Politechnika Łódzka
Instytut Matematyki

CLUSTERING PROJECT - CAPSTONE MODULE

Dimensionality Reduction and Clustering of High-Dimensional Biological Data using UMAP

Faculty of Technical Physics, Computer Science and Applied Mathematics
Supervisor: dr Grzegorz Skalski
Students: Igor Hryniewicz; Bartosz Kamiński; Hubert Wołodkiewicz
Student no.: 247055; 247060; 247077
Field of study: Mathematical Methods in Data Analysis

Łódź, 1.04.2025

Instytut Matematyki
90-924 Łódź, ul. Wólczańska 215, budynek B9
tel. 042 631 27 97, 042 632 97 57, fax 042 630 34 14 email: office@ics.p.lodz.pl

Abstract

This project utilizes data from *The Cancer Genome Atlas (TCGA)* [1], specifically focusing on Kidney Renal Clear Cell Carcinoma (KIRC)—the most common and aggressive subtype of kidney cancer. The dataset, sourced from the University of North Carolina TCGA Genome Characterization Center, includes gene expression profiles obtained through the Illumina HiSeq 2000 RNA Sequencing platform. Each sample is uniquely identified by a Sample ID. The primary objective of this project is to analyse the genomic patterns in the KIRC dataset through clustering techniques to identify anomalies and uncover relationships within the data. By applying dimensionality reduction methods and clustering algorithms based on relevant metrics, this project aims to detect hidden structures, group similar samples, and potentially highlight outliers that may provide insights into disease subtypes or variations. The data was log-normalized to standardize gene expression values across 20,531 columns. To simplify analysis and visualization, we used Uniform Manifold Approximation and Projection (UMAP) to reduce the dataset to three dimensions, enabling better interpretation of its structure. We performed DBSCAN to identify global clustering patterns and Hierarchical Clustering for local patterns.

Contents

Chapter 1

Exploratory Data Analysis

1.1 Dataset Description

The kidney cancer dataset consists of 606 rows, each representing a unique sample, and 20,531 columns corresponding to gene expression levels. All data points are numeric. Originally, the dataset underwent log-normalisation using $\log_2(x+1)$, but this transformation was later removed and replaced with z-normalisation.

1.2 First Insight

The dataset did not contain any missing values, so imputation techniques were unnecessary. Upon examining the population's mean and standard deviation, we confirmed that it was z-normalised. Additionally, an inspection of the column data types revealed that 20,530 columns were of floating-point type, while 1 column was of object type, representing the genome set.

Table 1.1: Z-normalized gene expression data (first 5 columns) for kidney cancer

Sample Id	ARHGEF10L	HIF3A	RNF17	RNF10
TCGA-B0-5402-01	-0.613869	3.907610	-0.092480	-0.428222
TCGA-CJ-4634-01	1.150781	0.451041	-0.092480	-0.582534
TCGA-B0-4828-01	0.450394	0.096415	-0.061711	0.075810
TCGA-CZ-5452-11	1.397206	0.478936	-0.092480	2.427314
TCGA-B4-5835-01	-1.127536	-0.586795	-0.092480	-0.651031

1.3 Cleaning & Preprocessing

In terms of data cleaning, no actions were required as there were no missing values. Removing any data could have been detrimental, as outliers may provide valuable insights.

In terms of preprocessing, we decided to apply $\ln(X + |\min X| + 1)$ function across all of our data to stabilize variance and handle small or zero values, ensuring the data is positive and more suitable for analysis. This transformation reduces skewness and enhances sensitivity to small, significant variations in medical data, which is a common practise.

Table 1.2: Gene expression data (first 5 columns) for kidney cancer after transformation

Sample Id	ARHGEF10L	HIF3A	RNF17	RNF10
TCGA-B0-5402-01	1.745665	2.327394	1.832758	1.777552
TCGA-CJ-4634-01	2.014151	1.916132	1.832758	1.751119
TCGA-B0-4828-01	1.916036	1.862528	1.837668	1.859323
TCGA-CZ-5452-11	2.046503	1.920228	1.832758	2.171439
TCGA-B4-5835-01	1.651740	1.750379	1.832758	1.739158

Chapter 2

Dimensionality Reduction

2.1 UMAP

2.1.1 Idea behind *UMAP*

UMAP (Uniform Manifold Approximation and Projection) [2] is a dimensionality reduction method, which heavily realies on algebraic topology and category theory. It gives similarly good outputs for visualisation as t-SNE, with a substantially better runtime, and may capture more of the global structure of the data.

It assumes that dataset D is uniformly distributed on a Riemannian manifold M, that is then embedded into $\mathbb{R}^N$, for some $N \in \mathbb{N}$. We also assume, that M is locally connected (so that there are no isolated points in D). Finally, we assume that the metric on M is locally constant, as this property will allow us to use lemma 2.1.1, that approximates distances in M between points in D that are close in $\mathbb{R}^N$.

The problem that appears when using lemma 2.1.1 is that we will get different metrics d_{x_i} depending on the choice of $x_i \in D$. To tackle this phenomenon, we turn to the language of *Category Theory, Simplicial Complexes* and especially *Fuzzy Simplicial Complexes*. After acquiring global metric, we focus on optimising the projection onto $\mathbb{R}^m$ for $m < N$, using *Cross Entropy*.

2.1.2 Underlying Manifold Approximation

Let $D = \{x_1, x_2, \ldots, x_k\}$ be a set of datapoints in $\mathbb{R}^N$. Assume there exists a Riemannian Manifold M such that $D \subset M$. We will use the following lemma to

show that we can change the metric on M, such that points in D are uniformly distributed.

Lemma 2.1.1. *Let (M, g) be a Riemannian manifold embedded in $\mathbb{R}^n$. Let $p \in M$ be a point. Suppose that g is locally constant. Let B be a ball in M, containing p, whose volume is $\frac{\pi^{\frac{n}{2}}}{\Gamma\left(\frac{n}{2}+1\right)}$ with respect to the metric on M. Then the distance of the shortest path in M from p to a point $q \in B$ is $\frac{1}{r} d_{\mathbb{R}^n}(p, q)$, where r is the radius of B in $\mathbb{R}^n$ and $d_{\mathbb{R}^n}(p, q)$ is the distance from p to q in $\mathbb{R}^n$*

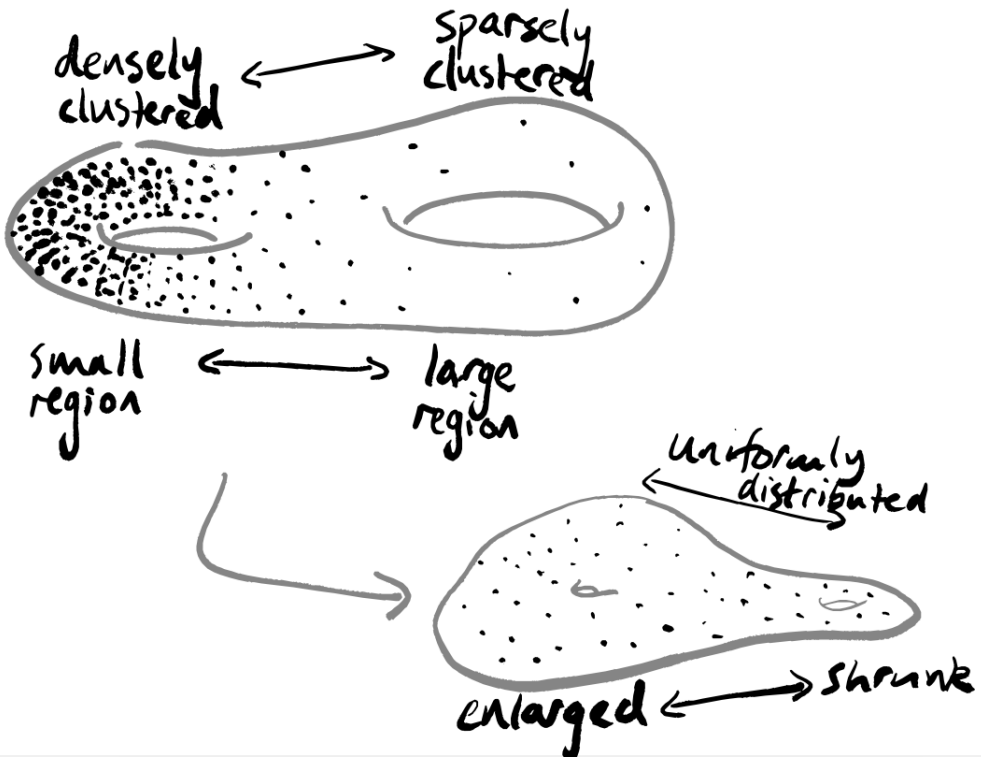


Figure 2.1: Forcing some non-uniform distribution of points on a manifold to make it uniform. [3]

Thanks to the lemma 2.1.1 we can approximate distances around M close to the datapoints by scaling the distance in $\mathbb{R}^N$. Having the assumption that data is uniformly distributed on M, any ball of a fixed volume R on M should have the same number of datapoints. Let us denote a ball $N_k(x)$ in $\mathbb{R}^N$ around a datapoint x that contains k nearest neighbours of x in $\mathbb{R}^N$.

For any datapoint x_i, let us consider the neighbourhood of M that is sent to $N_k(x_i)$ in $\mathbb{R}^N$. This ball should have the same volume, even if we repeat the procedure for any other datapoint x_j. From the lemma we have, that for k small enough,

we are able to approximate distances in M from datapoint x_i to one of its k nearest neighbours x_j. Fix k as a hyperparameter, and write $\{x_{i_1}, x_{i_2}, \ldots, x_{i_k}\}$ for the k nearest neighbours of x_i.

Also, by the lemma, we can take that the distance in M from x_i to x_j is approximately $\frac{1}{r_i} d_{\mathbb{R}^N}(x_i, x_j)$, where r_i is the distance to the k^{th} nearest neighbour of x_i. We can avoid the situation, where k^{th} nearest neighbour is a lot farther then the $(k-1)^{th}$ nearest neighbours clustered close to x_i, by taking such r_i that

$$\sum_{j=1}^{k} \exp\left(\frac{-|x_i - x_{i_j}|}{r_i}\right) = log_2(k).$$

2.1.3 Category Theory

This subsection is for the introductory notions of the category theory. It's language will be crucial to the problem of globalization of the metrics d_{x_i} for $x_i \in D$.

Definition 2.1.2 (Category)**.** *A* category $\mathcal{C}$ *is a collection of objects, denoted as* $Obj(\mathcal{C})$*, and between any two objects* $X, Y \in Obj(\mathcal{C})$ *a collection of morphisms denoted as* $Hom_{\mathcal{C}}(X, Y)$*.*

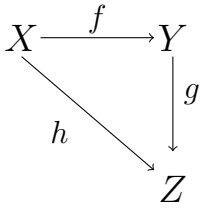
For easier notation we write $f\colon X \to Y$*, or* $X \xrightarrow{f} Y$ *to indicate that* $f \in Hom_{\mathcal{C}}(X, Y)$*.*

Morphisms have to satisfy the following conditions:

1. $\forall(X, Y, Z \in Obj(\mathcal{C}))((\exists(X \xrightarrow{f} Y \wedge Y \xrightarrow{g} Z)) \implies \exists(X \xrightarrow{h} Z)\ g \circ f = h)$,
2. $\forall X \in Obj(\mathcal{C})\ \exists X \xrightarrow{id_X} X$,
3. $\forall(X, Y \in Obj(\mathcal{C}))\forall(X \xrightarrow{f} Y) f \circ id_Y = id_X \circ f = f$,

where $\circ$ *is morphism composition - a primitive notion of category theory.*

Visually 1^{st} condition can be represented as the following diagram commuting.



We know of many examples of categories, but it may be beneficial to give some of them, for the better picture. Category **Set**, where objects are sets from set

theory, and morphisms are functions between sets. Category **Top**, where objects are topological spaces, and morphisms between objects are continuous functions.

Definition 2.1.3 (Opposite category)**.** *For a category $\mathcal{C}$ let $\mathcal{C}^{op}$ denote an* opposite category*, where $Obj(\mathcal{C}) = Obj(\mathcal{C}^{op})$, and morphisms are defined by $Hom_{\mathcal{C}^{op}}(X, Y) \coloneqq Hom_{\mathcal{C}}(Y, X)$.*

Intuitively it means 'reversing the arrows' - $X \xrightarrow{f} Y$ becomes $X \xleftarrow{f^{op}} Y$.

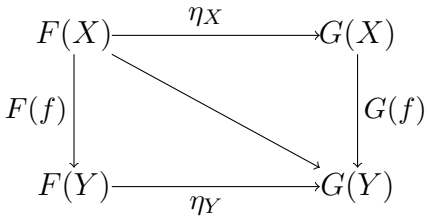
Now we can define maps between categories.

Definition 2.1.4 (Functor)**.** *Let $\mathcal{C}, \mathcal{D}$ be categories. A* functor $F\colon \mathcal{C} \to \mathcal{D}$ *maps objects $X \in \mathcal{C}$ to objects $F(X) \in \mathcal{D}$, and morphisms $f \in Hom_{\mathcal{C}}(X, Y)$ to morphisms $F(f) \in Hom_{\mathcal{D}}(F(X), F(Y))$, such that id_X becomes $id_{F(X)}$, and composition is preserved i.e. $F(g \circ f) = F(g) \circ F(f)$.*

Of course there always exists the identity functor $id_{\mathcal{C}}$ for every category $\mathcal{C}$.

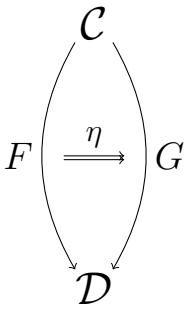
Having a notion of changing category using functors, we can compare *how* they change.

Definition 2.1.5. *Let $\mathcal{C}$, $\mathcal{D}$ be categories and $F, G\colon \mathcal{C} \to \mathcal{D}$ be functors between these categories. A* natural transformation $\eta\colon F \to G$ *is a collection of morphisms $\eta_X\colon F(X) \to G(X)$ for $X \in Obj(\mathcal{C})$, such that the following diagram commutes.*



Which can be written as $\eta_Y \circ F(f) = \eta_X \circ G(f)$.

We can also visualize the notion of natural transformation as follows:



The last definition is a weaker notion of equivalence between categories. 'Normal' equivalence of categories is that there exist functors $F\colon \mathcal{C} \to \mathcal{D}$ and $G\colon \mathcal{D} \to \mathcal{C}$ such that $F \circ G = id_{\mathcal{D}}$ and $G \circ F = id_{\mathcal{C}}$. The *adjunction* is a weaker condition.

Definition 2.1.6 (Adjunction)**.** *We say that two functors $F\colon \mathcal{C} \to \mathcal{D}$ and $G\colon \mathcal{D} \to \mathcal{C}$ form an* adjunction *with F as* left adjoint *and G as* right adjoint*, denoted as $F \dashv G$, if there are natural transformations $\eta\colon id_{\mathcal{C}} \to G \circ F$ and $\varepsilon\colon F \circ G \to id_{\mathcal{D}}$.*

This relationship is not symmetric, but it gives a satisfying notion of 'being the same'.

Having done and being familiar with these ideas, we can use this language to our advantage.

2.1.4 Fuzzy Simplicial Sets

A simplicial complex describes a topological space (space with associated data consisting of all open sets in the space. Manifolds and metric spaces are topological spaces with some more structure on them.) in a combinatorial way.

Definition 2.1.7. *A* geometric n-simplex *is the convex hull spanned by a set of $n+1$ linearly independent vertices $\{x_0, x_1, \ldots, x_n\}$ in Euclidean space. That is,*

$$\left\{ \sum_{i=0}^{n} t_i x_i \ : \ t_i \geqslant 0, \sum_{i=0}^{n} t_i = 1 \right\}.$$

Note that the convex hull spanned by a n-vertex subset of the x_i is itself an $(n-1)$-dimensional simplex. We call this a *face* of the n-simplex.

Remark 2.1.8. *For any $n \in \mathbb{N}$, any two geometric n-simplices T and U are* homeomorphic *i.e. there exists a continuous bijection $f\colon T \to U$ with a continuous inverse.*

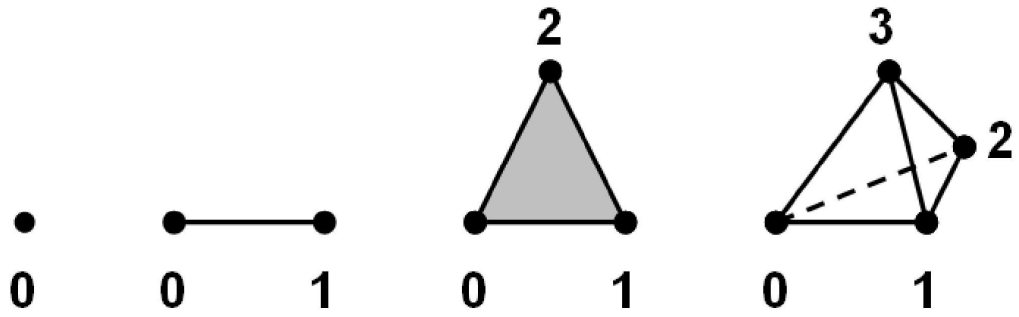


Figure 2.2: From the left: 0-simplex (single point), 1-simplex (interval), 2-simplex (triangle), 3-simplex (tetrahedron).

Manifolds can be decomposed into simplices.

Definition 2.1.9 (Geometric simplicial complex)**.** *A* geometric simplicial complex X *is a collection of simplices in* $\mathbb{R}^N$ *such that for any simplex* $T \in X$ *and any face* $U \subset T$, $U \in X$*; and if* $V, G \in X$*, then either* $V \cap G$ *is empty, or is their mutual face.*

Note that, up to homeomorphism, we can describe X by listing the vertices of each simplex. Vertices which appear in more than one simplex, allow us then to identify common faces.

We can generalise this idea further.

Definition 2.1.10 (Abstract simplicial complex)**.** *An* abstract simplicial complex X *is a sequence of sets* $(X^i)_{i\geq 1}$ *such that the elements of* X^n *are* $(n+1)$*-element sets which satisfy for any* $\{x_i\}_{i=0}^n \in X^n$*, any* n*-element subset of this set is in* X^{n-1}*.*

To abstract this idea even further let all the vertices of an abstract simplex come with an ordering. We can then write an ordered n-simplex as $[x_0, x_1, \ldots, x_n]$. Note that every simplex can be characterized by $n+1$ face maps, where the i^{th} face map is obtained by removing i^{th} vertex from the original simplex $[x_0, \ldots, x_{i-1}, x_{i+1} \ldots x_n]$. Let d_i be the i^{th} face map. Notice that for $i \leq j$ $d_i d_j = d_{j-1} d_i$.

Definition 2.1.11 (Delta complex)**.** *A* Delta complex X *is a collection of sets* X^i*, where for each* $n \geq 0$ *and* $0 \leq i \leq n$ *maps* $d_i \colon X^n \to X^{n-1}$ *satisfy the relation* $d_i d_j = d_{j-1} d_i$*, for all* $i \leq j$*.*

We interpret X^n as the n-simplices of a Delta complex. Unlike simplicial complexes, Delta complexes allow multiple simplices to share a vertex set.

Now we can define Delta complexes using the language of category theory.

Definition 2.1.12 (Categorical Delta complex)**.** *Let $\widehat{\Delta}$ be the category of finite ordered sets $[n] = \{0, 1, \ldots, n\}$ for $n \in \mathbb{N}$ with morphisms as strictly order-preserving maps $[m] \to [n]$, when $m \leqslant n$. Then a* Delta complex *is a functor $X : \widehat{\Delta}^{op} \to Set$.*

We can then translate between these two definitions of Delta complex as follows. The set $X([n])$ is the n -simplices of the Delta complex. A map $[n] \to [n-1]$, an opposite to an order-preserving injection is then a face map. We can write maps from $[m] \to [n]$ in $\widehat{\Delta}$ as compositions of maps $[k] \to [k+1]$, $m \leqslant k < n$. Then the images of the morphisms under X are compositions of face maps.

Now we will define a *categorical simplicial set*.

Definition 2.1.13 (Categorical simplicial complex)**.** *Let Δ be the category of finite ordered sets $[n]$, $n \in \mathbb{N}$ with order-preserving maps $[m] \to [n]$, $m \leqslant n$, as morphisms. A* simplicial set *is a functor $X : \Delta^{op} \to Set$.*

Unlike Delta complexes, simplicial sets allow degenerate simplices. For instance, X^2 may include a degenerate 2-simplex where a triangle collapses into an edge.

A *fuzzy set* generalizes a set, where instead of elements simply being in the set, there is a continuous membership function which indicates their membership. In set theory it is a set of objects A together with a function $\mu : A \to [0, 1]$, where $\mu(a) = 1$ indicates full membership.

Definition 2.1.14 (Fuzzy simplicial set)**.** *Let* Fuzz *be the category with fuzzy sets as objects, where morphisms $f : A \to B$ satisfy $f \circ \mu(a) \leq \mu(a)$. A* fuzzy simplicial set *is then a functor $X : \Delta^{\mathrm{op}} \to$ Fuzz.*

The given condition ensures that the elements taken from A to B can't end up with lower membership strength.

2.1.5 Switching between metric spaces and fuzzy simplicial sets

Fix $x_i \in D$. Having the manifold metric, it is possible to approximate the distance from x_i to some point $x_j \in D$ with $d'_{x_i}(x_i, x_j) = \frac{1}{r_i} d_{\mathbb{R}^N}(x_i, x_j)$. Thanks to this

approximation, we get a partly defined metric space, in which we know distances between x_i and x_j for all $j \in |D|$ but not the other way around, that is x_j and x_k for $j, k \neq i$.

Let ρ_i be the distance from x_i to the nearest neighbour in D in $\mathbb{R}^N$. Having the previous assumption about M being locally connected, we forced x_i to be connected to its nearest neighbour, for $j \neq i$, set that

$$d_{x_i}(x_i, x_j) = \frac{1}{r_i}(d_{\mathbb{R}^N}(x_i, x_j) - \rho_i)$$

Now, it means that the distance from x_i to its nearest neighbour is 0. For $j, k \neq i$, we do not really know the distances relative to x_i, between x_j and x_k, so we set them to ∞.

To wrap this up,

$$d_{x_i}(x_j, x_k) = \begin{cases} \frac{1}{r_i}\left(d_{\mathbb{R}^N}(x_j, x_k) - \rho_i\right) & \text{if } j = i \text{ or } k = i, \\ \infty & \text{otherwise.} \end{cases}$$

Because of it, we do not have a metric space anymore but instead the following generalisation.

Definition 2.1.15. *An* extended-pseudo-metric space *is a set X and a function $d\colon X \times X \to \mathbb{R} \cup \{\infty\}$ such that*

1. $d(x, y) \geq 0$,
2. $d(x, x) = 0$,
3. $d(x, y) = d(y, x)$,
4. $d(x, z) = \infty$ *or* $d(x, z) \leq d(x, y) + d(y, z)$

We should take it into account that it also allows infinite distances and $d(x, y) = 0$ when $x \neq y$.

Let **EPMet** represent the category of extended-pseudo-metric spaces where the morphisms are non-expansive mappings. Let **FinEPMet** denote the subcategory of **EPMet** whose objects are finite extended-pseudo-metric spaces. Note that every approximation of distances within D is an element of **FinEPMet**.

Let **sFuzz** denote the category of fuzzy simplicial sets (which are functors of the form $X\colon \Delta^{op} \to \mathbf{Fuzz}$) whose morphisms are natural transformations between these functors. Let **Fin-sFuzz** be the subcategory of **sFuzz** consisting of fuzzy simplicial sets with a finite number of non-degenerate simplices, defined as follows.

Definition 2.1.16. *Let X be a fuzzy simplicial set. An element of $X([n])$ is* degenerate *if its geometric realization is an $(n-1)$-simplex.*

The count of non-degenerate n-simplices of X is the number of non-degenerate elements of $X([n])$ with positive membership strength. Thus, X belongs to **Fin-sFuzz** if it has a finite number of non-degenerate n-simplices.

The main theorem provides a translation between these two categories.

Theorem 2.1.17. *There is an adjunction between* ***FinEPMet*** *and* ***Fin-sFuzz*** *given by FinSing:* ***FinEPMet*** $\to$ ***Fin-sFuzz*** *and FinReal :* ***Fin-sFuzz*** $\to$ ***FinEPMet****, with FinReal $\dashv$ FinSing.*

The functors in this theorem are as follows. The functor FinReal : **Fin-sFuzz** $\to$ **FinEPMet** takes a finite simplicial set X to the finite metric space T whose elements are the vertices of X. The metric on T is defined as follows. Let μ be the membership strength function for the simplices of X. For $x, y \in T$, let A be the set of all subsets of T containing both x and y. Then

$$d_T(x,y) = \min_{U \in A} -\log(\mu(U)).$$

The functor FinSing : **FinEPMet** $\to$ **Fin-sFuzz** takes a finite extended-pseudo-metric space Y to a finite fuzzy simplicial set, which is a functor from $\Delta^{op} \to \mathbf{Fuzz}$. It acts as follows: FinSing$(Y)([n])$ is the fuzzy set of $(n+1)$-vertex subsets of the datapoints $\{x_{k_0}, \ldots, x_{k_n}\}$, where the subset has membership strength

$$\mu(\{x_{k_0}, \ldots, x_{k_n}\}) = \min_{i,j} e^{-d(x_{k_i}, x_{k_j})}.$$

Given this translation, we can convert a set of datapoints D to a family of elements of **FinEPMet**, and from that to a family of finite fuzzy simplicial sets FinSing(D, d_{x_i}) for $x_i \in D$.

Now, denote the *fuzzy topological representation* of D to be

$$\bigcup_{i=1}^{n} \text{FinSing}(D, d_{x_i}),$$

where the union is some choice of a union of fuzzy sets.

We know that each of the $\text{FinSing}(D, d_{x_i})$ has the same set of objects, which are all simplices whose vertices are in D. Now, if (A, μ) and (A, ν) are two fuzzy sets with the same underlying set of objects, one reasonable definition of the union $(A, \mu) \cup (A, \nu)$ is $(A, (\mu \cup \nu))$, where $(\mu \cup \nu)(a) = \mu(a) \perp \nu(a)$ for $\perp$ some t-conorm.

The current implementation of UMAP uses $x \perp y = x + y - xy$, which is the obvious t-conorm to use if you interpret $\mu(a)$ and $\nu(a)$ as probabilities of the simplex a existing, assume these are independent between the different local metric spaces, and do not care about higher-dimensional simplices.

2.1.6 Going to lower-dimensions

We can now construct a fuzzy simplicial set from a given set of points in $\mathbb{R}^n$. Let the dataset D be in $\mathbb{R}^N$, and let E be a low-dimensional representation of D in $\mathbb{R}^m$, for $m < N$. To see how good of a representation E is we will compare the fuzzy simplicial set X constructed from D to the constructed from E, that we will denote as Y.

Note that with construction of Y we already know the metric of the underlying manifold as it is $\mathbb{R}^m$ itself. Thus $Y = FinSing((E, d))$ where d is the Euclidean metric on $\mathbb{R}^m$.

Consider the sets of edges in X and Y as fuzzy sets. Note that both of these have the same set of elements which differ only in the membership strength of the simplices. We define *cross-entropy* C of two fuzzy sets, (A, μ) and (A, ν) as follows:

$$C((A, \mu), (A, \nu)) = \sum_{a \in A} \left(\mu(a) \log \left(\frac{\mu(a)}{\nu(a)} \right) + (1 - \mu(a)) \log \left(\frac{1 - \mu(a)}{1 - \nu(a)} \right) \right).$$

Note that, for us, μ is fixed. In this formula $\mu(a) \log \left(\frac{\mu(a)}{\nu(a)} \right)$ provides the attractive force, as it gets small if short edges in D are also short in E, while $(1 - \mu(a)) \log \left(\frac{1-\mu(a)}{1-\nu(a)} \right)$ is the repulsive force as it is minimised if long edges in D correspond to short ones in E. We can then use gradient descent to optimise the embedding.

2.1.7 Results

After applying the previously mentioned transformations, we performed dimensionality reduction to three dimensions using UMAP to visualize the structure of our genome data. The application of UMAP revealed three clusters within the dataset that were further investigated in Chapter 3. This clustering effect is clearly visible in Figure 3.1, where the projected data points form separate and distinguishable groups.

Chapter 3

Clustering Algorithms

3.1 DBSCAN

3.1.1 Introduction

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) [4] is a widely used clustering algorithm that identifies clusters in data based on density. Unlike partition-based algorithms such as k-means, DBSCAN does not require the number of clusters to be predefined and can identify clusters of arbitrary shape, making it particularly suitable for complex datasets.

3.1.2 Key Concepts

DBSCAN is based on two main parameters:

— **Epsilon (ε):** The maximum distance between two points for one to be considered as in the neighborhood of the other.

— **MinPts:** The minimum number of points required to form a dense region.

The algorithm classifies points into three categories:

— **Core Points:** Points with at least MinPts points (including themselves) within their ε-neighborhood.

— **Border Points:** Points that are within the ε-neighborhood of a core point but do not have enough points in their own neighborhood to be a core point.

— **Noise Points:** Points that do not belong to any cluster.

3.1.3 Algorithm Description

DBSCAN operates as follows:

1. Arbitrarily select an unvisited point.
2. If the point is a core point, form a new cluster. Expand the cluster by recursively adding all points that are density-reachable.
3. Mark all noise points that are not density-reachable from any core point.
4. Repeat until all points are visited.

3.1.4 Mathematical Framework

Let $D = \{x_1, x_2, \ldots, x_n\}$ be a dataset in a metric space with a distance function d. Define the ε-neighborhood of a point x as:

$$N_\varepsilon(x) = \{y \in D \mid d(x, y) \leq \varepsilon\}.$$

A point x is a *core point* if:

$$|N_\varepsilon(x)| \geq \text{MinPts},$$

where $|\cdot|$ denotes the cardinality of a set.

A point y is *directly density-reachable* from x if:

$$x \text{ is a core point and } y \in N_\varepsilon(x).$$

A point z is *density-reachable* from x if there exists a sequence of points $x_1, x_2, \ldots, x_k$ such that:

$$x_1 = x,$$

$$x_k = z,$$

$$x_{i+1} \text{ is directly density-reachable from } x_i \text{ for all } i \in \{1, \ldots, k-1\}.$$

Points are clustered together if they are density-reachable from at least one core point.

3.2 HCA

3.2.1 Introduction

Agglomerative Clustering (also known as HCA) [5] is a hierarchical clustering algorithm that builds a hierarchy of clusters by successively merging smaller clusters. This bottom-up approach is widely used in various fields such as biology, social sciences, and image processing due to its interpretability and flexibility. Unlike partition-based methods, it does not require the number of clusters to be predefined.

3.2.2 Key Concepts

Agglomerative Clustering relies on a measure of similarity or distance between data points and clusters. The following components are essential:

— **Linkage Criteria:** Determines how the distance between clusters is computed. Common linkage methods include:

  – *Single Linkage:* Distance between the closest points of two clusters.

  – *Complete Linkage:* Distance between the farthest points of two clusters.

  – *Average Linkage:* Average distance between all pairs of points from two clusters.

  – *Ward's Method:* Minimizes the increase in total within-cluster variance after merging.

— **Dendrogram:** A tree-like diagram that illustrates the merging process and shows the hierarchy of clusters.

3.2.3 Algorithm Description

The algorithm follows these steps:

1. Initialize each data point as its own cluster.

2. Compute the pairwise distances between all clusters using the chosen linkage criteria.

3. Merge the two closest clusters to form a new cluster.

4. Update the distance matrix to reflect the distances between the new cluster and the remaining clusters.

5. Repeat steps 2–4 until all points belong to a single cluster or the desired number of clusters is reached.

3.2.4 Mathematical Framework

Let $D = \{x_1, x_2, \ldots, x_n\}$ be a dataset in a metric space with a distance function d. Initially, each data point x_i forms its own cluster $C_i = \{x_i\}$.

The distance between two clusters C_i and C_j is defined based on the linkage criteria:

— *Single Linkage:*

$$d(C_i, C_j) = \min_{x \in C_i, y \in C_j} d(x, y).$$

— *Complete Linkage:*

$$d(C_i, C_j) = \max_{x \in C_i, y \in C_j} d(x, y).$$

— *Average Linkage:*

$$d(C_i, C_j) = \frac{1}{|C_i| \cdot |C_j|} \sum_{x \in C_i} \sum_{y \in C_j} d(x, y).$$

— *Ward's Method:*

$$\Delta E = \sum_{x \in C_i \cup C_j} \|x - \mu_{ij}\|^2 - \sum_{x \in C_i} \|x - \mu_i\|^2 - \sum_{x \in C_j} \|x - \mu_j\|^2,$$

where μ_{ij} is the centroid of the merged cluster.

The process continues until a stopping criterion is met, such as reaching a single cluster or a predefined number of clusters.

3.3 Results

After dimensionality reduction, we applied the DBSCAN algorithm with parameters $\varepsilon = 1$ and minPts $= 3$. This resulted in the identification of three distinct clusters, as shown in Figure 3.1.

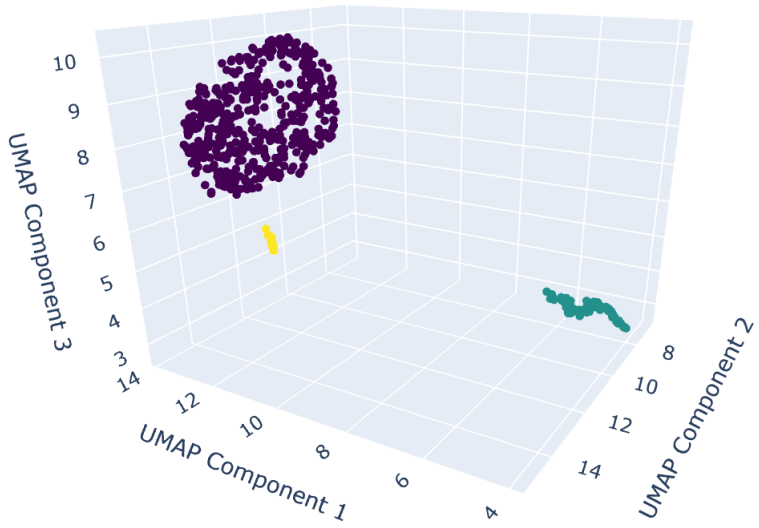


Figure 3.1: Clusters identified by DBSCAN in the 3-dimensional UMAP space.

Upon further examination, the medium-sized cluster exhibited characteristics suggesting potential substructure. To investigate this, we applied agglomerative clustering to this specific cluster. This method successfully partitioned the medium-sized cluster into three smaller, more refined clusters, revealing additional layers of structure in the data, as shown in Figure 3.2. The optimal number of clusters in the medium-sized group were determined by appropriate metrics shown in chapter 4.

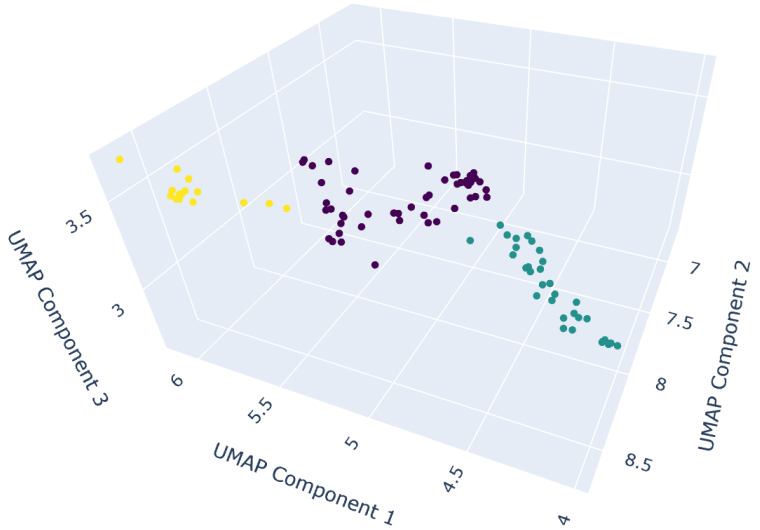


Figure 3.2: Clusters identified by agglomerative clustering in the medium-sized cluster.

Chapter 4

Metrics

4.1 Introduction

Evaluating the quality of clustering results is crucial for assessing the performance of clustering algorithms. Several metrics exist for this purpose, each capturing different aspects of the clustering structure. This document provides a mathematical and conceptual description of three commonly used metrics: the Silhouette Score, the Davies-Bouldin Index, and the Calinski-Harabasz Index.

4.2 Silhouette Score

The Silhouette Score measures the quality of a clustering by considering both cohesion (within-cluster similarity) and separation (between-cluster dissimilarity). For a given data point x_i, the Silhouette Score is defined as:

$$S(x_i) = \frac{b(x_i) - a(x_i)}{\max\{a(x_i), b(x_i)\}},$$

where:

— $a(x_i)$: The average distance between x_i and all other points in the same cluster (intra-cluster distance).

— $b(x_i)$: The average distance between x_i and points in the nearest cluster (inter-cluster distance).

The Silhouette Score for the entire dataset is the mean Silhouette Score of all points:

$$S = \frac{1}{n} \sum_{i=1}^{n} S(x_i),$$

where n is the number of data points. The score ranges from -1 to 1, where higher values indicate better clustering.

4.3 Davies-Bouldin Index

The Davies-Bouldin Index (DBI) evaluates clustering quality based on the average similarity between clusters. It is defined as:

$$DBI = \frac{1}{k} \sum_{i=1}^{k} \max_{j \neq i} R_{ij},$$

where:

— k: The number of clusters.

— $R_{ij} = \frac{S_i + S_j}{d(C_i, C_j)}$: The similarity between clusters C_i and C_j.

— S_i: The average intra-cluster distance for cluster C_i.

— $d(C_i, C_j)$: The distance between the centroids of clusters C_i and C_j.

Lower values of DBI indicate better clustering.

4.4 Calinski-Harabasz Index

The Calinski-Harabasz Index (CHI) measures the ratio of between-cluster dispersion to within-cluster dispersion. It is defined as:

$$CHI = \frac{B_k}{W_k} \cdot \frac{n - k}{k - 1},$$

where:

— B_k: The trace of the between-cluster dispersion matrix.

— W_k: The trace of the within-cluster dispersion matrix.

— n: The total number of data points.

— k: The number of clusters.

The between-cluster dispersion matrix is:

$$B_k = \sum_{i=1}^{k} n_i \|\mu_i - \mu\|^2,$$

where n_i is the size of cluster C_i, μ_i is the centroid of C_i, and μ is the overall mean of the dataset. The within-cluster dispersion matrix is:

$$W_k = \sum_{i=1}^{k} \sum_{x \in C_i} \|x - \mu_i\|^2.$$

Higher CHI values indicate better clustering.

4.5 Results

To determine the optimal number of sub-clusters within the medium-sized cluster, we evaluated clustering performance using several metrics. Agglomerative clustering was applied with the number of clusters ranging from 2 to 30, and the corresponding metric scores were plotted (see Figure 4.1).

From this analysis, we observed that the optimal number of clusters for our dataset is three. This conclusion is supported by the metric scores, which show a clear peak or stabilization at this value.

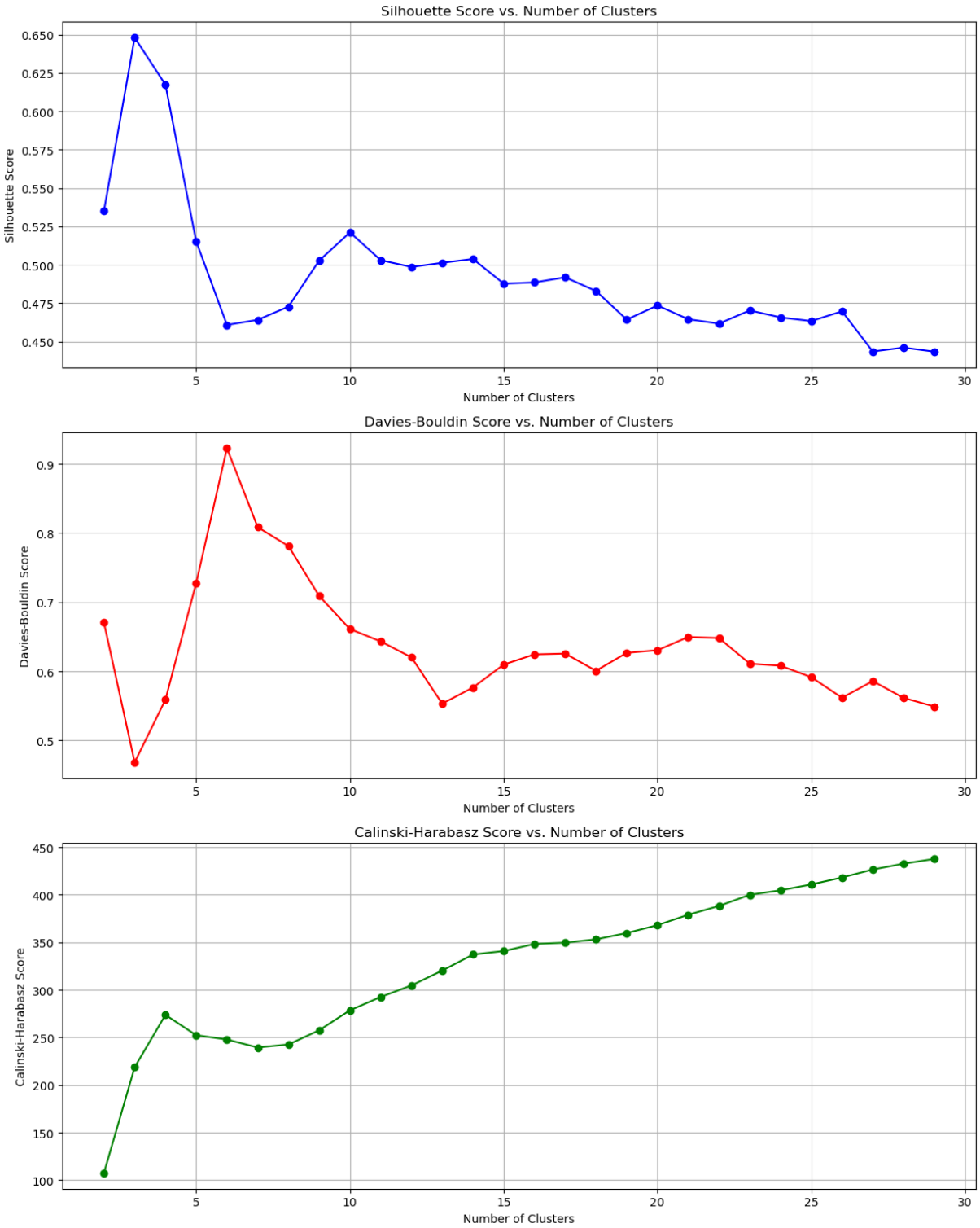


Figure 4.1: Evaluation metrics for different numbers of clusters using agglomerative clustering.

Chapter 5

Conclusion

Using the previously demonstrated and explained methods, we successfully identified distinct groups that warrant further analysis. Despite the absence of detailed descriptions of the features and their relative importance, our results can still contribute valuable insights to the medical field. By uncovering hidden patterns within the dataset, we provide a foundation for potential future studies that may lead to more precise diagnostic or therapeutic approaches.

Our analysis revealed three primary clusters. Interestingly, the largest and smallest clusters appear to be closely related, while the medium-sized cluster stands apart as a distinct entity. Moreover, within this medium cluster, we observed an internal structure suggesting the presence of three additional subclusters. This observation indicates the potential existence of subclasses within a specific type of cancer, highlighting the complexity and heterogeneity of the disease. Further investigation into these subgroups could lead to a deeper understanding of the underlying biological mechanisms and improve patient stratification for targeted treatments.

Bibliography

[1] URL: https://zenodo.org/records/8187729. (last access: 23.01.2025).

[2] URL: https://adelejackson.com/files/Maths_of_UMAP.pdf. (last access: 23.01.2025).

[3] URL: https://topos.institute/blog/2024-04-05-understanding-umap/. (last access: 23.01.2025).

[4] Martin Ester et al. "A density-based algorithm for discovering clusters in large spatial databases with noise". In: *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining (KDD)* (1996), pp. 226–231.

[5] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. 2nd. Springer, 2009, pp. 520–527.

List of Figures

List of Tables