix $\mathbf{x}$ with high collinearity like the continuous. Empirically a low activation like ≥ 1 tiles/cluster returns a covariate $\mathbf{x}$ where most clusters are positive to most patients. To illustrate the outcome of the continuous and binary modelling, consider the two patients in the two rows with 247 and 400 tiles respectively distributed on 4 clusters:

$$\begin{bmatrix} 0 & 15 & 230 & 2 \\ 5 & 23 & 370 & 2 \end{bmatrix} \xrightarrow{L1\ norm} \begin{bmatrix} 0 & 0.06 & 0.93 & 0.01 \\ 0.0125 & 0.0575 & 0.925 & 0.005 \end{bmatrix}$$

and for the binary modelling:

$$\begin{bmatrix} 0 & 15 & 230 & 2 \\ 5 & 23 & 370 & 2 \end{bmatrix} \xrightarrow{>4\ tiles\ activation} \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \end{bmatrix}$$

With statistical considerations described in section 2.3.3, K=4 clusters, is chosen as the only instance to model cox regression. A cluster is deemed significant if the CI does not cover 0 for the β coefficient (Notice how cluster 1 and K are deemed significant in figure 4.8 to the right). If a cluster is deemed significant, it will be compared against all the other clusters under the null hypothesis that the groups have similar survival functions S(t):

$$H_o : S_{cluster}(t) = S_{other}(t)$$
$$H_1 : S_{cluster}(t) \neq S_{other}(t)$$

The null hypothesis is tested using the log rank statistics, that follows a χ^2 distribution. The observed distribution χ^2_{obs} between two groups is defined as:

$$\chi^2_{obs} = \sum_{j=1}^{2} \frac{(\sum_{j=1}^{2} O_{j,t} - \sum_{j=1}^{2} E_{j,t})^2}{\sum_{j=1}^{2} E_{j,t}} \tag{4.14}$$

where $\sum_{j=1}^{2} O_{j,t}$ is the sum of observed events for group j and $\sum_{j=1}^{2} E_{j,t}$ is the expected number of events in group j over time. The subscript j denotes the group and t is included to emphasize that the observed and expected events are dependent on time although time itself is not included as a variable directly. Events take place during time, and the event rate might be higher or lower for different times of t. For j=1,2 groups, the expected number of events is:

$$E_{j=1,t} = N_{1,t} \frac{O_{tot}}{N_{tot}} \tag{4.15}$$
$$E_{j=2,t} = N_{2,t} \frac{O_{tot}}{N_{tot}} \tag{4.16}$$

where $N_{tot} = N_{1,t} + N_{2,t}$ is the total number of patients at risk and $O_{tot} = O_{1,t} + O_{2,t}$ is the number of observed events at each event time. From eq. 4.14 χ^2_{obs} is defined as the sum of the expected events from the two groups. The null hypothesis is compared to the critical value χ^2_{crit}, $\chi^2_{crit} = 3.84$ for $\alpha = 0.05$ and 1 degree of freedom. If $\chi^2_{obs} > \chi^2_{crit}$ H_0 is rejected. The log rank test statistic is calculated using the Lifelines library.

5

Results

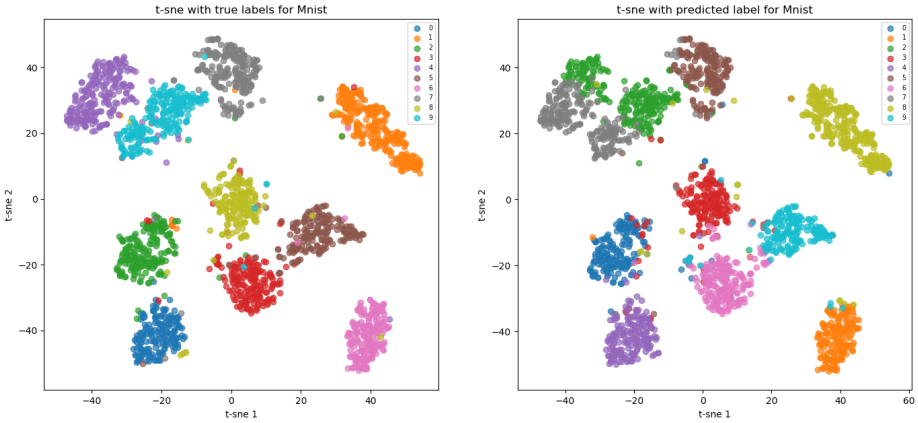
5.1 Baseline comparison

This section shows the results of the baseline experiments comparing Naive K-means and DCEC for the datasets MNIST and Tissues.

5.1.1 MNIST

Figure 5.1 shows latent representations for MNIST using tsne dimensionality reduction. Figure 5.1b shows the clustering from Naive K-means and figure 5.1b for DCEC, with $\delta = 0$. Both Naive K-means and DCEC have large overlap between the classes 4 and 9 (figure 5.1 (a, right) and (b, left)).

[a]



[b]

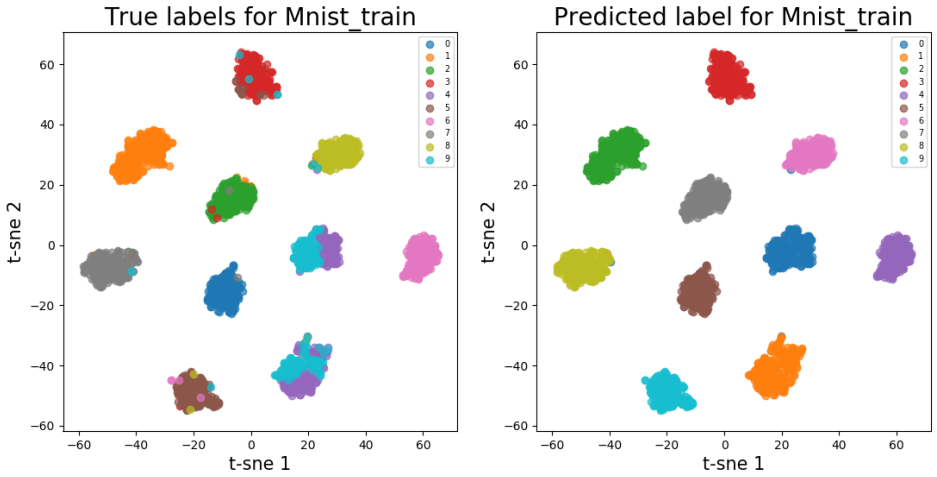


Figure 5.1: T-sne dimensionality reduction for n=2000 MNIST image samples, K=10. left plots are colored with ground truth labels and right with predicted labels. (a) Naive K-means. (b) DCEC $\gamma = 0.1$, $\delta = 0$.

Table 5.1 shows the external metrics for the two methods for both train and test partitions. DCEC shows higher external metrics for both train and test partitions. The models generalize well as the metrics are comparably high for train and test partitions.

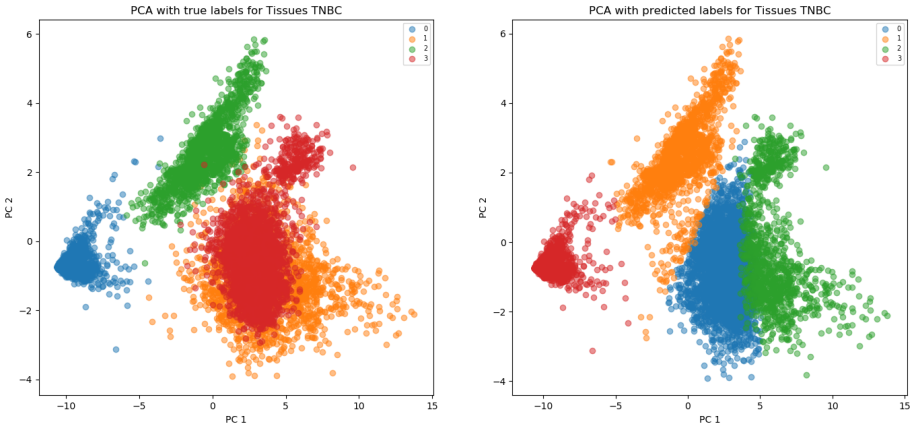
K=10 MNIST	NMI	ARI	ACC
Naive, train	0.77	0.73	0.84
Naive, test	0.79	0.75	0.86
DCEC, train	**0.87**	**0.84**	**0.88**
DCEC, test	**0.88**	**0.84**	**0.88**

Table 5.1: External metrics for MNIST.

5.1.2 Tissues

Figure 5.2a shows the PCA dimensionality reduction of the latent representation from Naive K-means and figure 5.2b for DCEC. Like figure 5.1b DCEC clusters groups tightly when $\delta = 0$. Note how fat and stroma tissue types are clustered to their own individual clusters and how lymphocytes and tumor tiles are overlapping.

[a]



[b]

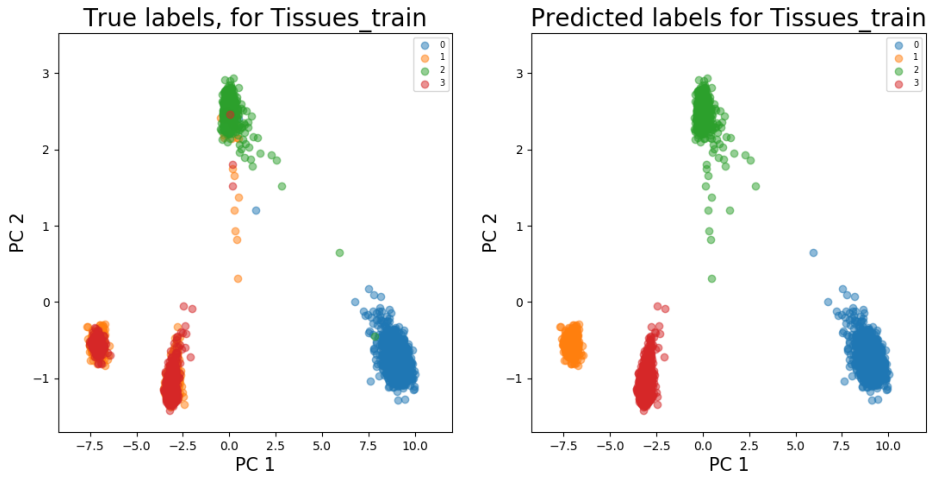


Figure 5.2: PCA dimensionality reduction for Tissues tiles, K=4. left plots are colored with ground truth labels and right with predicted labels. (a) Naive K-means. (b) DCEC $\gamma = 0.1$, $\delta = 0$.

Table 5.2 shows the external metrics for the two methods for both train and test partitions. In general DCEC shows higher metrics for both train and test partitions.

K=4, Tissues	NMI	ARI	ACC
Naive, train	0.70	0.63	0.82
Naive, test	0.78	0.86	0.83
DCEC, train	**0.76**	**0.70**	**0.83**
DCEC, test	**0.82**	**0.90**	**0.92**

Table 5.2: External metric scores for Tissues.

5.2 DCEC experiments

This sections shows DCEC results when varying K and γ.

5.2.1 Vary K

MNIST

This section shows the result for K=12 clusters for MNIST. For the other results (K=5, 8, 10, 16) see appendix A.2. Figure 5.3 shows 10 image samples from each cluster and table 5.3 the confusion matrix for same experiment. Note how class 1 is split between clusters 7 and 11 and how class 5 is split between clusters 4 and 9.

Figure 5.3: MNIST clusters for K=12. Each row is a cluster with 10 sampled images.

	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
0	0	14	25	18	4	0	0	6072	0	68
1	0	10	17	5972	0	33	0	1	18	114
2	1	7	3	0	5711	2	2	10	2	61
3	3	101	5889	51	1	6	3	38	6	1
4	2	0	0	5	0	2592	50	0	17	8
5	6	23	3	18	105	9	4	117	67	5657
6	5887	1	3	0	6	6	17	2	5	12
7	0	3372	0	0	1	0	3	9	9	3
8	14	1	1	1	9	11	5745	0	4	1
9	2	0	0	39	0	2752	79	0	22	11
10	7	11	11	27	5	9	13	6	5695	12
11	1	3202	6	0	0	1	2	10	6	1

Table 5.3: Confusion matrix MNIST, K=12. Columns are class labels, rows are cluster labels.

Tissues

This section shows the result for K=8 clusters for Tissues. For the other results (K=4, 12, 20) see appendix A.2. Figure 5.4 shows 10 image samples from each cluster and table 5.4 the confusion matrix for same experiment. Note how the stroma tiles (cluster rows 2 and 6) have been separated between two clusters and how the tiles in cluster row 5 consist of tumor tiles with a deep, pink stain.

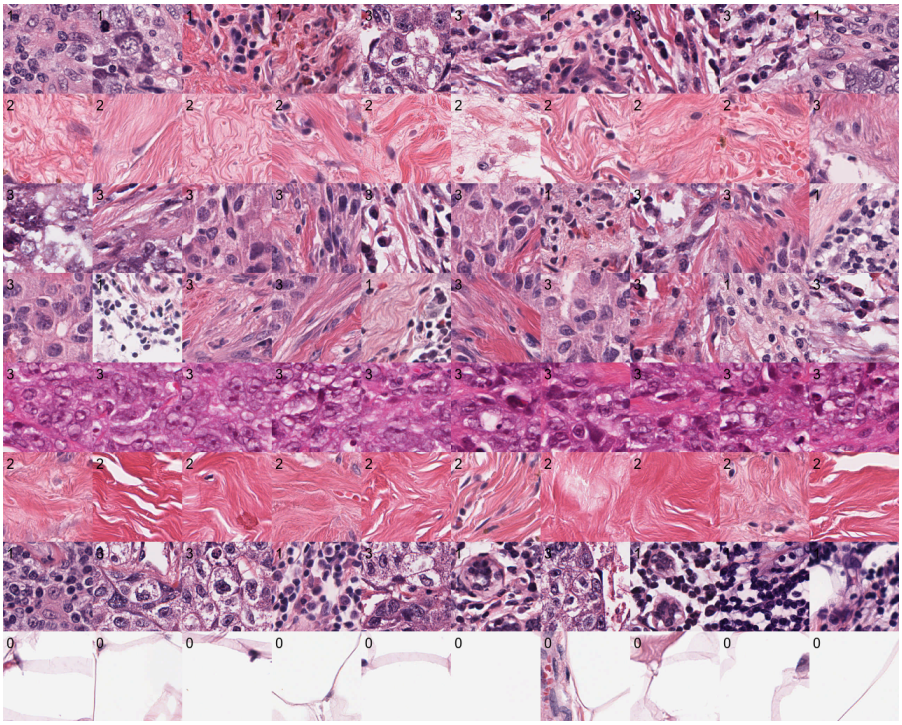


Figure 5.4: Tissues clusters for K=8. Each row represents a cluster with 10 tiles sampled from each cluster. The small numbers in the upper left corners of the tiles represent class labels: 0=fat, 1=lymphocyte, 2=stroma and 3=tumor.

	Fat	Lymphocytes	Stroma	Tumor
0	0	287	0	1158
1	2	9	1090	30
2	0	160	0	689
3	0	209	2	394
4	0	5	0	258
5	0	5	475	5
6	0	860	0	646
8	1562	0	1	0

Table 5.4: Confusion matrix Tissues. Columns are class labels, rows are cluster labels.

5.2.2 Vary gamma

This section show results for varying γ for MNIST and Tissues dataset. The results show that MNIST is more stabile than the Tissues dataset. This stability shows in figure 5.5 (MNIST) and 5.6 (Tissues), that show the external metrics as a function of the update interval T. MNIST in figure 5.5 shows a convergence towards a better solution with better metrics. Figure 5.6 does not converge towards better metrics. The results show how the datasets vary in stability and converge.

MNIST

Figure 5.5 shows an example of how external metrics improve as a function of target updates T and table 5.5 shows the external metrics varying γ. The metrics are comparable when varying γ for MNIST. The other plots varying γ are shown in appendix A.1, figure A.1.

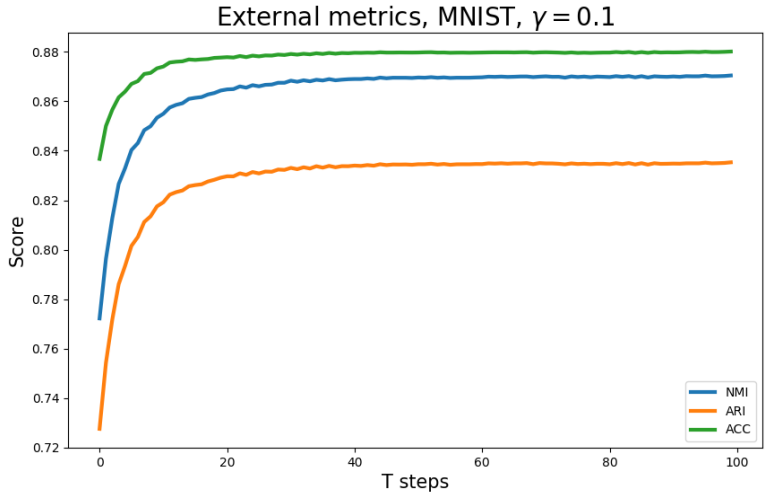


Figure 5.5: External metrics as a function of T (update intervals). T updates twice per epoch.

γ	NMI	ARI	ACC
0.1	0.87	0.83	0.88
0.4	0.86	0.83	0.88
0.6	0.86	0.82	0.88
1	0.86	0.83	0.88

Table 5.5: MNIST external metrics varying γ.

Tissues

This section shows results when clustering Tissues until convergence $\delta = 0.001$. Figure 5.6 shows the external metrics as a function of T similar to figure 5.5. The metrics vary more for this experiment and the resulting external metrics are lower than how initialized.

Table 5.6 shows external metrics for Tissues when varying γ. Here $\gamma = 1$ obtains the best results. The other results are seen in appendix A.1, figure A.2.

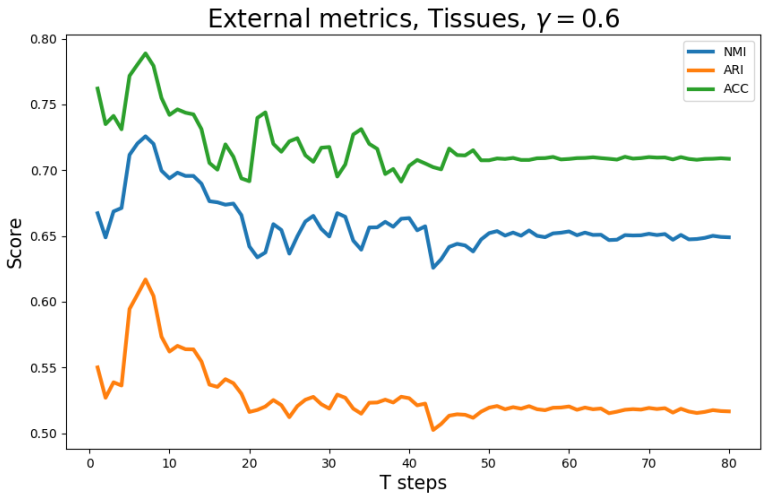


Figure 5.6: External metrics as a function of T for Tissues data. T updates twice per epoch.

γ	NMI	ARI	ACC	Convergence epochs
0.1	0.70	0.55	0.72	60
0.4	0.66	0.55	0.77	47
0.6	0.65	0.52	0.71	40
1	**0.72**	**0.64**	**0.83**	36

Table 5.6: Tissues external metrics results when varying only γ. Experiments converge at different epochs suggesting $\gamma = 1$ being the most efficient and obtaining best scores.

5.3 Tumor experiments

5.3.1 Vary K

Table 5.7 shows the results varying K for the sampled Tumor dataset. Results show the internal scores for three experiments for K=4, 8, 12. Results for CHI vary a lot compared to SSE and DBI as CHI is not a normalized score. Results show that the best score for CHI and SSE is with K=8 clusters and the best score for DBI is K=4 clusters.

K	CHI	SSE	DBI
4	265781	0.78	**0.29**
8	**655364**	**0.79**	0.31
12	326181	0.77	0.37

Table 5.7: Internal metrics for Tumor dataset, $\gamma = 0.4$, for three different number of clusters K. Notice the large variations in CHI values compared to the smaller variations for SSE and DBI.

5.3.2 Vary γ

Table 5.8 shows the internal metrics when varying γ for Tumor. A lower CHI for $\gamma = 0.1$ and $\gamma = 0.4$ suggests that the better choices are $\gamma = 0.6$ or $\gamma = 1$. The overall best choice $\gamma = 0.6$. This result K=4 and $\gamma = 0.6$ is used in section 5.4 to analyze the clustering outcome.

γ	**CHI**	**SSE**	**DBI**
0.1	278930	0.75	0.33
0.4	258890	0.77	0.30
0.6	**330930**	**0.78**	**0.28**
1	309260	0.77	0.29

Table 5.8: Internal metrics for Tumor dataset, K=4. Best scores obtained for $\gamma = 0.6$.

5.4 Clustering outcome

This section includes results with $\gamma = 0.6$, K=4. K=4 is chosen for the highest EPV. The analyses in this section originate from this experiment.

5.4.1 Cluster distributions

Figure 5.7-5.10 show the cluster distributions for PAM50, Lehmann, tumor stage and neoplasm stage. Figure 5.7 shows no clear genetic distinction between the clusters and the basal group. The basal fractions roughly correspond to the overall fraction count of basal-patients (77/98=0.79) from table 3.3. Figure 5.8 shows the Lehman distributions where BL1 has a slightly higher fraction in clusters 0 and 1 than 2 and 3. LAR subtype is higher for clusters 2 and 3 compared to 0 and 1,

cluster 1 has the lowest fraction. Figure 5.9 and 5.10 show no apparent patterns relating clusters to tumor or neoplasm stages.

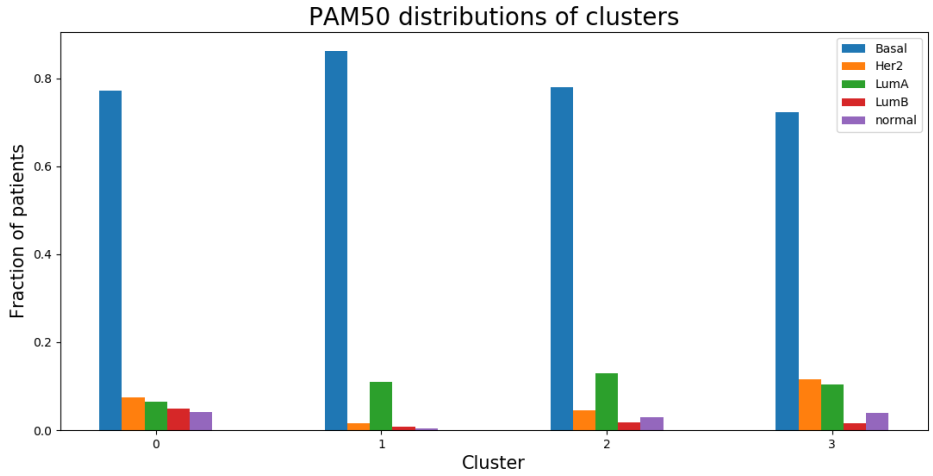


Figure 5.7: PAM50 genetic cluster distributions. Blue=Basal, yellow=Her2, green=LumA, red=LumB, purple=normal.

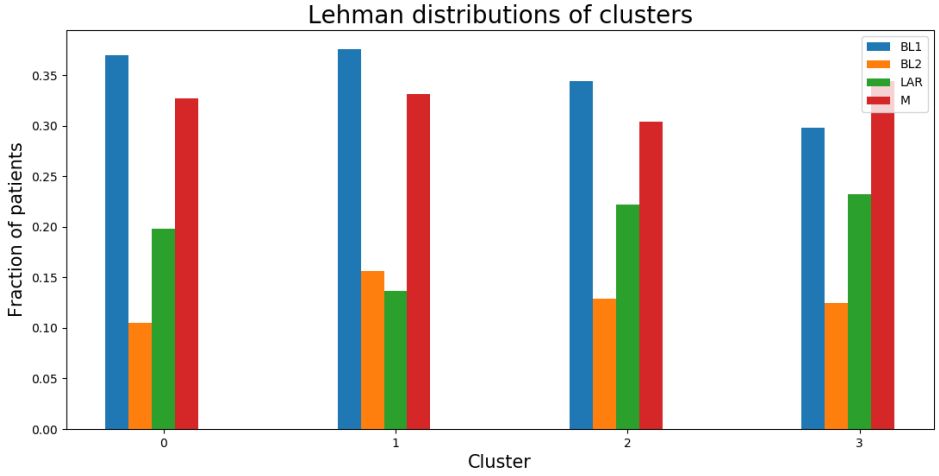


Figure 5.8: Lehman histological cluster distributions. Blue=BL1, yellow=BL2, green=LAR, red=M.

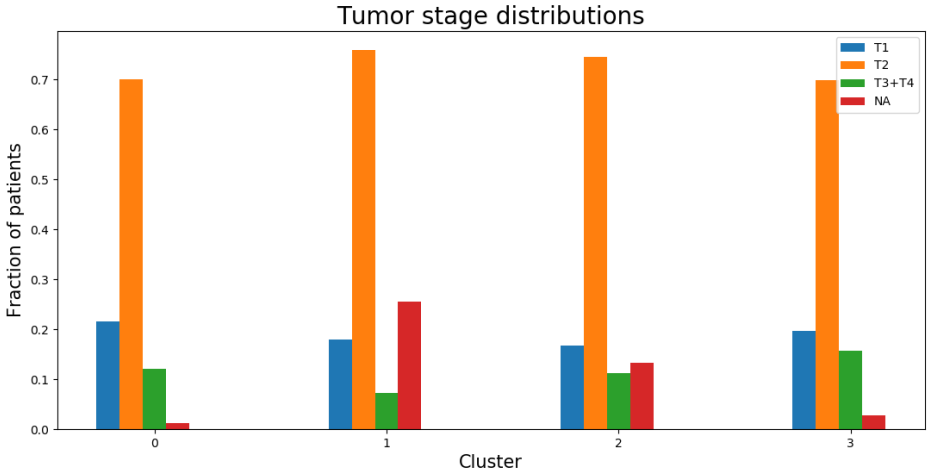


Figure 5.9: Tumor stage cluster distributions. Tumor stages: Blue=I, yellow=II, green=III+IV and red=NA.

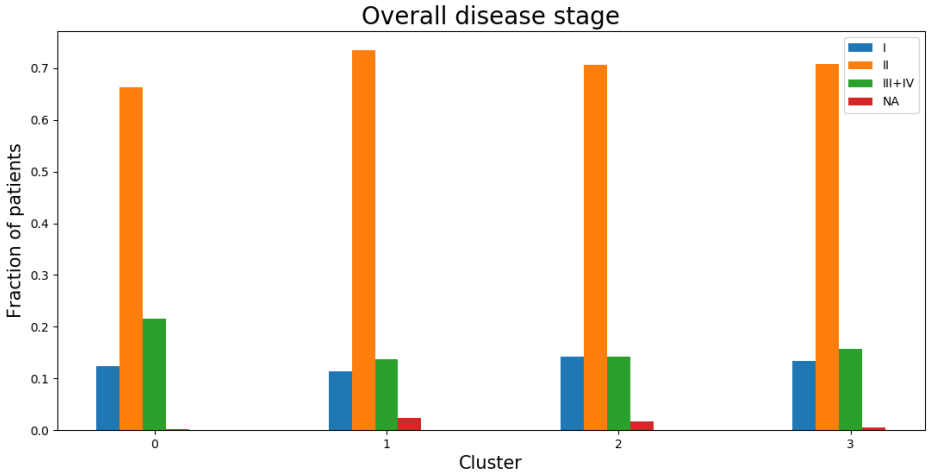


Figure 5.10: Overall neoplasm disease stage cluster distributions. Neoplasm stages: Blue=I, yellow=II, green=III+IV, red=NA.

5.4.2 Cox survival modelling

Figure 5.11 shows the β estimates for the continuous CPH modelling and 5.12 for the binary. The error bars indicate 95% CIs for the coefficient estimates. Results from the continuous modelling in figure 5.11 show coefficient estimates with large CI overlaps around $\beta = 0$ for all 4 clusters. None of the continuous β estimates are significantly related to a better or worse survival. Patients from cluster 0 in figure 5.12 have a significantly better outcome compared to the cluster 0 negative patients.

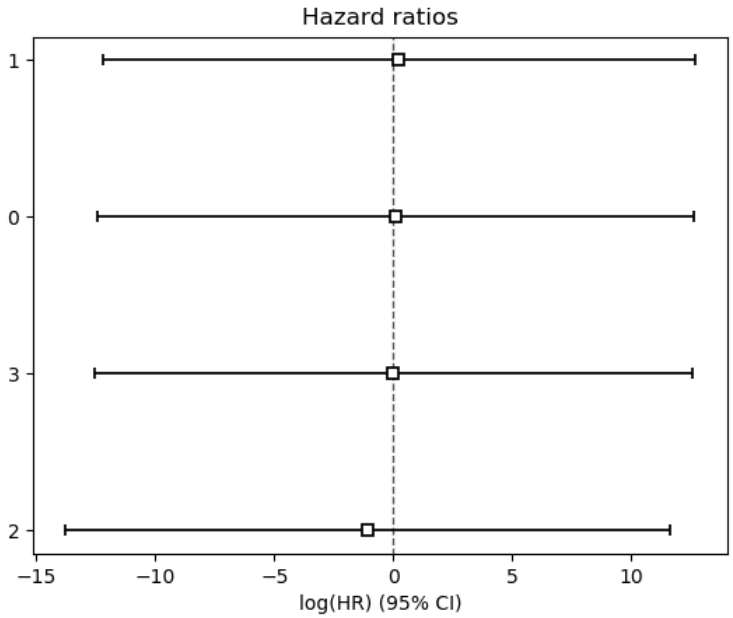


Figure 5.11: β estimates for the continuous covariate modelling.

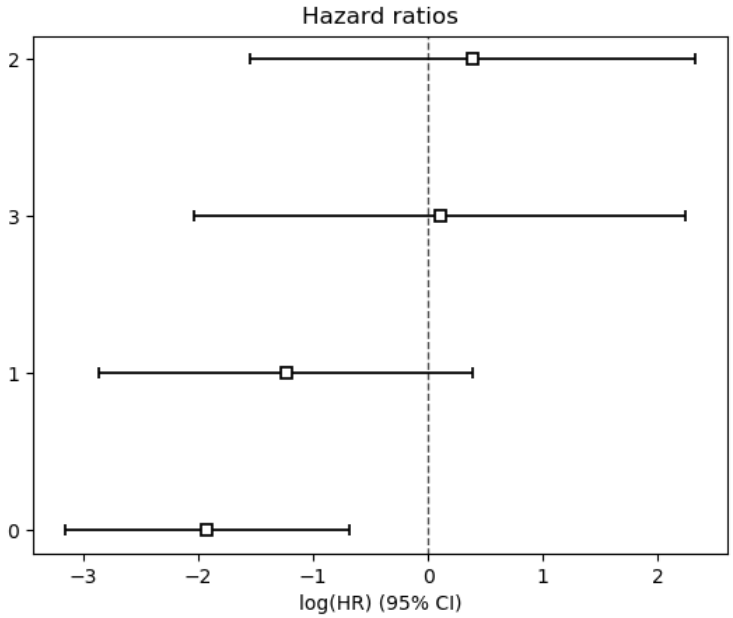


Figure 5.12: β estimates for the binary covariate modelling. Cluster 0 positive patients have significantly better prognosis than cluster 0 negative.

5.4.3 Cluster analysis

This section shows the clustering and survival analysis from the binary modelling. Figure 5.13 shows the survival curves for the cluster 0 positive (n=92) and negative (n=6) patients. Figure 5.14 list some of the most probable tiles in cluster 0 assigned with the highest soft label probability. 311/400 of the most likely tiles come from the same 4 patients with similar stains. Table 5.9 shows the survival

times, Lehmann and PAM50 subtypes for the two groups. Cluster 0 group has $15/92 = 0.16$ event/right censoring ratio and the Others group has $4/6 = 0.67$ event/right censoring ratio. Figure 5.15 shows the distributions for the cluster 0 negative patients. Note how most of the tiles are assigned to cluster 1.

Figure 5.16 shows 10 tiles for each cluster that have been assigned with the lowest (a) and highest (b) probabilities. In figure 5.16a clusters 0, 1 and 2 have tiles with fatty regions, suggesting that these are the hardest tiles to cluster. Figure 5.16b shows that the most likely tiles in cluster 0 have similar stain as in figure 5.14, cluster 1 has tiles with similar characteristic stain and clusters 2 and 3 have similar stain colors. This result suggests that there is a residual stain variation despite the use of Macenko stain normalization.

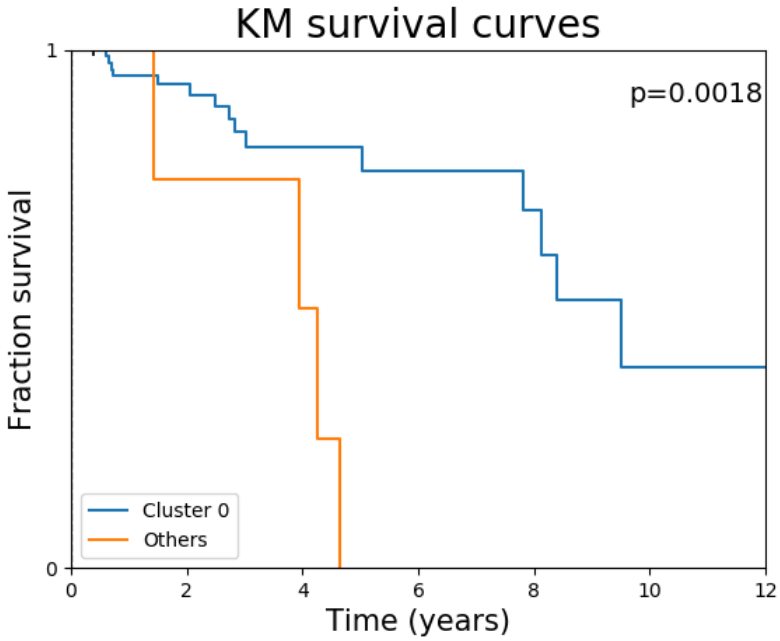


Figure 5.13: Survival curves for cluster 0 (n=92) vs. other (n=6).

ID	Counts
A8-A08R	102
A8-A07R	84
A8-A07O	81
A8-A07C	44

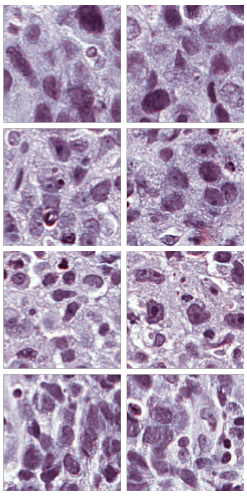


Figure 5.14: 311/400 of the most probable tiles are from the A8- patients with similar stains.

	Survival			Lehman			
	n	Time	Events	BL1	BL2	LAR	M
Cluster 0	92	3.02	15	33	11	17	31
Others	6	2.62	4	1	2	2	1
	PAM50						
	Basal	Her2	LumA	LumB	Normal		
Cluster 0	73	6	8	2	3		
Others	4	-	2	-	-		

Table 5.9: Patient subtypes for cluster 0 (n=92) and others (n=6).

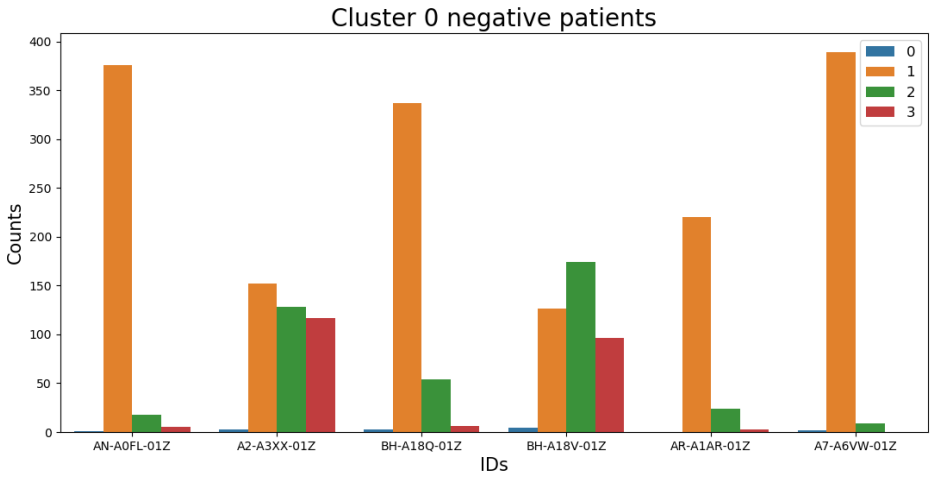
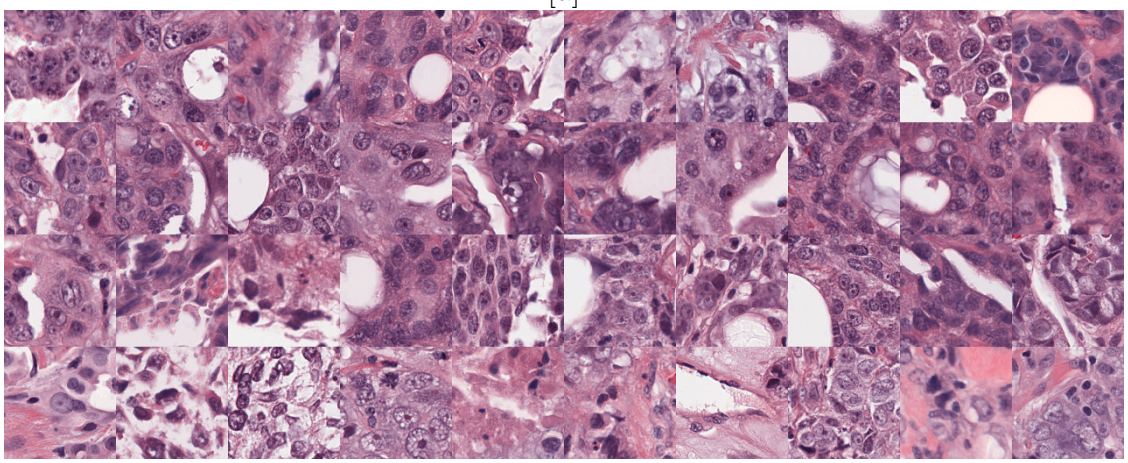


Figure 5.15: Cluster distributions for the 6 cluster 0 negative patients.

[a]



[b]

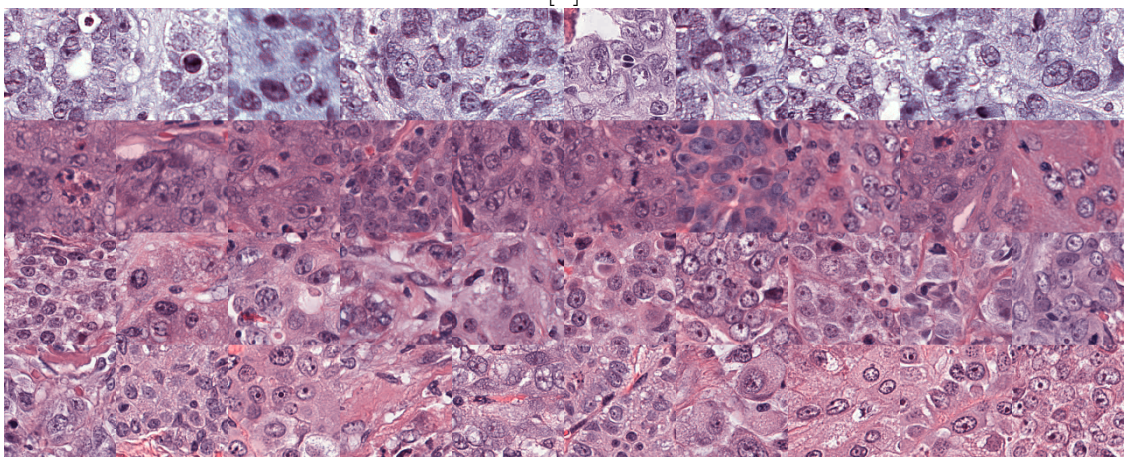


Figure 5.16: The 10 least (a) and most (b) probable tiles for each cluster.

6

Discussion

6.1 Tile extraction

The ROIs in the Tissues dataset were marked manually and the tissue tiles are inherently biased towards how the ROIs are marked. Because of the abundance of fat and stroma tissues, extraction of these tissue types was overall consistent and homogeneous as it was possible to mark large ROIs and therefor extract tiles from a homogeneous ROI. Extracting the lymphocytes was less consistent as the lymphocyte ROIs were much smaller and the lymphocyte tiles have larger overlaps to surrounding tissues.

Tumor infiltrating lymphocytes (TILs) are regions in the WSI where lymphocyte cells have infiltrated the tumor cells, that have been associated with an overall better prognosis (15). The ROIs from which lymphocyte or tumor tiles are extracted might have been TILs. The tile extraction procedure in this thesis does not take any consideration towards TILs which was considered beyond the objective for this thesis. The tile extraction of TILs could be interesting to include a more detailed disease pattern for survival analysis. Given the small lymphocyte ROIs and TILs, labels for lymphocytes and tumors are, to a certain degree, expected to be overlapping. The labels lymphocytes are therefore not as certain as the fat or stroma labels.
This shows the reality of working with WSIs - tissues are not as separate and split and tissues overlap. The Tissues dataset has given valuable insights to how tissue types are clustered and works well as a dataset to benchmark the unsupervised learning methods.

The tumor ROIs were marked automatically with a pretrained neural network. The only quality assessment of the tumor ROIs was by qualitative visual inspection on few ROIs. Partly evaluation is inadequate as it does not account for all the tumor ROIs. A full evaluation of all tumor ROIs would however be very time consuming and is infeasible. A skilled pathologist would further be required to evaluate the tumor detection app, which was beyond the scope of this thesis. Thus, the automated tumor predictions were assumed to be correct.

Edge erosion of the tumor edges was done on the tumor ROIs. The amount of erosion depends on the ROI shape. From the criteria regarding the area size of the

tumor (area of at least 10 tiles) and edge erosion by one tile width, 9 WSIs were excluded.
How strictly to define the criteria for tumor size and mathematical erosion is an interesting subject for further research. For instance imagine setting a smaller threshold to tumor area size. This could give a detailed insight to smaller tumor micro-environments as they would otherwise have been excluded. On the contrary, a smaller area threshold could also give rise to more false positives - including tissues that are actually healthy. The threshold goes well in hand with the earlier discussion regarding the manually extracted labeled tissue types: how much can and should one put in a label and how much is this label part of a larger histological context. For example consider the local tumor growth in figure 6.1. The figure shows the same 5X area from a WSI where tumor regions have infiltrated otherwise healthy tissues. The tumor ROIs are enclosed within the blue masks where the top figure is before and the bottom is after edge erosion.

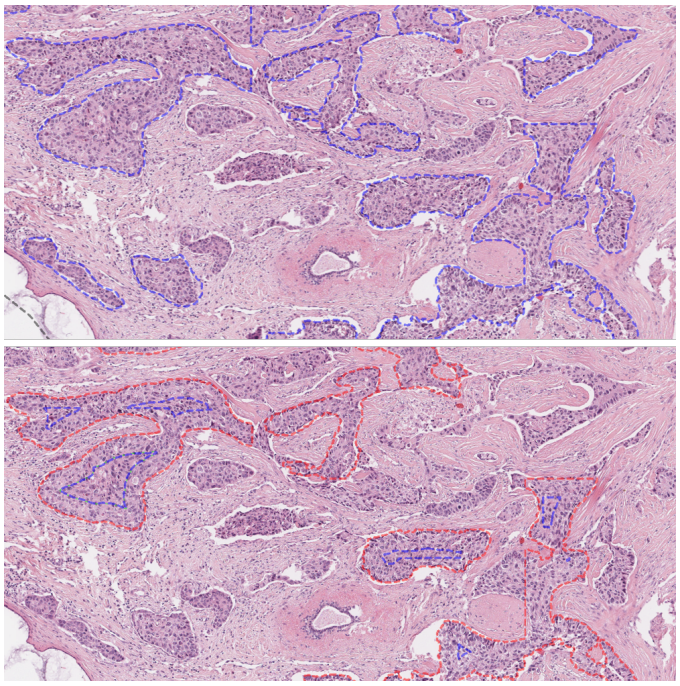


Figure 6.1: Effect of edge erosion. Blue masks are tumor ROIs. Top slide is before erosion and bottom is after. ROIs are eroded by the length corresponding to the perpendicular distance red:blue.

This example shows an important consequence of the tumor extraction method: the extracted tiles after edge erosion will reflect more tumor morphology and take less account to neighbouring tissues. Note how much small tumor ROIs close to healthy tissues are reduced. The consequence of the edge erosion is that infiltrating tumors with thin lines like in figure 6.1 will have smaller importance as they are removed.

6.2 Baseline comparison

Two clustering methods Naive K-means and DCEC are used. The baseline method comparison shows that DCEC does a better job at clustering both MNIST and Tis-

sues dataset. The models share a lot of common traits: They are based on the same CAE network architecture, they are both pretrained and optimized with respect to the MSE loss, they cluster latent representations of similar dimensions (R^{10} for MNIST and R^{512} for Tissues and Tumor dataset). The trainable centroids in DCEC are even initialized by assigning K centroids.

The difference between the methods is the joint optimization for DCEC as the methods are otherwise identical. This experiment shows how clustering differs if we model the latent representation to follow a distribution P and assign cluster labels q_{ij} (eq. 2.10 and 2.9) and optimize the KL divergence given Q and P.

Two experiments were carried out for the MNIST and Tissues datasets. Both MNIST and Tissues (section 5.1.1 and 5.1.2) show how the external metrics generally are higher for DCEC than for Naive K-means. A higher NMI suggests that DCEC assigns clusters that are more correlated to the class labels which is also confirmed by a higher ARI (label assignments are less random) and ACC.

The class imbalances in table 3.2 bring some biases to the metrics. The fat class for the test partition (n=963) is much higher than the other classes. Indeed this will result in an artificially high external metrics when testing how well the the models generalize (see table. 5.2). It is therefore not representative to evaluate the external metrics independently for the test partition, but rather to compare the external metrics between Naive K-means and DCEC comparatively. For future improvement the test partition for the Tissues dataset should be balanced equally between the classes to avoid inflated metrics like an artificially high ACC because most fat tiles are assigned to the same cluster.
The number of tumor tiles from the training partition is intentionally about twice the size of the other classes. The motivation for doing so is to have a more varied tumor dataset revealing more aspects to tumor heterogeneity. The results of tumor-tumor variations is shown best when varying K for the Tissues dataset in figure 5.4.

The baseline experiment shows the comparison between two methods with the same CAE architecture. Interesting future work is to develop better and more sophisticated CAEs that might capture more complex and high-dimensional features from the latent representations. The idea is with inspiration from Xie *et al.* (40) as described in section 2.2.3, where they investigated the differences between Inception V2 and V3 ResNets.
A potential pitfall to deep clustering methods is that they might fail to reveal the underlying structure of data. To investigate the underlying structures of data it could be interesting to investigate generative models. The concept for generative models is to uncover more generic data structures and be able to generate new samples. An interesting example of such is the Variational Autoencoder (VAE) (17). VAE is a generative AE that models the latent representation to follow a defined a priori distribution. The advantage of VAE is that it will be able to generate samples as well as working as an AE. VAE does however suffer from higher computational complexity as it optimizes with respect to two objectives like DCEC. The design of the loss function is important for convergence as a poorly designed objective function could corrupt the feature space and therefore generate samples that do not resemble a meaningful distribution.

6.3 DCEC experiments

Experiments for DCEC were carried out to evaluate how central parameters K (number of clusters) and γ (clustering weight) influence the model. Section 5.2.1 shows selected results obtained from from varying K for MNIST (K=12) and Tissues (K=8). The motivation for varying K was to see how potential subsets of images with similarities would appear. Note that no effort was made in merging cluster labels to keep external metrics like ACC or ARI. Instead these experiments show qualitative characteristics.

For MNIST notice how the clusters for low K=8 (table A.2 and figure A.4) show how digits 4 and 9 are assigned to the same cluster suggesting high similarities between the two. For high K=16 (figure A.6 and table A.4) how the digits split to subtypes. Digit 1 is split in cluster 1 and 13 (row 2 and 14) in figure A.6 where cluster 1 shows italic and cluster 13 shows straight "ones".
The experiments Varying K for the Tissues dataset show that although K clusters were initialized, all the clusters were not necessarily assigned once the network had converged. Notice how figure A.8 and A.9 have K= 12 and 20 clusters initialized respectively, but only 10 and 12 clusters are returned respectively. From a data-driven point of view this shows that for high K the model may converge towards another solution than initiated that will reduce the KL divergence error. Although the model might converge towards a solution that is better from a KL divergence objective, this behaviour is not always attractive. If there is no lower bound in the method to how few clusters it should assign it might converge towards assigning all samples to the same cluster. This is an unwanted trait as it makes cluster analysis meaningless with only one cluster label. This trait could be part of the explanation to why the model diverges with a LR that is too high, namely that the model during training decides to cluster all samples to a pseudo-convergence.

Results from Tissues further indicate that tiles are clustered dependent on their color. This is not obvious for K=4, but appears more clearly for K=8 clusters. Notice how cluster 4 (row 5 in figure 5.4) has a brighter pink color. From the 263 tiles assigned to cluster 4 almost all the tiles come from same patients (n=260 tiles). This brings suspicion to an unwanted color dependency - the algorithm learns to cluster with respect to colors. This takes away focus from the original objective, to cluster based on morphology. This is the reason why Macenko stain normalization described in section 4.3.2 was used on the Tumor dataset.

6.4 Tumor experiments

For the Tumor dataset internal metrics were used to evaluate the experiments. For computational considerations only a subset of the data was used to train the clustering network. This was done by sampling up to 400 tiles from each WSI randomly balancing the tiles from each patient.

For future work sampling a data subset differently could be interesting to investigate. An interesting approach to sampling a subset would be sampling technique 2 mentioned by Naylor *et al.* (29) (see literature review section 2.2.3). Here they

firstly cluster feature vectors in a subspace and subsequently sample feature vectors from each cluster. A sampling like this would give a more stratified type of sampling and according to Naylor *et al.* this improves the performance in their downstream prediction.

Table 5.7 shows the internal metrics for varying K=4, 8 and 12. CHI and SSE scores suggest that K=8 is the best cluster while the best DBI score suggests K=4. In general CHI score varies a lot in value, while the values for SSE and DBI seem to suggest smaller differences for the different values of K. This comes from the fact that CHI is not limited by an upper boundary. It is therefore good to have SSE and DBI as additional internal metrics evaluating the clustering quality because they give a normalized suggestion to the best clustering outcomes. Apart from CHI, the metrics look much alike. The internal metrics do however decrease going from K=8 to K=12 suggesting that K=12 is not be the optimal number of clusters.

Internal metrics have their limitations because they only score the between and inter-clusters dispersion. That does not indicate that the content of the clusters is relevant, only a matter of how separated the clusters are. Figure 5.2b shows how tissue types are clustered without a stop criterion ($\delta = 0$). Because the stop criterion in the baseline method comparison is removed, the clusters have been squeezed tightly together. Although the clusters are well separated, lymphocytes and tumors are still overlapping a lot, which confirms that a high internal metric does not guarantee that tiles with similar morphology are clustered together. Sampling cluster tiles is a good additional approach to evaluate the clusters qualitatively.
Because of a low EPV ratio, K=4 was chosen as the number of clusters to evaluate further. Table 5.8 shows the outcome of varying gamma and the results indicate that $\gamma = 0.6$ is the best clustering weight to cluster the Tumor dataset.

6.5 Clustering outcome

Cluster distributions were investigated with respect to PAM50, Lehman, tumor stage and overall disease stage. Figure 5.7 shows the distribution of the PAM50 subtypes. Here Basal-like is the most frequently occurring genetic subtype and only minor deviations for HER2, LumA, LumB and normal are seen. Figure 5.9 and 5.10 show the cluster distributions for tumor and neoplasm disease stages. The hypothesis for tumor and disease stages was that low stage tumors like stage I and II could be identified indicating a better prognosis. The cluster distributions do however look much alike and no noteable trends are seen. The results show that there is a large overall correlation with the cluster distributions and the WSI level subtypes from table 3.3. A potential issue with assigning labels to tiles in this manner is loss of intra tumor heterogenity. The problem with this kind of assignment is that all tiles are labeled equally to the slide-level. If a patient is Lehmann subtype LAR, then all tiles from that patient are labeled as LAR. One could imagine that intra tumor heterogeneity however gives rise to histologically local tumor regions that show other morphology patterns than what it is assigned from a slide-level.

6.6 Survival modelling

Two covariates, continuous and binary, were modelled to fit CPH models. The continuous covariate matrix was to identify potential interactions between clusters that could give a better insight to survival outcome. L1 normalization turns out to model the covariate matrix in a manner that creates highly linear β estimates that look alike. L1 normalization is a good way of balancing weights across patients as a few patients in this thesis had less than 400 tumor tiles in total. L1 normalization balances these patients to the same scale so each row sums to 1. The actual outcome of the L1 normalization is that it scales almost all elements with 1/400 as most patients have 400 tiles, and the covariate $\mathbf{x}$ for the continuous modelling mostly reflects a scaled version of the count matrix $\mathbf{m}$ described in section 4.7.2. Figure 5.12 shows the results from the binary modelling. Cluster 0 shows evidence for a significantly better outcome being positive for that cluster.

To further analyze the content of cluster 0, survival curves for cluster 0 positive patients were plotted against the patients not in cluster 0. Figure 5.13 shows the two survival curves, where p=0.0018 indicates a significant difference between cluster 0 (n=92) and others (n=6). Table 5.9 shows the survival times, histology subtype and PAM50. Note how 15/92 (16%) and 4/6 (66%) of the patients die for cluster 0 and others average follow-up times of 3.02 and 2.62 years respectively. This is a large difference in fraction of patient deaths, but should be analyzed cautiously because of the small population size in the others group.
10 tiles from each cluster were sampled to investigate what is the most and least likely characteristic for each cluster. Notice how cluster 0 in figure 5.16b has its own slightly faint stain intensity, 1 has a darker, more intense stain and how clusters 2 and 3 have similar stain intensities. Results from figure 5.16b suggests that the most characteristic tiles are based on stain intensity.

The apparent confounding between the stain variations and the clusters makes it hard to assess histology as a determinant to survival outcome. Stain dependency disturbs how the covariate matrix is modelled: if the lightly stained patients from cluster 0 have a better overall survival the survival outcome will depend on staining. A clear conclusion to this part of the experiment is that the Macenko stain normalization described in section 4.3.2 has not stain normalized the tiles to a satisfactory degree. The Macenko stain normalization works by quantile scaling tiles individually, and transforms images as seen in figure 4.5. There are still residuals with variations, which needs improvements for future work. Quantile normalizing tiles individually turns out to be a normalization that leaves stain residuals. Another stain normalization that could be interesting to investigate is presented by Vahadane *et al.* (37). This method normalizes all tiles relative to a pre-defined target image making up for a more homogeneous yet biased stain. The outcome of this normalization would scale the tiles relative to a fixed HE target image, but decrease generalization to stain variations. Stain augmentation as presented by Tellez *et al.* (36) could be an interesting data augmentation method, as it will compensate for the homogeneous Vahadane stain normalization. Stain augmentation works by varying the tile stains artificially by changing brightness and contrast while keeping the morphology unchanged. Tellez quantifies the effects of various stain augmen-

tation methods on CNNs and concludes that stain augmentation should always be used in order to improve generalization of the CNN and make network architectures more robust against stain variations.

7

Conclusion

Triple negative breast cancer (TNBC) is heterogeneous both genetically and histologically. The objective of this thesis was to use deep unsupervised clustering to identify morphological patterns of TNBC and relate these to known subtypes and survival outcome. Experiments showed that Deep Convolutional Embedded Clustering (DCEC) was a better unsupervised method in clustering images compared to the simpler K-means clustering of a latent representation from a convolutional autoencoder.

With K=4 clusters for the DCEC model, the tumor tiles were clustered and analyzed. The clustering outcome showed no apparent correlation to either the genetic PAM50 or histological Lehmann subtypes or the tumor stages (tumor size and neoplasm disease).
The cluster distributions from the tumor tiles were used as a covariate to model the Cox proportional hazard model to relate survival outcomes with the clusters. A binary covariate was modelled by assigning patients as positive to a cluster if more than 4 tiles were assigned to a given cluster and otherwise negative. The binary survival modelling showed that positive patients from a single cluster had a significantly better outcome compared to patients negative of the same cluster.

The tumor tiles were stain normalized using Macenko normalization to reduce stain variations between patients. Despite the use of stain normalization, clustering analysis still correlated to a residual stain variation. This implies that the outcome of the survival analysis is confounded with a stain variation, which impairs the validity of the survival outcome.
To improve unsupervised clustering of TNBC histology, future work should focus on implementing a data pre-processing procedure that reduces stain variations and stain augmentations to make a neural network more robust to stain variations.

Bibliography

[1] M. L. W. D. M.-J. Acharyya, S. *The Molecular Basis of Cancer*. Elsevier Inc., 4 edition, 2014.

[2] O. Arbelaitz, I. Gurrutxaga, J. Muguerza, J. M. Pérez, and I. Perona. An extensive comparative study of cluster validity indices. *Pattern Recognition*, 46(1):243–256, 2013.

[3] J. Ash, G. Darnell, D. Munro, and B. Engelhardt. Joint analysis of gene expression levels and histological images identifies genes associated with tissue morphology. *bioRxiv*, page 458711, 2018.

[4] P. S. Bernard, J. S. Parker, M. Mullins, M. C. Cheung, S. Leung, D. Voduc, T. Vickery, S. Davies, C. Fauron, X. He, Z. Hu, J. F. Quackenhush, I. J. Stijleman, J. Palazzo, J. S. Matron, A. B. Nobel, E. Mardis, T. O. Nielsen, M. J. Ellis, and C. M. Perou. Supervised risk predictor of breast cancer based on intrinsic subtypes. *Journal of Clinical Oncology*, 27(8):1160–1167, 2009.

[5] F. M. Blows, K. E. Driver, M. K. Schmidt, A. Broeks, F. E. van Leeuwen, J. Wesseling, M. C. Cheang, K. Gelmon, T. O. Nielsen, C. Blomqvist, P. Heikkila, T. Heikkinen, H. Nevanlinna, L. A. Akslen, L. R. Bégin, W. D. Foulkes, F. J. Couch, X. Wang, V. Cafourek, J. E. Olson, L. Baglietto, G. G. Giles, G. Severi, C. A. McLean, M. C. Southey, E. Rakha, A. R. Green, I. O. Ellis, M. E. Sherman, J. Lissowska, W. F. Anderson, A. Cox, S. S. Cross, M. W. Reed, E. Provenzano, S. J. Dawson, A. M. Dunning, M. Humphreys, D. F. Easton, M. García-Closas, C. Caldas, P. D. Pharoah, and D. Huntsman. Subtyping of breast cancer by immunohistochemistry to investigate a relationship between subtype and short and long term survival: A collaborative analysis of data for 10,159 cases from 12 studies. *PLoS Medicine*, 7(5), 2010.

[6] N. E. Breslow. Analysis of Survival Data under the Proportional Hazards Model. *International Statistical Review / Revue Internationale de Statistique*, 43(1):45–57, 1975.

[7] I. Carmichael, B. C. Calhoun, K. A. Hoadley, M. A. Troester, J. Geradts, H. D. Couture, L. Olsson, C. M. Perou, M. Niethammer, J. Hannig, and J. S. Marron. Joint and individual analysis of breast cancer histologic images and genomic covariates. pages 1–31, 2019.

[8] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. *arXiv e-prints*, page arXiv:1802.02611, 2 2018.

[9] A. M. Chiu, M. Mitra, L. Boymoushakian, and H. A. Coller. Integrative analysis of the inter-tumoral heterogeneity of triple-negative breast cancer. *Scientific Reports*, 8(1):1–14, 2018.

[10] D. Collett. *Modelling Survival Data in Medical Research*. CRC Press, 2 edition, 2003.

[11] N. Dimitriou, O. Arandjelović, and P. D. Caie. Deep Learning for Whole Slide Image Analysis: An Overview. *Frontiers in Medicine*, 6(November):1–7, 2019.

[12] C. Fitzmaurice, T. F. Akinyemiju, F. H. Al Lami, T. Alam, R. Alizadeh-Navaei, C. Allen, U. Alsharif, N. Alvis-Guzman, E. Amini, B. O. Anderson, O. Aremu, A. Artaman, S. W. Asgedom, R. Assadi, T. M. H. Atey, L. Avila-Burgos, A. Awasthi, H. O. Saleem, A. Barac, J. R. Bennett, I. M. Bensenor, N. Bhakta, H. Brenner, L. Cahuana-Hurtado, C. A. Castañeda-Orjuela, F. Catalá-López, J. Y. J. Choi, D. J. Christopher, S. C. Chung, M. P. Curado, L. Dandona, R. Dandona, J. Das Neves, S. Dey, S. D. Dharmaratne, D. T. Doku, T. R. Driscoll, M. Dubey, H. Ebrahimi, D. Edessa, Z. El-Khatib, A. Y. Endries, F. Fischer, L. M. Force, K. J. Foreman, S. W. Gebrehiwot, S. V. Gopalani, G. Grosso, R. Gupta, B. Gyawali, R. R. Hamadeh, S. Hamidi, J. Harvey, H. Y. Hassen, R. J. Hay, S. I. Hay, B. Heibati, M. K. Hiluf, N. Horita, H. D. Hosgood, O. S. Ilesanmi, K. Innos, F. Islami, M. B. Jakovljevic, S. C. Johnson, J. B. Jonas, A. Kasaeian, T. D. Kassa, Y. S. Khader, E. A. Khan, G. Khan, Y. H. Khang, M. H. Khosravi, J. Khubchandani, J. A. Kopec, G. A. Kumar, M. Kutz, D. P. Lad, A. Lafranconi, Q. Lan, Y. Legesse, J. Leigh, S. Linn, R. Lunevicius, A. Majeed, R. Malekzadeh, D. C. Malta, L. G. Mantovani, B. J. McMahon, T. Meier, Y. A. Melaku, M. Melku, P. Memiah, W. Mendoza, T. J. Meretoja, H. B. Mezgebe, T. R. Miller, S. Mohammed, A. H. Mokdad, M. Moosazadeh, P. Moraga, S. M. Mousavi, V. Nangia, C. T. Nguyen, V. M. Nong, F. A. Ogbo, A. T. Olagunju, P. A. Mahesh, E. K. Park, T. Patel, D. M. Pereira, F. Pishgar, M. J. Postma, F. Pourmalek, M. Qorbani, A. Rafay, S. Rawaf, D. L. Rawaf, G. Roshandel, S. Safiri, H. Salimzadeh, J. R. Sanabria, M. M. Milicevic, B. Sartorius, M. Satpathy, S. G. Sepanlou, K. A. Shackelford, M. A. Shaikh, M. Sharif-Alhoseini, J. She, M. J. Shin, I. Shiue, M. G. Shrime, A. H. Sinke, M. Sisay, A. Sligar, M. B. Sufiyan, B. L. Sykes, R. Tabarés-Seisdedos, G. A. Tessema, R. Topor-Madry, T. T. Tran, B. X. Tran, K. N. Ukwaja, V. V. Vlassov, S. E. Vollset, E. Weiderpass, H. C. Williams, N. B. Yimer, N. Yonemoto, M. Z. Younis, C. J. Murray, and M. Naghavi. Global, regional, and national cancer incidence, mortality, years of life lost, years lived with disability, and disability-adjusted life-years for 29 cancer groups, 1990 to 2016 a systematic analysis for the global burden of disease study global burden o. *JAMA Oncology*, 4(11):1553–1568, 2018.

[13] H. Gonçalves, M. R. Guerra, J. R. Duarte Cintra, V. A. Fayer, I. V. Brum, and M. T. Bustamante Teixeira. Survival Study of Triple-Negative and Non–Triple-Negative Breast Cancer in a Brazilian Cohort. *Clinical Medicine Insights: Oncology*, 12, 2018.

[14] X. Guo, X. Liu, E. Zhu, and J. Yin. Deep Clustering with Convolutional Autoencoders. *Lecture Notes in Computer Science (including subseries Lecture*

Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 10635 LNCS:373–382, 2017.

[15] S. Hendry, R. Salgado, T. Gevaert, P. Russell, T. John, B. Thapa, M. Christie, K. Van de Vijver, and V. Estrada. *Assessing the host immune response, TILs in invasive breast carcinoma and ductal carcinoma in situ, metastatic tumor deposits and areas for further research*, volume 24. 2018.

[16] R. S. L.-J. L. K. L. Jia Deng, Wei Dong and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. 2009.

[17] D. P. Kingma and M. Welling. Auto-Encoding Variational Bayes. (Ml):1–14, 2013.

[18] D. C. Koboldt, R. S. Fulton, M. D. McLellan, H. Schmidt, J. Kalicki-Veizer, J. F. McMichael, L. L. Fulton, D. J. Dooling, L. Ding, E. R. Mardis, R. K. Wilson, A. Ally, M. Balasundaram, Y. S. Butterfield, R. Carlsen, C. Carter, A. Chu, E. Chuah, H. J. E. Chun, R. J. Coope, N. Dhalla, R. Guin, C. Hirst, M. Hirst, R. A. Holt, D. Lee, H. I. Li, M. Mayo, R. A. Moore, A. J. Mungall, E. Pleasance, A. G. Robertson, J. E. Schein, A. Shafiei, P. Sipahimalani, J. R. Slobodan, D. Stoll, A. Tam, N. Thiessen, R. J. Varhol, N. Wye, T. Zeng, Y. Zhao, I. Birol, S. J. Jones, M. A. Marra, A. D. Cherniack, G. Saksena, R. C. Onofrio, N. H. Pho, S. L. Carter, S. E. Schumacher, B. Tabak, B. Hernandez, J. Gentry, H. Nguyen, A. Crenshaw, K. Ardlie, R. Beroukhim, W. Winckler, G. Getz, S. B. Gabriel, M. Meyerson, L. Chin, R. Kucherlapati, K. A. Hoadley, J. T. Auman, C. Fan, Y. J. Turman, Y. Shi, L. Li, M. D. Topal, X. He, H. H. Chao, A. Prat, G. O. Silva, M. D. Iglesia, W. Zhao, J. Usary, J. S. Berg, M. Adams, J. Booker, J. Wu, A. Gulabani, T. Bodenheimer, A. P. Hoyle, J. V. Simons, M. G. Soloway, L. E. Mose, S. R. Jefferys, S. Balu, J. S. Parker, D. N. Hayes, C. M. Perou, S. Malik, S. Mahurkar, H. Shen, D. J. Weisenberger, T. Triche, P. H. Lai, M. S. Bootwalla, D. T. Maglinte, B. P. Berman, D. J. Van Den Berg, S. B. Baylin, P. W. Laird, C. J. Creighton, L. A. Donehower, M. Noble, D. Voet, N. Gehlenborg, D. Di Cara, J. Zhang, H. Zhang, C. J. Wu, S. Yingchun Liu, M. S. Lawrence, L. Zou, A. Sivachenko, P. Lin, P. Stojanov, R. Jing, J. Cho, R. Sinha, R. W. Park, M. D. Nazaire, J. Robinson, H. Thorvaldsdottir, J. Mesirov, P. J. Park, S. Reynolds, R. B. Kreisberg, B. Bernard, R. Bressler, T. Erkkila, J. Lin, V. Thorsson, W. Zhang, I. Shmulevich, G. Ciriello, N. Weinhold, N. Schultz, J. Gao, E. Cerami, B. Gross, A. Jacobsen, R. Sinha, B. A. Aksoy, Y. Antipin, B. Reva, R. Shen, B. S. Taylor, M. Ladanyi, C. Sander, P. Anur, P. T. Spellman, Y. Lu, W. Liu, R. R. Verhaak, G. B. Mills, R. Akbani, N. Zhang, B. M. Broom, T. D. Casasent, C. Wakefield, A. K. Unruh, K. Baggerly, K. Coombes, J. N. Weinstein, D. Haussler, C. C. Benz, J. M. Stuart, S. C. Benz, J. Zhu, C. C. Szeto, G. K. Scott, C. Yau, E. O. Paull, D. Carlin, C. Wong, A. Sokolov, J. Thusberg, S. Mooney, S. Ng, T. C. Goldstein, K. Ellrott, M. Grifford, C. Wilks, S. Ma, B. Craft, C. Yan, Y. Hu, D. Meerzaman, J. M. Gastier-Foster, J. Bowen, N. C. Ramirez, A. D. Black, R. E. Pyatt, P. White, E. J. Zmuda, J. Frick, T. M. Lichtenberg, R. Brookens, M. M. George, M. A. Gerken, H. A. Harper, K. M. Leraas, L. J. Wise, T. R. Tabler, C. McAllister, T. Barr, M. Hart-Kothari, K. Tarvin, C. Saller, G. Sandusky, C. Mitchell, M. V. Iacocca, J. Brown, B. Rabeno, C. Czerwinski, N. Pe-

trelli, O. Dolzhansky, M. Abramov, O. Voronina, O. Potapova, J. R. Marks, W. M. Suchorska, D. Murawa, W. Kycler, M. Ibbs, K. Korski, A. Spychała, P. Murawa, J. J. Brzeziński, H. Perz, R. Łaźniak, M. Teresiak, H. Tatka, E. Leporowska, M. Bogusz-Czerniewicz, J. Malicki, A. Mackiewicz, M. Wiznerowicz, X. Van Le, B. Kohl, N. Viet Tien, R. Thorp, N. Van Bang, H. Sussman, B. D. Phu, R. Hajek, N. P. Hung, T. V. T. Phuong, H. Q. Thang, K. Z. Khan, R. Penny, D. Mallery, E. Curley, C. Shelton, P. Yena, J. N. Ingle, F. J. Couch, W. L. Lingle, T. A. King, A. M. Gonzalez-Angulo, M. D. Dyer, S. Liu, X. Meng, M. Patangan, F. Waldman, H. Stöppler, W. K. Rathmell, L. Thorne, M. Huang, L. Boice, A. Hill, C. Morrison, C. Gaudioso, W. Bshara, K. Daily, S. C. Egea, M. D. Pegram, C. Gomez-Fernandez, R. Dhir, R. Bhargava, A. Brufsky, C. D. Shriver, J. A. Hooke, J. L. Campbell, R. J. Mural, H. Hu, S. Somiari, C. Larson, B. Deyarmin, L. Kvecher, A. J. Kovatich, M. J. Ellis, T. Stricker, K. White, O. Olopade, C. Luo, Y. Chen, R. Bose, L. W. Chang, A. H. Beck, T. Pihl, M. Jensen, R. Sfeir, A. Kahn, A. Chu, P. Kothiyal, Z. Wang, E. Snyder, J. Pontius, B. Ayala, M. Backus, J. Walton, J. Baboud, D. Berton, M. Nicholls, D. Srinivasan, R. Raman, S. Girshik, P. Kigonya, S. Alonso, R. Sanbhadti, S. Barletta, D. Pot, M. Sheth, J. A. Demchok, K. R. Shaw, L. Yang, G. Eley, M. L. Ferguson, R. W. Tarnuzzer, J. Zhang, L. A. Dillon, K. Buetow, P. Fielding, B. A. Ozenberger, M. S. Guyer, H. J. Sofia, and J. D. Palchik. Comprehensive molecular portraits of human breast tumours. *Nature*, 490(7418):61–70, 2012.

[19] S. Koren and M. Bentires-Alj. Breast Tumor Heterogeneity: Source of Fitness, Hurdle for Therapy. *Molecular Cell*, 60(4):537–546, 2015.

[20] A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*, 2:1097–1105, 2012.

[21] H. W. Kuhn. The Hungarian Method for the Assignment Problem. In M. Jünger, T. M. Liebling, D. Naddef, G. L. Nemhauser, W. R. Pulleyblank, G. Reinelt, G. Rinaldi, and L. A. Wolsey, editors, *50 Years of Integer Programming 1958-2008: From the Early Years to the State-of-the-Art*, pages 29–47. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.

[22] G. H. Laurens van der Maaten. Visualizing Data using t-SNE. *Journal of machine learning research*, 9:2579 2605, 2008.

[23] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *proc. OF THE IEEE*, 1998.

[24] B. D. Lehmann, B. Jovanović, X. Chen, M. V. Estrada, K. N. Johnson, Y. Shyr, H. L. Moses, M. E. Sanders, and J. A. Pietenpol. Refinement of Triple-Negative Breast Cancer Molecular Subtypes: Implications for Neoadjuvant Chemotherapy Selection. *PloS one*, 11(6):e0157368, 2016.

[25] C. A. Livasy, G. Karaca, R. Nanda, M. S. Tretiakova, O. I. Olopade, D. T. Moore, and C. M. Perou. Phenotypic evaluation of the basal-like subtype of invasive breast carcinoma. *Modern Pathology*, 19(2):264–271, 2006.

[26] M. Macenko, M. Niethammer, J. S. Marron, D. Borland, J. T. Woosley, X. Guan, C. Schmitt, and N. E. Thomas. A method for normalizing histology slides for quantitative analysis. *Proceedings - 2009 IEEE International Symposium on Biomedical Imaging: From Nano to Macro, ISBI 2009*, pages 1107–1110, 2009.

[27] E. Min, X. Guo, Q. Liu, G. Zhang, J. Cui, and J. Long. A Survey of Clustering with Deep Learning: From the Perspective of Network Architecture. *IEEE Access*, 6:39501–39514, 2018.

[28] H. Muhammad, C. S. Sigel, G. Campanella, T. Boerner, L. M. Pak, S. Büttner, J. N. M. IJzermans, B. G. Koerkamp, M. Doukas, W. R. Jarnagin, A. L. Simpson, and T. J. Fuchs. Unsupervised Subtyping of Cholangiocarcinoma Using a Deep Clustering Convolutional Autoencoder. In D. Shen, T. Liu, T. M. Peters, L. H. Staib, C. Essert, S. Zhou, P.-T. Yap, and A. Khan, editors, *Medical Image Computing and Computer Assisted Intervention – MICCAI 2019*, pages 604–612, Cham, 2019. Springer International Publishing.

[29] P. Naylor, J. Boyd, M. Lae, F. Reyal, and T. Walter. Predicting residual cancer burden in a triple negative breast cancer cohort. *Proceedings - International Symposium on Biomedical Imaging*, 2019-April(Isbi):933–937, 2019.

[30] P. Peduzzi, J. Concato, A. R. Feinstein, and T. R. Holford. Importance of events per independent variable in proportional hazards regression analysis II. *Journal of Clinical Epidemiology*, 48(12):1503–1510, 1995.

[31] C. M. Perou, T. Sørile, M. B. Eisen, M. Van De Rijn, S. S. Jeffrey, C. A. Ress, J. R. Pollack, D. T. Ross, H. Johnsen, L. A. Akslen, Fluge, A. Pergammenschlkov, C. Williams, S. X. Zhu, P. E. Lønning, A. L. Børresen-Dale, P. O. Brown, and D. Botstein. Molecular portraits of human breast tumours. *Nature*, 406(6797):747–752, 2000.

[32] K. Polyak. Review series introduction Heterogeneity in breast cancer. *J.Clin.Invest.*, 121(10):2011–2013, 2011.

[33] C. Song, F. Liu, Y. Huang, L. Wang, and T. Tan. Auto-encoder based data clustering. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 8258 LNCS(PART 1):117–124, 2013.

[34] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi. Inception-v4, inception-ResNet and the impact of residual connections on learning. *31st AAAI Conference on Artificial Intelligence, AAAI 2017*, pages 4278–4284, 2017.

[35] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the Inception Architecture for Computer Vision. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016-Decem:2818–2826, 2016.

[36] D. Tellez, G. Litjens, P. Bándi, W. Bulten, J. M. Bokhorst, F. Ciompi, and J. van der Laak. Quantifying the effects of data augmentation and stain color

normalization in convolutional neural networks for computational pathology. *Medical Image Analysis*, 58, 2019.

[37] A. Vahadane, T. Peng, A. Sethi, S. Albarqouni, L. Wang, M. Baust, K. Steiger, A. M. Schlitter, I. Esposito, and N. Navab. Structure-Preserving Color Normalization and Sparse Stain Separation for Histological Images. *IEEE Transactions on Medical Imaging*, 35(8):1962–1971, 2016.

[38] E. Vittinghoff and C. E. McCulloch. Relaxing the rule of ten events per variable in logistic and cox regression. *American Journal of Epidemiology*, 165(6):710–718, 2007.

[39] J. Xie, R. Girshick, and A. Farhadi. Unsupervised deep embedding for clustering analysis. *33rd International Conference on Machine Learning, ICML 2016*, 1:740–749, 2016.

[40] J. Xie, R. Liu, J. Luttrell, and C. Zhang. Deep learning based analysis of histopathological images of breast cancer. *Frontiers in Genetics*, 10(FEB):1–19, 2019.

[41] B. S. Yadav, P. Chanana, and S. Jhamb. Biomarkers in triple negative breast cancer: A review. *World Journal of Clinical Oncology*, 6(6):252–263, 2015.

[42] B.-N. Zhang, X.-C. Cao, J.-Y. Chen, J. Chen, L. Fu, X.-C. Hu, Z.-F. Jiang, H.-Y. Li, N. Liao, D.-G. Liu, O. Tao, Z.-M. Shao, Q. Sun, S. Wang, Y.-S. Wang, B.-H. Xu, and J. Zhang. Guidelines on the diagnosis and treatment of breast cancer (2011 edition). *Gland surgery*, 1(1):39–61, 2012.

[43] Y. Zhang, W. Nock, M. Wyse, Z. Weber, E. Adams, S. Stockard, D. Tallman, E. P. Winer, N. U. Lin, M. Cherian, S. Asad, S. Stockard, D. Tallman, E. P. Winer, N. U. Lin, M. Cherian, M. B. Lustberg, B. Ramaswamy, S. Sardesai, J. VanDeusen, N. Williams, R. Wesolowski, and D. G. Stover. Machine learning predicts rapid relapse of triple negative breast cancer. *bioRxiv*, 2019.

Appendix A

A.1 Vary γ

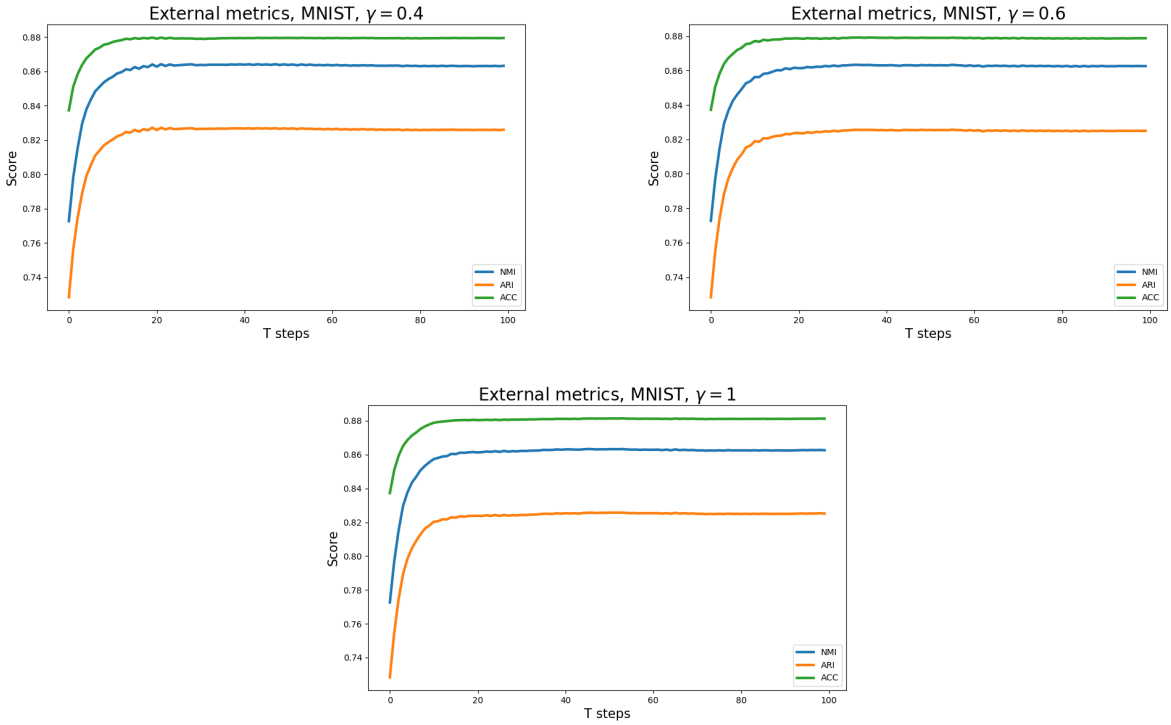


Figure A.1: MNIST external metrics as a function of T for $\gamma = 0.4$ (top left), $\gamma = 0.6$ (top left) and $\gamma = 1$ (bottom).

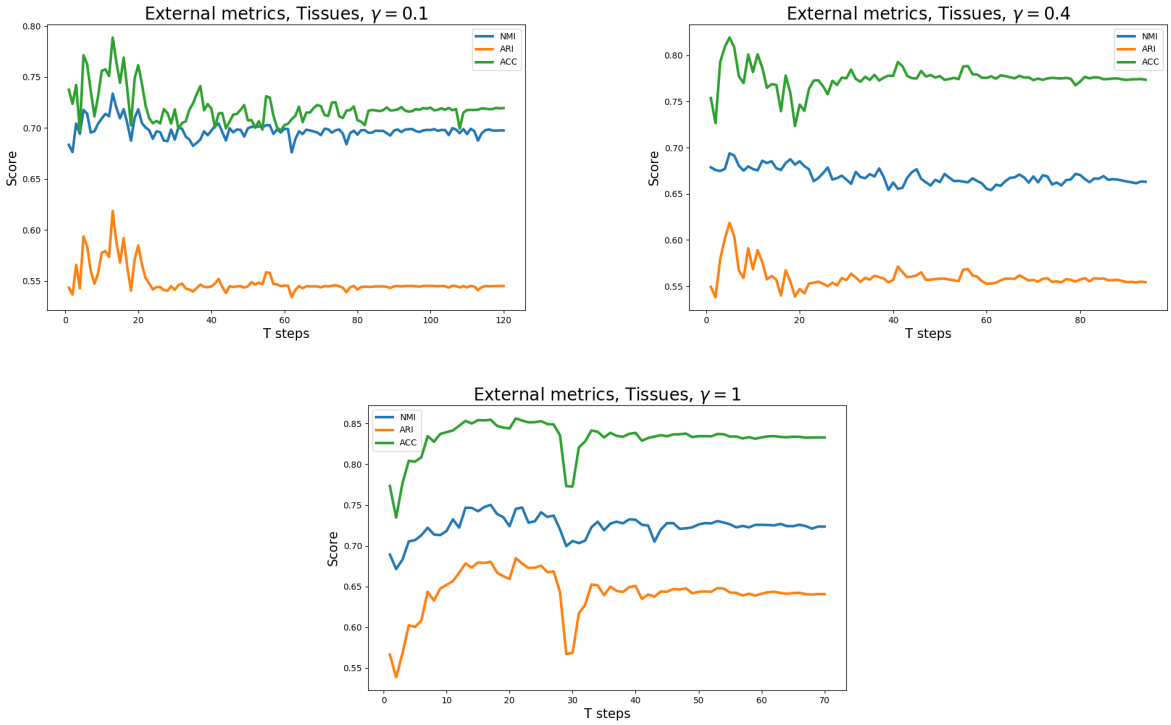


Figure A.2: Tissues external metrics as a function of T for $\gamma = 0.1$ (top left), $\gamma = 0.4$ (top left) and $\gamma = 1$ (bottom).

A.2 Vary K

MNIST

Figure A.3: K=5 clusters for with 10 samples each MNIST.

	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
0	1	28	17	33	5706	28	5	5959	41	5697
1	1	6574	28	251	62	1165	117	213	2574	52
2	5912	1	9	6	61	25	5742	3	18	20
3	6	31	22	5786	2	4197	40	17	3191	176
4	3	108	5882	55	11	6	14	73	27	4

Table A.1: MNIST confusion matrix for K=5. Each row represents a cluster to the 10 class labels in columns.

Figure A.4: K=8 clusters for MNIST.

	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
0	3	2	5	18	6	3509	128	7	5050	18
1	15	1	1	0	12	15	5742	0	5	1
2	7	13	37	6038	2	1867	21	9	726	166
3	0	26	19	12	52	3	0	5594	17	1597
4	5893	1	5	1	12	6	15	2	7	15
5	1	128	5877	45	7	4	3	65	14	1
6	3	29	9	16	5749	16	5	574	25	4148
7	1	6545	5	1	2	1	4	14	7	3

Table A.2: MNIST confusion matrix for K=8. Each row represents a cluster to the 10 class labels in columns.

Figure A.5: K=10 clusters for MNIST. Each row represents a cluster.

	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
0	1	25	2	2	2543	6	1	51	29	2764
1	0	16	1	3	3248	6	3	91	25	2910
2	1	6533	5	0	1	0	2	10	5	2
3	3	12	17	5973	0	278	0	3	10	95
4	7	0	2	0	5	21	5703	0	4	0
5	5888	1	3	1	7	5	17	3	6	14
6	19	14	23	60	30	35	21	17	5727	76
7	2	121	5883	52	3	4	2	39	9	1
8	0	20	22	24	5	0	0	6051	2	77
9	2	0	0	16	0	5066	169	0	34	10

Table A.3: MNIST confusion matrix for K=10. Each row represents a cluster to the 10 class labels in columns.

Figure A.6: K=16 clusters for MNIST.

	0	1	2	3	4	5	6	7	8	9
0	4	1	2	0	2	21	3199	0	5	0
1	1	2876	9	0	2	0	1	8	5	3
2	1	8	16	5938	0	40	0	1	8	122
3	0	14	20	12	5	0	0	3040	1	44
4	0	3	3	4	2420	6	4	42	14	2248
5	9	0	2	0	1	1	2664	0	1	0
6	0	11	26	24	3	0	0	3076	2	39
7	0	10	2	0	3342	0	1	2	4	15
8	0	2	0	40	0	2966	6	1	4	7
9	1	1	3363	31	0	7	0	17	9	1
10	5894	1	4	2	11	4	7	2	6	16
11	1	0	0	3	0	2356	16	0	15	3
12	0	120	2500	37	2	0	2	17	2	0
13	0	3673	1	0	1	0	0	8	4	1
14	10	5	10	36	12	14	18	2	5701	32
15	2	17	0	4	41	6	0	49	70	3418

Table A.4: MNIST confusion matrix for K=16. Each row represents a cluster to the 10 class labels in columns.

Tissues

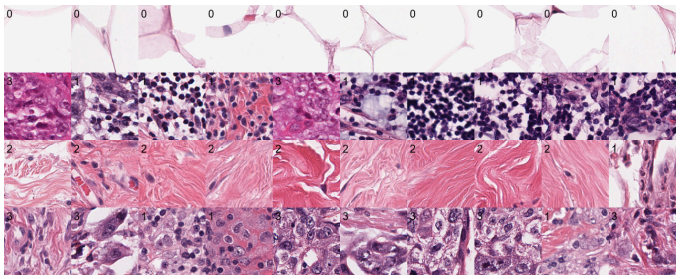


Figure A.7: K=4 clusters for Tissues.

	Fat	Lymphocytes	Stroma	Tumor
0	1562	0	2	0
1	0	835	0	337
2	2	30	1566	3
3	0	670	0	1224

Table A.5: Tissues confusion matrix for K=4. Each row represents a cluster to the 4 labels.

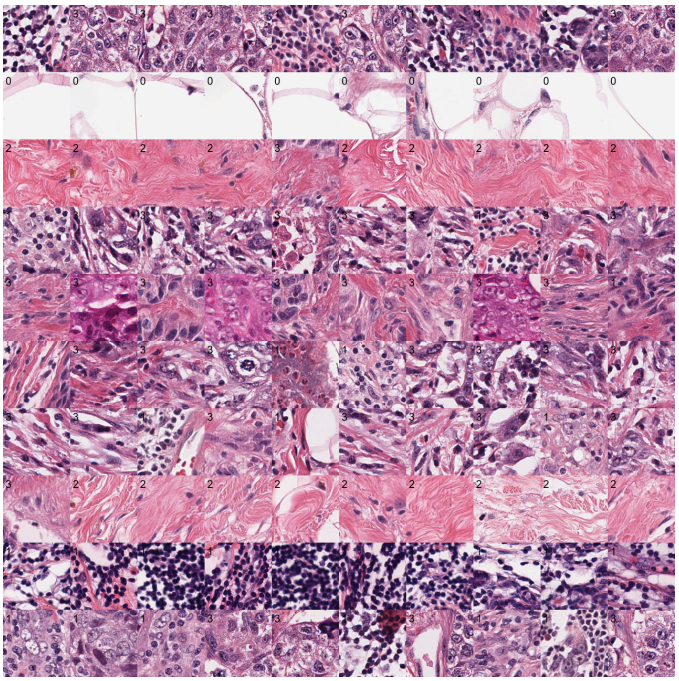


Figure A.8: K=12 clusters for Tissues. Notice that although 12 clusters are initialized, only 10 return with labels.

	Fat	Lymphocytes	Stroma	Tumor
0	0	664	0	708
1	1562	0	0	0
2	0	10	615	76
3	0	180	0	843
4	0	84	0	822
5	0	33	0	63
6	0	224	0	551
7	2	8	953	8
8	0	231	0	0
9	0	101	0	109

Table A.6: Tissues confusion matrix for K=12. Each row represents a cluster to the 4 labels. Notice only 10 rows because only 10 clusters were assigned although initializing with K=12.

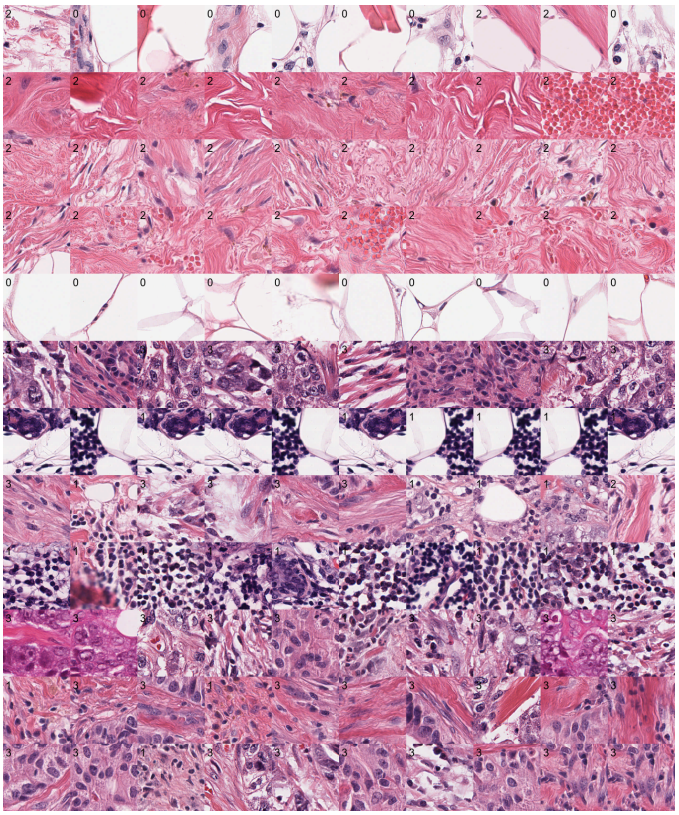


Figure A.9: K=20 clusters for Tissues. 12 clusters are assigned.

	Fat	Lymphocytes	Stroma	Tumor
0	30	0	4	0
1	0	2	413	9
2	2	2	662	6
3	0	1	483	4
4	1532	0	0	0
5	0	354	0	1729
6	0	2	0	0
7	0	111	4	137
8	0	652	0	0
9	0	206	0	890
10	0	54	2	126
11	0	151	0	279

Table A.7: Tissues confusion matrix for K=20. Each row represents a cluster to the 4 labels. Notice only 12 rows because only 12 clusters were assigned although initializing with K=20.

A.3 Vary update interval T

A.3.1 Tissues

This experiment shows the results of varying update interval T for 100 epochs with $\delta = 0$. In these experiments $T = \{140, 100, 80, 60, 40\}$. Table A.8 shows the experiments.

T	NMI	ARI	ACC	time	updates/epoch
140	0.69	0.54	0.71	2h27m	1
100	0.70	0.57	0.77	3h11m	2
80	0.69	0.56	0.77	3h57m	3
60	0.70	0.57	0.76	4h40m	4
40	0.69	0.56	0.76	6h21m	6

Table A.8: T update intervals show that external metrics are not very sensitive to T. Time consumption is a more important factor here.

A.4 Internal metrics

A.4.1 ARI

This shows an example of how to calculate ARI. ARI is defined by:

$$ARI = \frac{\sum_{ij}\binom{n_{ij}}{2} - \left(\sum_i \binom{a_i}{2}\sum_j \binom{b_j}{2}\right)/\binom{n}{2}}{\frac{1}{2}\left(\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2}\right) - \left(\sum_i \binom{a_i}{2}\sum_j \binom{b_j}{2}\right)/\binom{n}{2}} \tag{A.1}$$

where i refers to the row index number, j columns count, $\sum_{ij}\binom{n_{ij}}{2}$ is the sum of binomial counts at the ij^{th} entry, $\sum_i \binom{a_i}{2}$ is the row sums and $\sum_j \binom{b_j}{2}$ is the column sums.

With a simple example imagine the confusion matrix between labels $Y = Y_1, Y_2, Y_3$ and clusters $X = X_1, X_2, X_3$:

	Y_1	Y_2	Y_3	row sum
X_1	3	0	1	4
X_2	1	2	1	4
X_3	0	2	2	4
col sum	4	4	4	n=12

from eq. A.1 the terms $\sum_{ij}\binom{n_{ij}}{2}$, $\sum_i \binom{a_i}{2}$ and $\sum_j \binom{b_j}{2}$ are calculated:

$$\sum_i \binom{a_i}{2} = \binom{4}{2} + \binom{4}{2} + \binom{4}{2} = 6 + 6 + 6 = 18$$

$$\sum_j \binom{b_j}{2} = \binom{4}{2} + \binom{4}{2} + \binom{4}{2} = 6 + 6 + 6 = 18$$

which gives the final ARI:

$$ARI = \frac{6 - (18 \cdot 18)/\binom{12}{2}}{\frac{1}{2}(18+18) - (18 \cdot 18)/\binom{12}{2}} = \frac{6 - 4.91}{18 - 4.91} = 0.083 \tag{A.2}$$

A.5 Log rank hypothesis example

This example shows how the log rank hypothesis is calculated. The example is seen in http://sphweb.bumc.bu.edu/otlt/MPH-Modules/BS/BS704_Survival/BS704_Survival5.html. The null hypothesis is:

$$\begin{aligned} H_0 &: \ S_1(t) = S_2(t) \\ H_1 &: \ S_1(t) \neq S_2(t) \end{aligned} \tag{A.3}$$

χ^2_{obs} is calculated by:

$$\chi^2_{obs} = \sum_{j=1}^{2} \frac{(\sum_{j=1}^{2} O_{j,t} - \sum_{j=1}^{2} E_{j,t})^2}{\sum_{j=1}^{2} E_{j,t}} \tag{A.4}$$

where $\sum_{j=1}^{2} O_{j,t}$ is the sum of observed events for group j and $\sum_{j=1}^{2} E_{j,t}$ is the expected number of events in group j over time. Events occur at different times. For j=1,2 groups, the expected number of events is:

$$E_{j=1,t} = N_{1,t} \frac{O_{tot}}{N_{tot}} \tag{A.5}$$

$$E_{j=2,t} = N_{2,t} \frac{O_{tot}}{N_{tot}} \tag{A.6}$$

$N_{tot} = N_{1,t} + N_{2,t}$ is the total number of patients at risk and $O_{tot} = O_{1,t} + O_{2,t}$ is the number of observed events at each event time. Table A.9 shows the log rank data to calculate χ^2_{obs}.

Time	$N_{1,t}$	$N_{2,t}$	N_{tot}	$O_{1,t}$	$O_{2,t}$	O_{tot}	$E_{1,t}$	$E_{2,t}$
8	10	10	20	1	0	1	0.500	0.500
12	8	10	18	1	0	1	0.444	0.556
14	7	10	17	1	0	1	0.412	0.588
21	5	10	15	1	0	1	0.333	0.667
26	4	8	12	1	0	1	0.333	0.667
27	3	8	11	1	0	1	0.273	0.727
28	2	8	10	0	1	1	0.200	0.800
33	1	7	8	0	1	1	0.125	0.875
41	0	5	5	0	1	1	0.000	1.000
				6	**3**		**2.620**	**6.380**

Table A.9: Log rank data example with risk ($N_{1,t}, N_{2,t}, N_{tot}$), event ($O_{1,t}, O_{2,t}, O_{tot}$) and expected ($E_{1,t}, E_{2,t}$) counts.

The test statistic χ^2_{obs} is now calculated:

$$\begin{aligned} \chi^2_{obs} &= \sum_{j=1}^{2} \frac{(\sum_{j=1}^{2} O_{j,t} - \sum_{j=1}^{2} E_{j,t})^2}{\sum_{j=1}^{2} E_{j,t}} \\ &= \frac{(6 - 2.620)^2}{2.620} + \frac{(3 - 6.380)^2}{6.380} = 4.360 + 1.791 = 6.151 \end{aligned} \tag{A.7}$$

The critical value for 1 degree of freedom is $\chi^2_{crit} = 3.84$. Because $\chi_{obs} > \chi^2_{crit} = 3.84$ the null hypothesis is rejected and the survival curves are significantly different.