Chapter 1

Introduction

We begin with the basic definitions and notations for topological data analysis. Some of the definitions were referenced from [10]. This section introduces the theoretical aspect of topology which will gradually build up towards the applications of topology in later sections.

1.1 Topology

Definition 1.1.1 (Topology)**.** Let X be a set and $\mathcal{P}(X)$ denote the power set of X. A topology on a set X is a subset $T \subseteq \mathcal{P}(X)$ such that:

(a) If S_1, $S_2 \in T$, then $S_1 \cap S_2 \in T$.

(b) Let I be the index set that labels the subscripts of S_i for any $i \in I$. If $\{S_i | i \in I\} \in T$, then $\cup_{i \in I} S_i \in T$.

(c) $\emptyset, X \in T$.

Definition 1.1.2 (Open, Closed Sets)**.** Let X be a set and T be a topology. $S \in T$ is an open set. Closed sets are $X - S$, where $S \in T$.

Remark 1.1.3. A set can be only open, only closed or both open and closed or even neither. For instance, $\emptyset$ is both open and closed by definition.

Definition 1.1.4 (Cover)**.** Let I be the index set that labels the subscripts of S_i for any $i \in I$. A cover C of a set X is a collection of sets $\{S_i | i \in I\}$ such that

$$X \subseteq \bigcup_{i \in I} S_i. \tag{1.1}$$

Definition 1.1.5 (Open Cover)**.** Let C be a cover of a set X and T be a topology of X. C is an open cover if every element in C is an open set. In other words, $S_j \subseteq T$ for all $j \in J$.

Definition 1.1.6 (Topological Space)**.** The pair (X, T) of a set X and a topology T is a topological space.

Remark 1.1.7. By convention, $\mathbb{X}$ is commonly used as the notation for a topological space (X, T).

Definition 1.1.8 (Contractible Space)**.** Let $\mathbb{X}$ be a topological space. $\mathbb{X}$ is contractible if it can be deformed into a single point within that space (see, e.g. Homotopy of [10]).

Definition 1.1.9 (Good Cover)**.** Let C be an open cover. C is a good cover if all its open sets and all intersections of finitely many open sets are contractible.

Next, we pay attention to the remaining definitions of Metric, Open Ball, Metric space and the Euclidean space.

Definition 1.1.10 (Metric)**.** A metric or distance function $d : X \times X \to \mathbb{R}$ is a function satisfying the following axioms:

(a) For all $x, y \in X$, $d(x, y) \geq 0$ (positivity).

(b) If $d(x, y) = 0$, then $x = y$ (non-degeneracy).

(c) For all $x, y \in X$, $d(x, y) = d(y, x)$ (symmetry).

(d) For all $x, y, z \in X$, $d(x, y) + d(y, z) \geq d(x, z)$ (triangle inequality).

Definition 1.1.11 (Open Ball)**.** The open ball $\mathcal{B}(x, r)$ with center x and radius $r > 0$ with respect to metric d is defined as $\mathcal{B}(x, r) = \{y | d(x, y) < r\}$.

Definition 1.1.12 (Metric Space)**.** A set X with metric d is known as a metric space. A metric space is also a topological space. Commonly defined metric spaces includes the Euclidean spaces.

Definition 1.1.13 (Euclidean Space)**.** The Cartesian product of n copies of $\mathbb{R}$ along with the Euclidean metric

$$d(x, y) = \sqrt{\sum_{i=1}^{n} (e_i(x) - e_i(y))^2} \tag{1.2}$$

is the n-dimensional Euclidean space $\mathbb{R}^n$.

1.2 Simplical Complex

Definition 1.2.1 (Simplex)**.** A geometric k-simplex $\sigma^k = \{v_0, v_1, v_2, \cdots, v_k\}$ is the convex hull formed by $k + 1$ affinely independent points $v_0, v_1, v_2, \cdots, v_k$ in the Euclidean space $\mathbb{R}^d$ as follows:

$$\sigma^k = \left\{ \lambda_0 v_0 + \lambda_1 v_1 + \cdots + \lambda_k v_k \middle| \sum_{i=0}^{k} \lambda_i = 1; 0 \leq \lambda_i \leq 1, i = 0, 1, \cdots, k \right\}.$$

A *face* τ of k-simplex σ^k is the convex hull of a non-empty subset. We denote it as $\tau \leq \sigma^k$. In general, an m-face of σ^k is the m-dimensional subset of $m + 1$ vertices, where $0 \leq m \leq k$. The most commonly used simplices in $\mathbb{R}^3$ are shown in Figure 1.

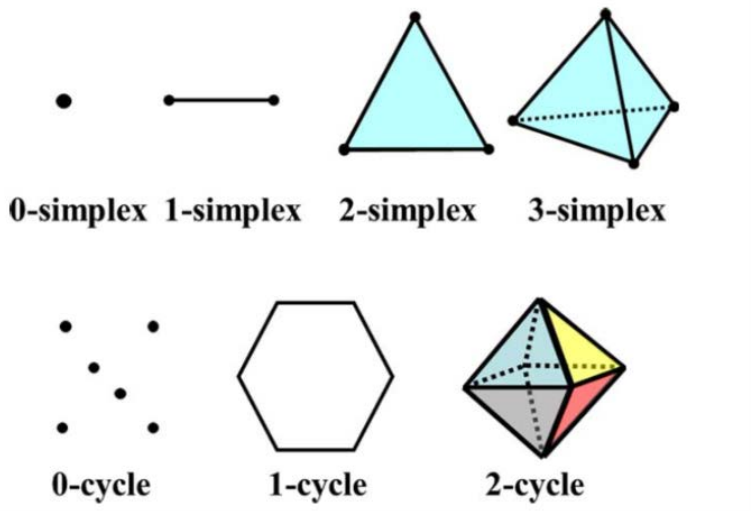


Figure 1.1: Illustration of Common Simplices

Definition 1.2.2 (Simplical Complex)**.** A geometric simplical complex K is a finite set of geometric simplices that satisfy two essential conditions:

- Any face of a simplex from K is also in K.
- The intersection of any two simplices in K is either empty or shares faces.

The commonly used methods to define simplical complexes are Čech complex, Vietoris-Rips complex, Alpha complex, Clique complex, Cubic complex, Morse complex, etc.

(a) An simplical complex.

(b) Not an abstract simplical complex. Their intersection is not along a shared simplex.

Figure 1.2: Illustration of a Simplical Complex

Definition 1.2.3 (Abstract Simplical Complex)**.** An abstract simplical complex K is a finite set of elements $\{v_0, v_1, \cdots, v_k\}$ called abstract vertices, together with a collection of subsets $(v_{i_0}, v_{i_1}, \cdots, v_{i_k})$ called abstract simplexes, with the property that any subset of a simplex is still a simplex.

Remark 1.2.4. Note that one important example of abstract simplicial complex is the Vietoris-Rips complex. More details will be illustrated in the next chapter.

Let X be a point set in Euclidean space $\mathbb{R}^d$ and $\mathcal{U}$ be a good cover of X, i.e. $X \subseteq \cup_{i\in I}\mathcal{U}_i$.

Definition 1.2.5. The nerve $\mathcal{N}$ of $\mathcal{U}$ is defined by the following two conditions:

- $\emptyset \in \mathcal{N}$.

- If $\cap_{j\in J}\mathcal{U}_j \neq \emptyset$ for $J \subseteq I$, then $J \in \mathcal{N}$.

Theorem 1.2.6 (Nerve Theorem)**.** *The geometric realization of the nerve of* $\mathcal{U}$ *is homotopy equivalent to the union of sets in* $\mathcal{U}$*.*

Definition 1.2.7 (Čech Complex)**.** The Čech complex $\check{\mathrm{C}}_\varepsilon(X)$ with parameter ε of X is the nerve of the collection of balls $\mathcal{B}(X,\varepsilon)$.

$$\text{i.e. } \check{\mathrm{C}}_\varepsilon := \left\{\sigma \in X \,\middle|\, \bigcap_{x\in\sigma} \mathcal{B}(X,\varepsilon) \neq \emptyset\right\}.$$

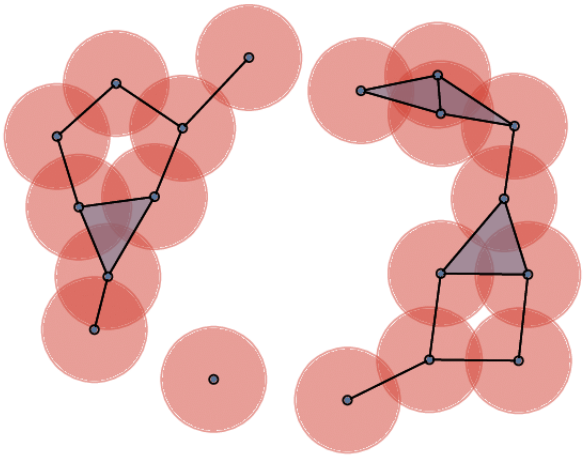


Figure 1.3: Example of Čech complex

Definition 1.2.8 (Vietoris-Rips Complex)**.** The Vietoris-Rips complex $\mathcal{R}_\varepsilon(X)$ with parameter ε is the set of all $\sigma \subseteq X$, such that the largest Euclidean distance between any of its points is at most 2ε.

Another simplical complex used is the alpha complex. The alpha complex is used in computational geometry. It is a set of subcomplexes from Delaunay triangulation of a point set in $\mathbb{R}^d$.

Definition 1.2.9. Let X be a finite point set in $\mathbb{R}^d$. The Voronoi cell of a point $x \in X$ is the set of points $V_x \subseteq R^d$ for which x is the closest of the points in X, i.e. $V_x = \{u \in \mathbb{R}^d : |u - x| \leq |u - x'|, \forall x' \in X\}$.

Definition 1.2.10 (Delaunay Complex)**.** The Delaunay complex of a finite set $X \in \mathbb{R}^d$ is defined as the nerve of the Voronoi diagram.

We then define $R_\delta(x)$ to be the intersection of the Voronoi cell V_x with $\mathcal{B}(X,\delta)$, i.e. $R_\delta(x) = V_x \cap \mathcal{B}(X,\delta)$ for every $x \in X$.

Definition 1.2.11 (Alpha Complex)**.** The alpha complex $A_\delta(X)$ is defined as the nerve of the covers formed by $R_\delta(x)$ for every $x \in X$, i.e. $A_\delta(X) := \{\sigma \in X | \cap_{x\in\sigma} R_\delta(x) \neq \emptyset\}$.

1.3 Homology

The fundamental group is a great tool for characterizing holes in topological spaces. Unfortunately it is difficult to work with algorithmically. Homology groups give the best of both worlds. They are useful for characterizing surfaces and provide fast algorithms. In this section, we first introduce algebraic topology, which uses algebra to understand homology groups. We then extend the idea of homology groups to persistent homology which becomes the foundation of topological data analysis.

1.3.1 Chains

Let K be a simplical complex. We define a p-chain as a finite formal sum of p-simplices in K, written as $c = \sum a_i \sigma_i$. In this class, we consider the coefficients of a_i under $\mathbb{Z}_2$. The p-chain is simply a collection of p-simplices. We can define a binary operation $+$, over the set of p-chains for a simplical complex as follows. Let $c_0 + c_1 = \sum a_i \sigma_i + \sum b_i \sigma_i = \sum (a_i + b_i \mod 2)\sigma_i$, since the addition over $\mathbb{Z}_2$ is addition modulo 2.

Lemma 1.3.1. *Let K be a simplical complex and let C_p be the set of p-chains over K. The set C_p with the operator $+$ forms a group, denoted by $(C_p, +)$. Furthermore, $(C_p, +)$ is a free abelian group.*

Let $[v_0, \cdots, v_p]$ denote a p-simplex. The *boundary* of this p-simplex is the various $(p-1)$-dimensional simplices $[v_0, \cdots, \hat{v}_i, \cdots, v_p]$, where the $\hat{}$ symbol means that v_i has been removed from the sequence $v_0, \cdots, v_p$. We define the boundary operator ∂_p for a simplex σ as follows:

$$\partial_p \sigma = \sum_{i=0}^{p} (-1)^i [v_0, \cdots, \hat{v}_i, \cdots, v_p].$$

Note that the signs are inserted to take orientation into account as shown below:

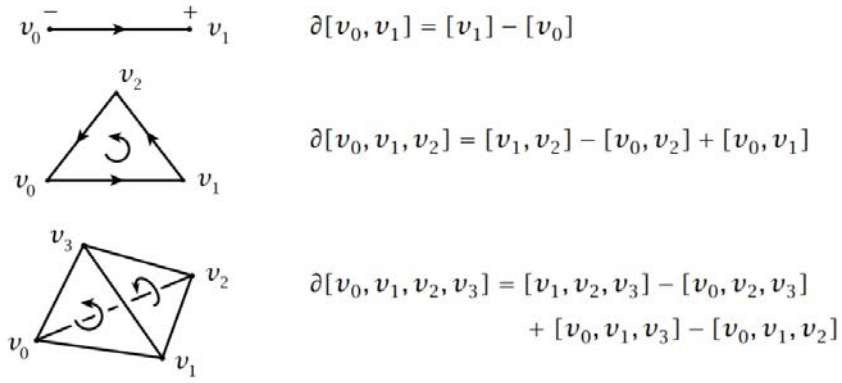


Figure 1.4: Examples of ∂_p

The boundary operator ∂_p has the following nice properties:

- For a given p-chain $c = \sum a_i \sigma_i$, the boundary is the sum of the boundaries of its simplices, $\partial_p c = \sum a_i \partial_p \sigma_i$.
- $\partial_p(\lambda c) = \lambda \partial_p c$.
- $\partial_p(c_i + c_j) = \partial_p(c_i) + \partial_p(c_j)$,

where c_i, c_j are both chains and λ is a constant.

Thus, the map $\partial_p : C_p \to C_{p-1}$ is a homomorphism. Hence, the chain complex is a sequence of chain groups connected by the boundary homomorphisms.

$$\cdots \xrightarrow{\partial_{p+2}} C_{p+1} \xrightarrow{\partial_{p+1}} C_p \xrightarrow{\partial_p} C_{p-1} \xrightarrow{\partial_{p-2}} \cdots$$

Lemma 1.3.2 (Fundamental Lemma of Homology)**.** *Intuitively, this lemma states that the boundary of the boundary is the zero map. i.e. For every integer p and every $(p+1)$-chain complex d, $\partial_p \partial_{p+1} d = 0$.*

1.3.2 Simplical Homology

In order to define homology groups, we must first focus on two particular types of chains, i.e. p-cycles and p-boundaries. A p-cycle c is a p-chain whose boundary is zero. i.e. $\partial_p c = 0$. The set of all p-cycles form a group denoted as $Z_p \subseteq C_p$. Z_p is a subgroup of C_p. Since Z_p is the set of all p-chains that are mapped to identity of C_p under the pth boundary homomorphism, Z_p is the *kernel* of ∂_p denoted as $Z_p = \ker \partial_p$. The p-boundaries then form another abelian group, denoted as $B_p = \operatorname{Im} \partial_{p+1}$.

Due to the Fundamental Lemma of Homology, $B_p \subseteq Z_p$. B_p is also a subgroup of Z_p. Hence, we can construct a quotient group via the cosets of p-cycles.

Definition 1.3.3. The pth homology group is the quotient group $H_p = Z_p / B_p$. The *rank* of pth homology group H_p is called the pth Betti number and satisfies $\beta_p = \operatorname{rank} Z_p - \operatorname{rank} B_p$.

Here, β_p is the pth Betti number of a simplical complex K. Informally, the pth Betti number is the number of unconnected p-dimensional surfaces. The first few Betti numbers have an easy intuitive meaning.

- β_0 is the number of connected components.
- β_1 is the number of two-dimensional or "circular holes".
- β_2 is the number of three-dimensional holes or "voids".

For example, a torus has Betti numbers: $\beta_0 = 1$ (1 connected component), $\beta_1 = 2$ (The middle of torus and the circle in tube) and $\beta_2 = 1$ (The inside of the tube).

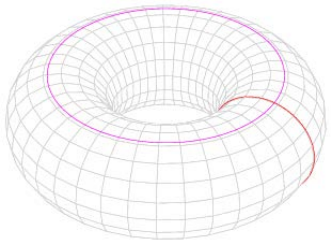


Figure 1.5: Illustration of Torus

Additionally, in the illustration of the chain complexes, we can visualize via the following:

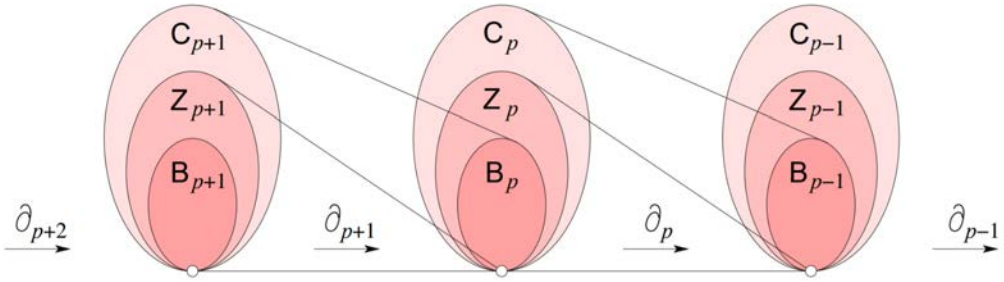


Figure 1.6: Illustration of Chain Complex as a linear sequence of chain, cycle, and boundary groups connected by boundary homomorphisms

1.3.3 Persistent Homology

The essense of Persistent Homology (PH) is its filtration process, during which a series of topological spaces in different scales are generated.

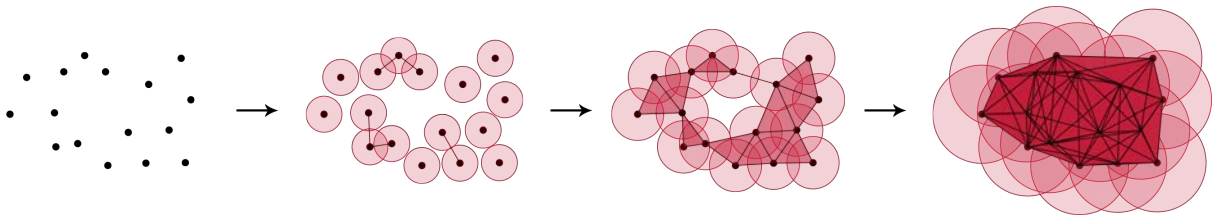


Figure 1.7: Illustration of Filtration Process in PH

In other words, PH is a method of reintroducing metric measurements to the topological structures. A specified measurement is used to an index i to a sequence of nested topological spaces $\{\mathbb{X}_i\}$. Such a sequence is thus a filtration,

$$\emptyset \subseteq \mathbb{X}_0 \subseteq \mathbb{X}_1 \subseteq \cdots \subseteq \mathbb{X}_m = \mathbb{X}.$$

As each inclusion induces a mapping of chains, it induces a linear map of homology,

$$\emptyset \subseteq H(\mathbb{X}_0) \subseteq H(\mathbb{X}_1) \subseteq \cdots \subseteq H(\mathbb{X}_m) = H(\mathbb{X}).$$

The above sequence then describes an evolution of homology generators. We first define a composition mapping from $H(\mathbb{X}_i)$ to $H(\mathbb{X}_j)$ as $\xi_i^j : H(\mathbb{X}_i) \to H(\mathbb{X}_j)$. This sequence can then be observed as a birth and death process where new homology classes are born or dead throughout this evolution. We can then associate a duration or persistence length for each homology generator c. i.e. $\text{persist}(c) = h_j - h_i$. The persistence length can then be represented as a barcode with respect to the Betti number β_n. Figure 1.8 provides a visual illustration of the filtration process using the nested topological spaces.

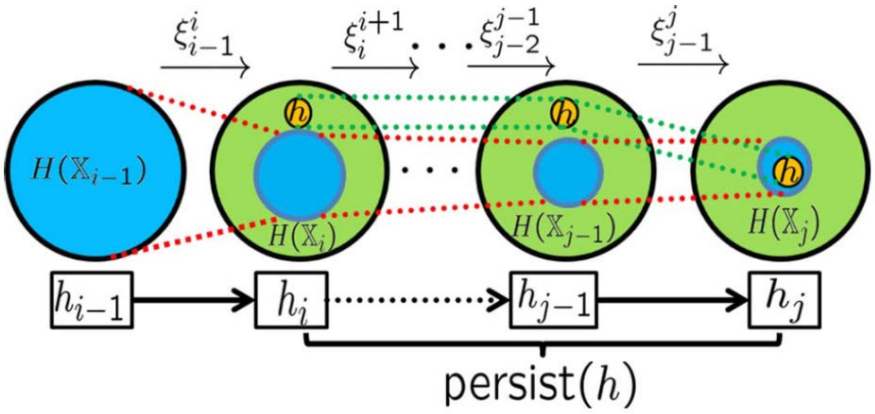


Figure 1.8: Illustration of Birth and Death of Homology Generator [7]

This section concludes the basic theoretical knowledge of Persistent Homology. In the next few chapters, we will visit several applications of Persistent Homology and its significant connections to biology and Protein-Ligand Binding.

Chapter 2

Applications of Persistent Homology

In this chapter, we review and discuss two basic applications of persistent homology. First, we review the techniques for applying persistent homology on point cloud data. Secondly, we review the techniques for applying persistent homology on volumetric data.

2.1 Persistent Homology of Fullerene

Fullerene is one of the simplest molecular structures. In particular, the Fullerene C_{60} has just 60 carbon atoms with a mixture of single and double bonds. Hence, Fullerene serves as one of the basic examples in illustrating Persistent Homology as it is not computationally expensive and it consists of symmetry which can be observed through persistent homology. Figure 2.1 illustrates the visual representation of filtration of C_{60}.

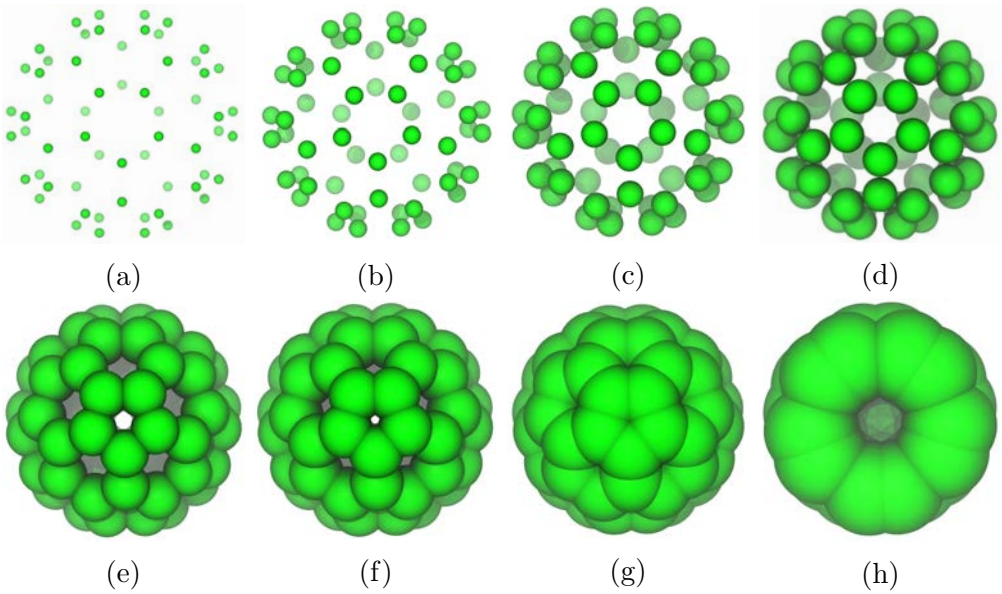
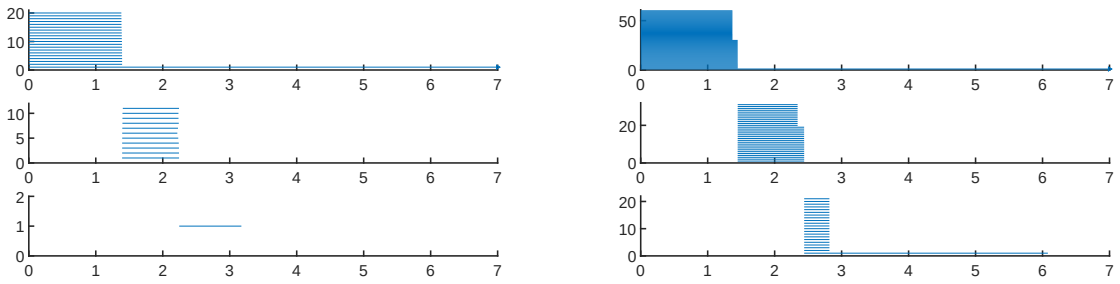


Figure 2.1: Filtration of C_{60}

By applying the Vietoris-Rips complex up to a certain threshold distance d on the filtration parameter, we can simply implement the Euclidean-distance based filtration on C_{20} and C_{60}. This is one example of a point cloud input filtration where our input data is the 3-D coordinates of the atoms of C_{20} and C_{60}. This method can be easily performed via

JavaPlex[13]. It is also mentioned in [7] that this is commonly used due to its efficiency and simplicity. Figure 2.2 shows the barcodes obtained via JavaPlex.



(a) (b)

Figure 2.2: (a): Barcodes for Fullerene C_{20} (b): Barcodes for Fullerene C_{60}

Before we describe the barcodes obtained via JavaPlex, we introduce the definition of Euler characteristic, a topological property that is invariant under nondegenerate shape deformation. This definition is needed in order to understand the topological fingerprints from the barcodes.

Definition 2.1.1 (Euler Characteristic)**.** The Euler characteristic χ was classically defined for the surfaces of polyhedra, according to the formula

$$\chi = V + E + F,$$

where V, E, F are the number of vertices (corners), edges and faces of the polyhedron respectively. Any convex polyhedron has the following equation

$$V + E + F = 2.$$

This equation is known as the Euler's polyhedron formula.

In this case, for the fullerene cage C_N, we have n_p and n_h number of pentagons and hexagons respectively. It is not difficult to see that the number of faces of C_N is simply $(n_p + n_h)$. Furthermore, every vertex in C_N is shared by 3 faces and every edge in C_N is shared by 2 faces. Hence, we obtain the 2 formulas for V and E:

$$N = V = \frac{5n_p + 6n_h}{3}, \quad E = \frac{5n_p + 6n_h}{2}.$$

By the Euler characteristic, we can solve the following equation to obtain $n_p = 12$. In particular, since $N = V$, if $N = 60$, $n_h = 20$ and if $N = 20$, $n_h = 0$.

Now we discuss the barcodes obtained for C_{20} and C_{60} in Figure 2.2. The figure shows the barcode plots of C_{20} and C_{60} for β_0, β_1 and β_2. From Section 2.3, the persistence length of each homology generator in fullerene represents itself as a barcode in the Betti number β_n. i.e. $\text{persist}(c_i^n) = L_i^{\beta_n}$ where $L_i^{\beta_n}$ is the persistence length of the ith homology generator in β_n. In the case of the fullerene, β_0 represents the number of atoms in the fullerene cage and as the radius of the points increases, the points will grow into large spheres which intersects with other spheres and generate higher dimensional simplices. When the spheres intersect, the barcodes in β_0 also ends. Hence, the persistence length of the β_0 bars actually provides the information about the bond length between the carbon

atoms in fullerene. In the case of C_{20}, the β_0 bars are all around 1.45Å while in the case of C_{60}, it can be seen that there are 30 longer bars that persist up till 1.45Å and the other 30 shorter bars that only persist up till 1.37Å. This is because the shorter bars are attributed from the double bonds of carbon atoms which have stronger bonds and are nearer whereas the 30 longer bars are due to single carbon bonds.

For the β_1 bars, we note that from the Euler characteristic, C_{20} has 12 pentagons and 0 hexagons. However, if we observe closely at the barcodes of β_1, we can only find 11 bars which represents 11 cycles based on the previous section. This is because the edges of the 12th pentagon can be constructed as a linear combination of the other 11 pentagons. The 11 bars persist from 1.45Å to 2.34 Å as at 2.34Å, the spheres overlap and form a surface around the pentagon. Similarly, by the Euler characteristic, we also have 12 pentagons in C_{60} hence we have 11 bars of β_1 persisting from 1.45Å to 2.35Å. However, C_{60} has 20 hexagons hence we have another 20 β_1 bars persisting from 1.45Å to 2.44Å.

Lastly, for the β_2 bars, there is only one β_2 bar for fullerene C_{20} which corresponds to the void in the center of the fullerene. However, for the fullerene C_{60}, there are 20 β_2 bars persisting from 2.44Å to 2.82Å. This is due to the fact that the Vietoris-Rips complex is an abstract simplicial complex and when upon reaching 2.44Å, each hexagon forms 8 triangles (2-simplex) which under the definition of abstract simplicial complex, the subset forms a 3-simplex which contributes to a β_2 bar since an abstract simplicial complex does not account for geometrical properties. Figure 2.3 shows the 8 triangles. Since there are 20 hexagons in C_{60}, hence there are 20 β_2 bars as such in Figure 2.2. The fullerene C_{60} also has one β_2 bar due to the void in the center (See the last image of Figure 2.1).

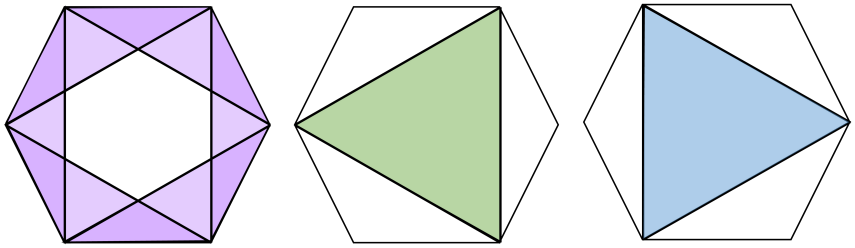


Figure 2.3: Illustration of the 8 triangles that form a 3-simplex.

Next we introduce another method of filtration where we construct a correlation matrix between every two atoms in the molecule from the point cloud data. This is known as the correlation matrix based filtration. Based on past literature in protein stability (see [7]), the flexibility-rigidity index (FRI) theory has proven to be an efficient and accurate model. Here, we define the correlation matrix based on the FRI.

Definition 2.1.2 (Correlation Matrix)**.** Let the coordinates of the atoms in a molecule be denoted as $\mathbf{r}_1, \mathbf{r}_2, \cdots, \mathbf{r}_N \in \mathbb{R}^3$ and the Euclidean distance between the ith atom and jth atom be r_{ij}.Then the ijth entry of general correlation matrix is

$$C_{ij} = w_j \Phi(r_{ij}, \eta_j), \tag{2.1}$$

where w_j is a weight parameter, $\eta_j > 0$ is the atom-type characteristic distance and $\Phi(r_{ij}, \eta_j)$ is the radial basis correlation kernel.

Due to the significance of FRI model, the choice of kernel narrows down to two commonly used kernels which produces highly predictive results (see [3, 8, 9]). The two commonly used kernels are the Exponential Kernel and the Lorentz Kernel. The Exponential Kernel is defined as

$$\Phi(r_{ij}, \eta_j) = e^{-(r_{ij}/\eta_j)^\kappa}, \quad \eta_j > 0, \kappa > 0. \tag{2.2}$$

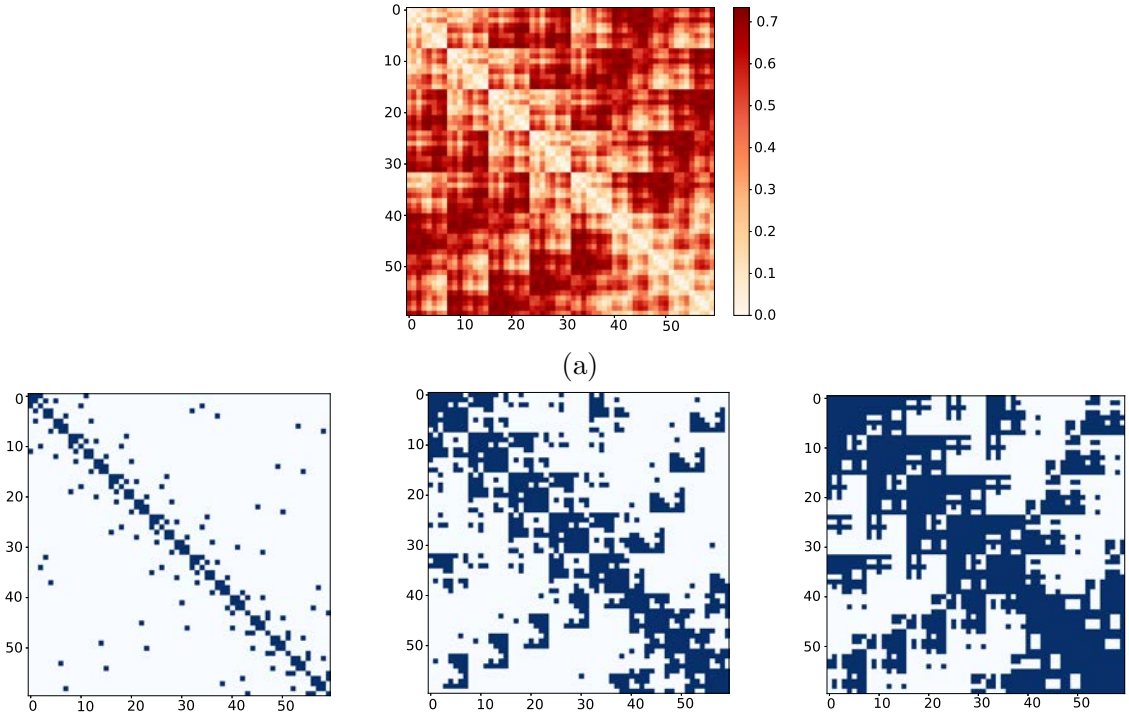
Note that when $\kappa = 2$, we get the special case which is known as the Gaussian Kernel. The Lorentz Kernel is defined as

$$\Phi(r_{ij}, \eta_j) = \frac{1}{1 + (r_{ij}/\eta_j)^\nu}, \quad \eta_j > 0, \nu > 0. \tag{2.3}$$

In order to view the correlation matrix, we notice that Equation (2.1) presents an opposite meaning of persistent homology. As a workaround, we simply use the matrix M_{ij}:

$$M_{ij} = 1 - w_j \Phi(r_{ij}, \eta_j). \tag{2.4}$$

Figure 2.4(a) shows the correlation matrix constructed from Equation (2.4) with Gaussian Kernel and with parameters $\eta = 6.0$Å and $w_j = 1.0$ for all $1 \leq j \leq N$. Both axes on the plot represents the atomic numbers. Note that as the distance between the ith and jth atoms is nearer, the correlation values are higher and this can be seen from the plot with the darker regions of red. This is also similar to observing the Rips complex. To observe the correlation matrix better, we set a filtration threshold d that limits the max distance between two atoms. Figures 2.4(b)-(d) shows the connectivity of atoms with the threshold $d = 0.1$, 0.3 and 0.5Å respectively. The dark blue regions represents atom pairs forming simplices with each other.



(a) (b) (c) (d)

Figure 2.4: (a): Correlation Matrix based filtration of Fullerene C_{60} (b)-(d): Connectivity Matrix between atoms with filtration threshold $d = 0.1$Å, 0.3Å, 0.5Å respectively.

Next, we construct nanotube structures using VMD to observe its barcode properties as illustrated in Figure 2.5. Our nanotube is constructed by setting the nanotube length to a 10 unit layers and 3 unit layers. The coordinates were then extracted and converted to barcode representation via JavaPlex. Unlike the barcodes in Figure 2.2, the long β_1 bar is due to the tube circle of the whole nanotube structure. The β_2 bars have 2 distinctive regions: one is around $2.5 - 2.7$Å and another appears after 7.0Å.

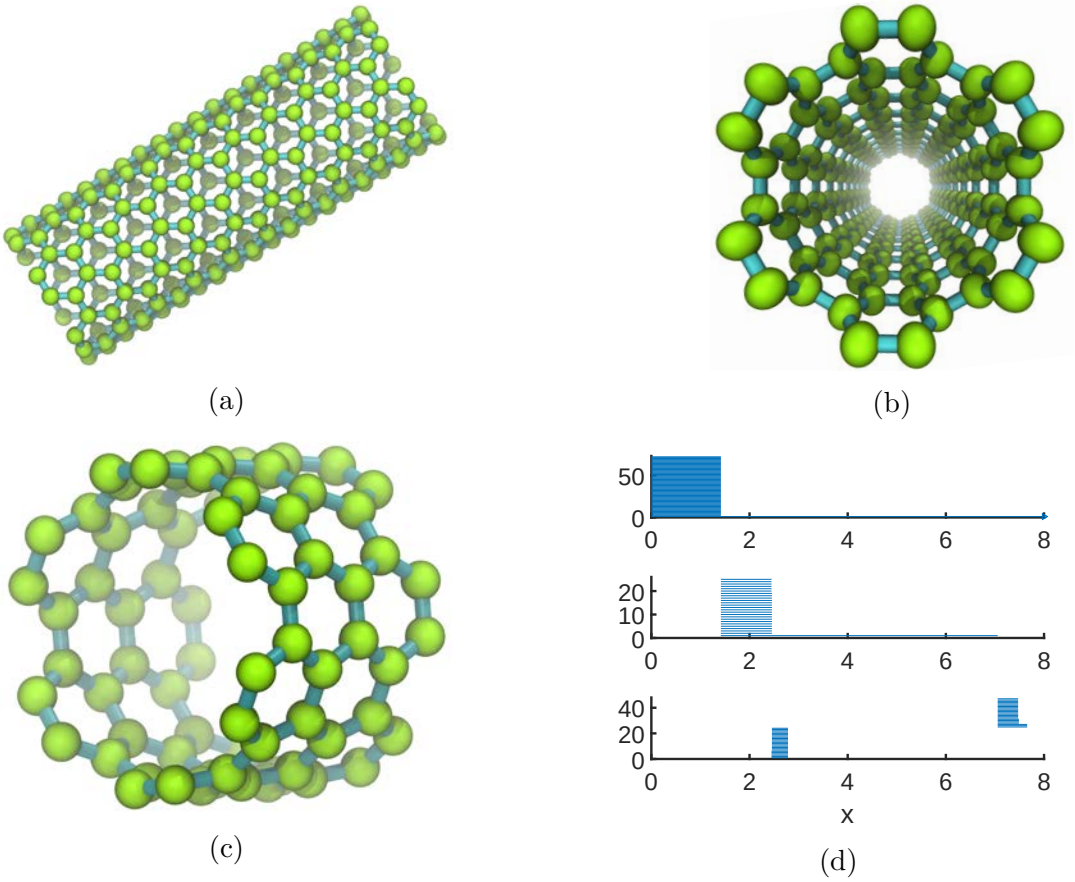


Figure 2.5: Illustration of persistent homology analysis for a nanotube. (a)-(b) The generated nanotube structure of 10 layers. (c) A 3-unit layer segment extracted from the nanotube in (a). (d) Barcodes representation of the topology of the nanotube segment.

In summary, this section shows that the persistent homology is able to represent the topological features of biomolecules at multiscale level. More importantly, these multiscale representation may provide some correlation to certain significant biological properties of biomolecules such as curvature energies, B-Factors, binding affinities etc. For example, in [7], strong correlations were found between the betti bars and predictions of curvature analysis. In the next section, we discuss the use of multiresolution geometric representation by using density data derived from the original point cloud.

2.2 Multiresolution Topological Simplification

After investigating basic filtration of biomolecules via point cloud data, we turn to investigating filtrations using density data calculated from point cloud. For example, we will construct a multiresolution geometric representation using the Flexibility-Rigidity Index

(FRI) method. This method converts the point cloud data into a density map. In order to understand this method, we first experiment on a 2-dimensional hexagonal fractal image.

A two-dimensional hexagonal point cloud data which consists of the 6 corners of small and big hexagons is first constructed. A simple illustration of hexagonal fractal image is presented in Figure 2.6. The points are constructed as follows: A large hexagon H with center $(0, 0)$ and of edge length 20 is set. Note that we can simply compute the 6 corners of the any hexagon with center (x, y) and edge length l as our new points in the following manner: $(x + l, y)$, $(x - l, y)$ and $(x \pm l/2, y \pm \frac{\sqrt{3}}{2}l)$. Hence, for hexagon H, we have these 6 points: $(-20, 0)$, $(20, 0)$ and $(\pm 10, \pm 10\sqrt{3})$. The rest follows by taking each of these 6 points as centers and constructing another hexagon with edge length 5. Lastly, we repeat the process again by obtaining another set of hexagons with edge length 1.25. In total, we have 216 points generated after the last step (see the red points in Figure 2.6).

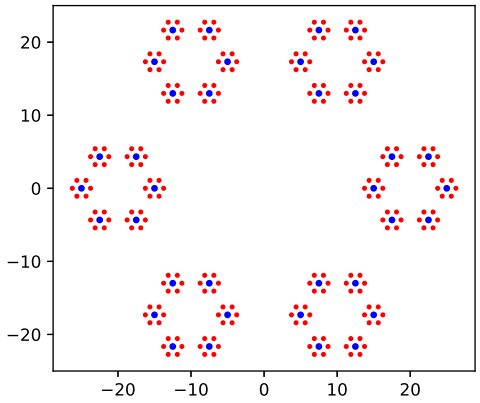


Figure 2.6: Illustration of 2D Hexagonal Point Cloud Data in $\Omega = [-30, 30] \times [-30 \times 30]$

Next, using the rigidity index of the ith entry,

$$\text{i.e. } \mu_i = \sum_{j=0}^{N} w_j \Phi(r_{ij}; \eta_j), \tag{2.5}$$

where $r_{ij} = ||\mathbf{r}_i - \mathbf{r}_j||$ is the generalized distance between the ith and jth entries, w_j is a weight, which can be the element number of the jth atom, and $\Phi(r_{ij}; \eta_j)$ is a real-valued monotonically decreasing correlation function satisfying the following conditions:

$$\Phi(r_{ij}; \eta_j) = 1 \text{ as } r_{ij} \to 0. \tag{2.6}$$
$$\Phi(r_{ij}; \eta_j) = 0 \text{ as } r_{ij} \to \infty. \tag{2.7}$$

In this method, we use GUDHI [12] to compute the persistent homology of the cubical complex of this our input data. Note that GUDHI uses Perseus as a backend C++ library. In addition, note that Perseus/GUDHI only works when the input data are in integers. Perseus was created by Konstantin Mischaikow and Vidit Nanda to compute the persistent homology of a cubical complex with reduction in computational complexity. This was performed via a series of algorithms and by constructing a morse complex (see

[11]). Hence, the input data $\mu(\mathbf{r})^s$ is prepared by normalizing the matrix $\mu(\mathbf{r})$ via the following equation:

$$\mu(\mathbf{r})^s = 1 - \frac{\mu(\mathbf{r})}{\mu_{\max}}, \quad \forall \mathbf{r} \in \Omega, \tag{2.8}$$

where $\mu(\mathbf{r})$ and $\mu(\mathbf{r})^s$ are the original data and input data respectively. $\mu_{\max}$ is the maximum density value in the matrx $\mu(\mathbf{r})$. Using the input data, we produce the multiresolution geometric representation of the 2D hexagonal fractal image as shown in Figure 2.7.

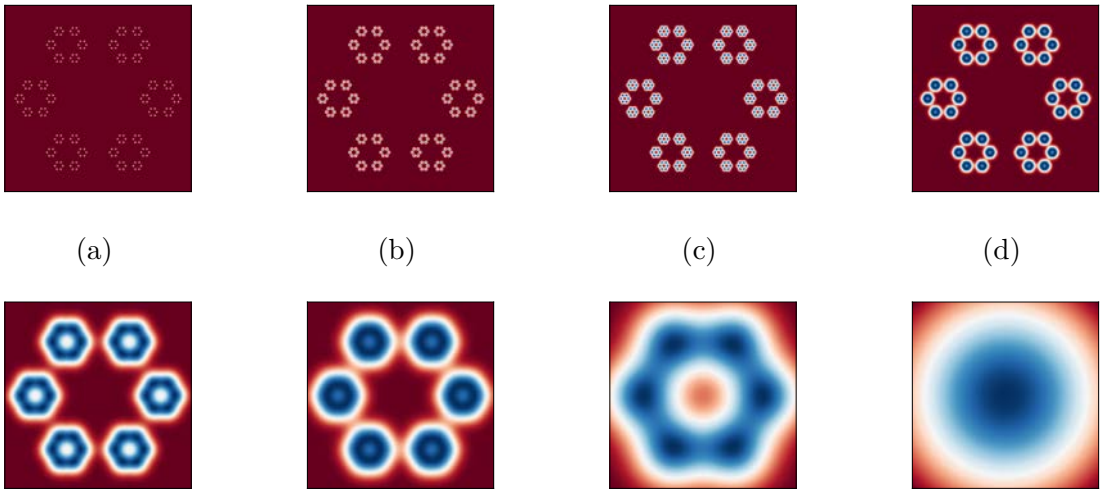
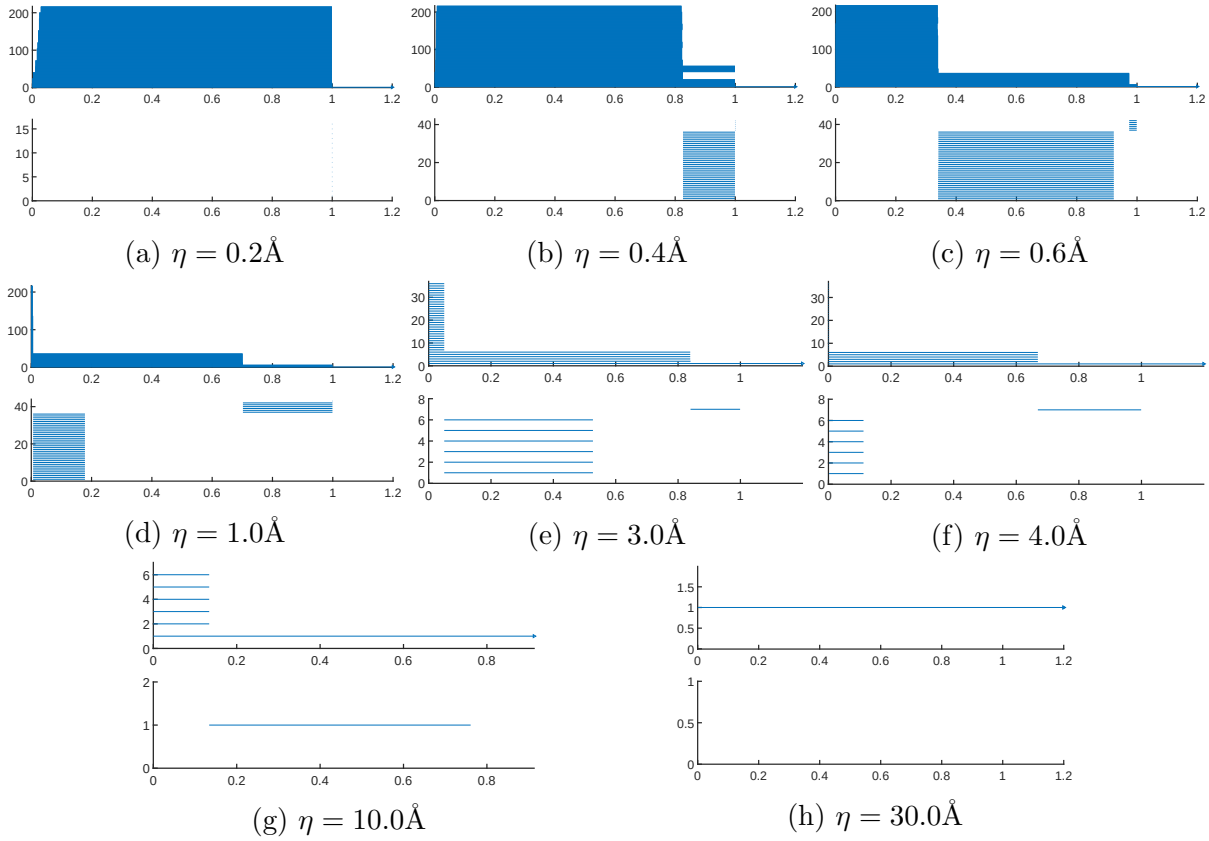


Figure 2.7: Illustration of multiresolution geometric analysis of a 2D hexagonal fractal image. The rigidity functions $\mu(r)$ are constructed from various η values. From (a) to (h), η values are set to 0.2, 0.4, 0.6, 1.0, 3.0, 4.0, 10.0, 30.0, respectively.

Using the python library GUDHI, we can use the 2D density matrices with various η to perform a cubical complex to obtain the persistence barcodes for respective values of η. Figure 15 shows the barcode results from GUDHI. Similar to the barcodes in the previous section, β_0 here represents the no. of points in our data which is 216 in Figure 2.8(a). As η increases, the resolution becomes poorer and the β_0 decreases until 36 bars in Figure 2.8(e). This is due to the individual points starting to form edges and hence creating 6 small hexagons in β_1 of Figure 2.8(e). The β_0 decreases further until 6 in Figure 2.8(g) and finally 1 in Figure 2.8(h). This is consistent with the images in Figure 2.7(g) and (h) respectively with the 6 large hexagonal points and the 1 large hexagon ring. More observations from the barcodes can be found in [5, 6]. This multiresolution topological analysis can be visualized further in 2D persistent homology. By discretization of the barcodes, we can bin the number of barcodes within the given filtration parameter range. This value is known as the Persistent Betti Numbers (PBNs). By varying η and performing this binning process, we can plot the $\log_{10}(\text{PBN} + 1)$ values against η. Figure 2.9 shows the β_0 and β_1 plots. Several large colour bands were observed in this 2D persistent homology which captures the unique and sequential multiscale topological properties of the data. Moreover, these images can be further used as features for training in convolutional neural networks or in machine learning methods search for correlations with other biological properties too.

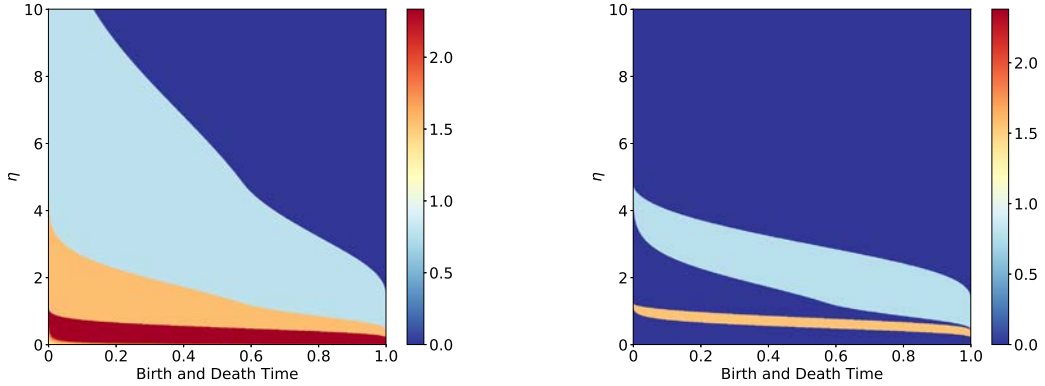


(a) $\eta = 0.2$Å (b) $\eta = 0.4$Å (c) $\eta = 0.6$Å

(d) $\eta = 1.0$Å (e) $\eta = 3.0$Å (f) $\eta = 4.0$Å

(g) $\eta = 10.0$Å (h) $\eta = 30.0$Å

Figure 2.8: Multiresolution persistence of the hexagonal fractal image.



(a) β_0 PBN (b) β_1 PBN

Figure 2.9: Illustration of 2D persistent homology in terms of PBNs for the hexagonal fractal images.

Essentially, this method helps us to analyse the topological patterns of larger biomolecules and the resolution can be adjusted to suit our interest. As seen in [4, 5, 6], this method was implemented to analyse several larger macromolecules and cryo-EM data which is better than the standard filtration based methods in the previous section. For instance, a multiscale geometric analysis on DNA molecule 2M54 is presented in Figure 2.10. Although the barcodes Note that we could perform a Vietoris-Rips complex filtration on

many biomolecules (e.g. PDB ID: 2M54) but such filtration does not show a relationship on how the density affects the topological data analysis of biomolecules by varying η. More importantly, a normal flexibility-rigidity model does not tell us how to vary the molecular structure to observe the changes in the topological fingerprints. Furthermore, it is also stated in [5, 6] that it becomes computationally too expensive to perform standard FRI models on viruses and microtubules as the structure contains huge number of atoms.

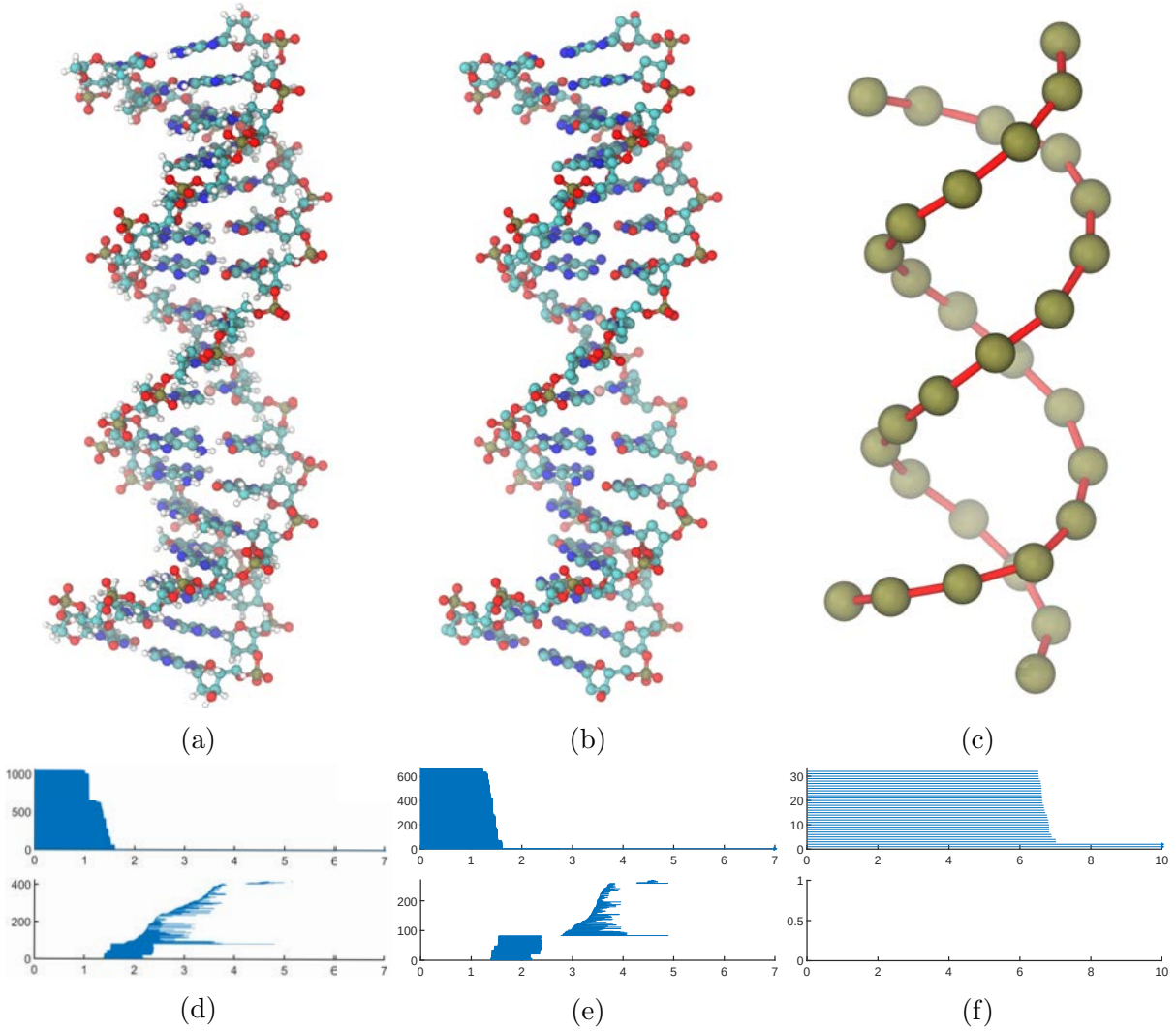


Figure 2.10: Illustration of multiresolution geometric analysis of the DNA molecule 2M54. (a): All atom representation. (b): All atom representation without hydrogen atoms. (c): Coarse-grained representation with only phosphorus atoms. (d) shows that with the hydrogen atoms, the local and global representations are not separated in β_1 whereas in (e), we can see the local topological fingerprints of pentagons and hexagonal rings appearing after 1.2Å. The global topological fingerprints started to appear awhile later. As for (c), we see that for β_0, we started off with the number of phosphorus atoms and ended with 2 long β_0 bars which represents the double string structure in (f).

We now repeat this multiresolution topological analysis for biomolecules. We consider the RNA molecule of 4QG3 which can also be found in [5, 6]. The PDB file obtained for 4QG3 is then converted into a density map file (i.e. .dx file) where we can visualise the geometric structure of the molecule given a fixed value of η and varying the isovalue in

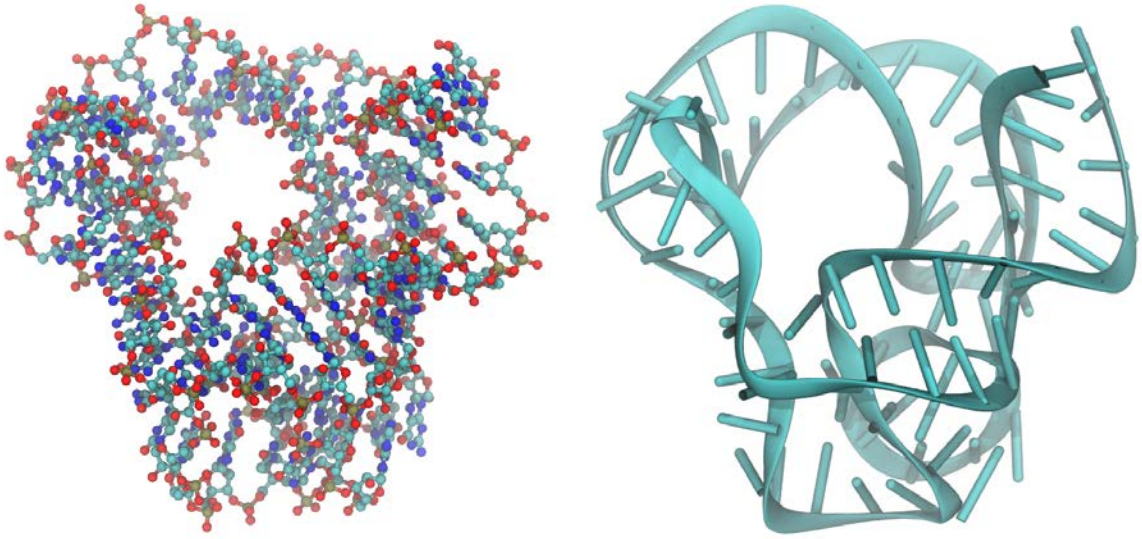
VMD. In this implementation, we use the same rigidity index shown previously:

$$\text{i.e. } \mu(\mathbf{r}) = \sum_{j=0}^{N} w_j \Phi(||\mathbf{r} - \mathbf{r}_j||; \eta). \tag{2.9}$$

Here, w_j refers to the atomic number of the jth atom. For example,

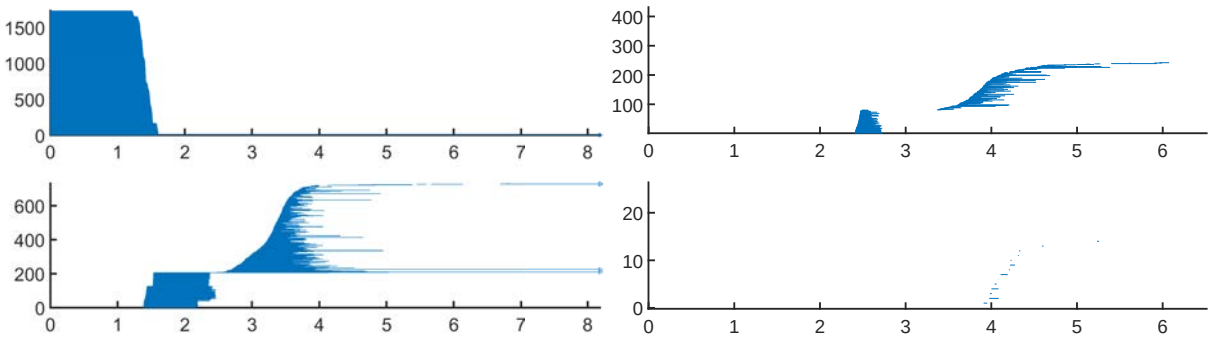
```
{'C':6, 'N':7, 'O':8, 'S':16, 'P':15, 'F':9, 'CL':17, 'BR':35, 'I':53}
```

We first illustrate the multiscale geometric analysis of RNA 4QG3 in Figure 2.11. Figure 2.11(c) shows the β_0 and β_1 barcodes. There are some similar observations as Figure 2.10. On one hand, as there are no hydrogen atoms in the RNA, the local pentagon and hexagon rings (from nucleotide base and sugar) appears earlier in β_1 for Figure 2.11(c). On the other hand, we can see that the global properties such as the global helix structure in (b) is evident from the β_1 bars that appeared later in (c). There might be some local properties that also started appearing earlier in β_2 in Figure 2.11(d). However, it is still difficult to deduce the topological fingerprints for β_2 and β_3 in (d).



(a) Atom-Bond Representation

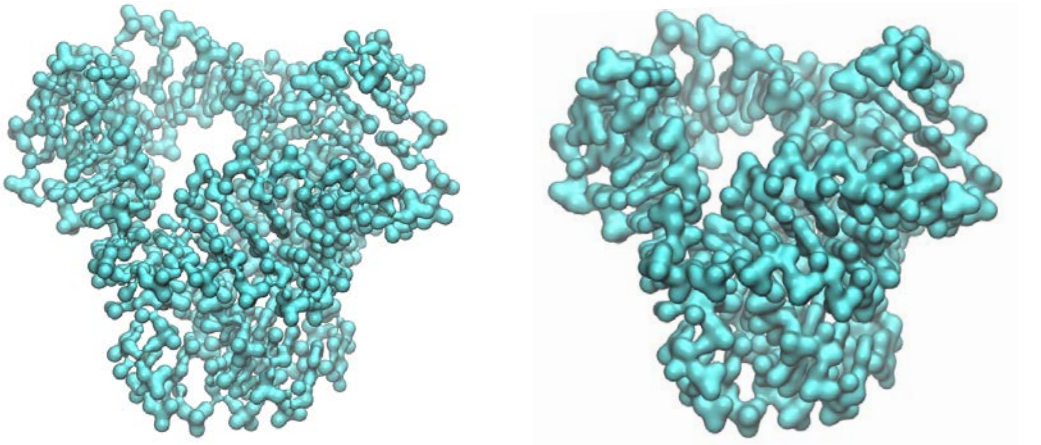
(b) Cartoon Representation



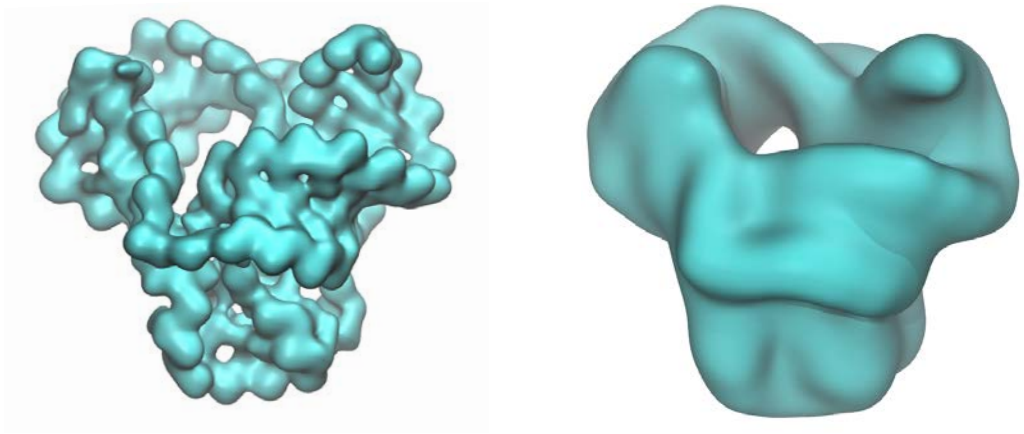
(c) β_0 and β_1

(d) β_2 and β_3

Figure 2.11: Illustration of geometric analysis of the RNA of 4QG3 structure.

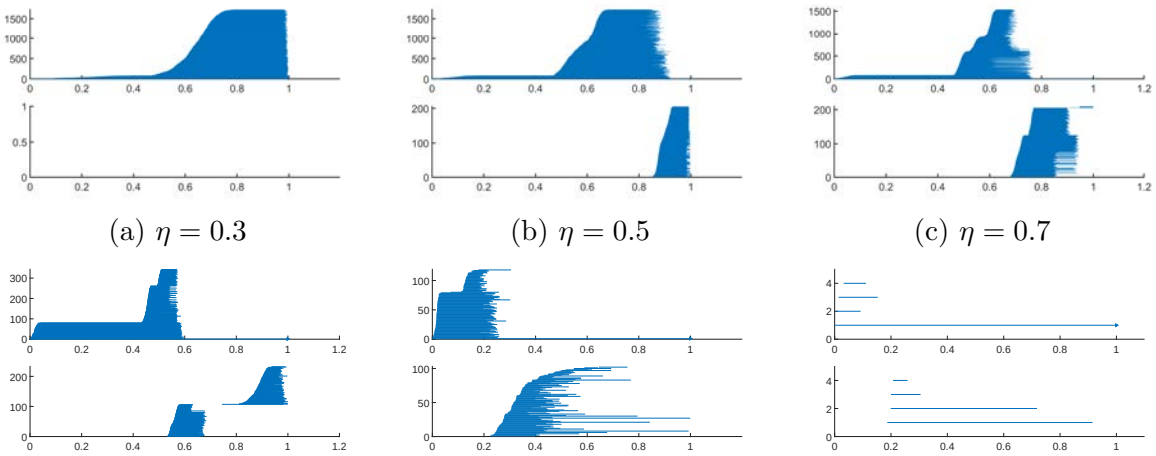


(a) $\eta = 0.7$ (b) $\eta = 1.0$



(c) $\eta = 2.0$ (d) $\eta = 4.0$

Figure 2.12: Illustration of multiresolution geometric analysis of the RNA of 4QG3 structure. The rigidity functions $\mu(\mathbf{r})$ are constructed from various η values. From (a) to (d), η values are set to 0.7, 1.0, 2.0, 4.0, respectively.

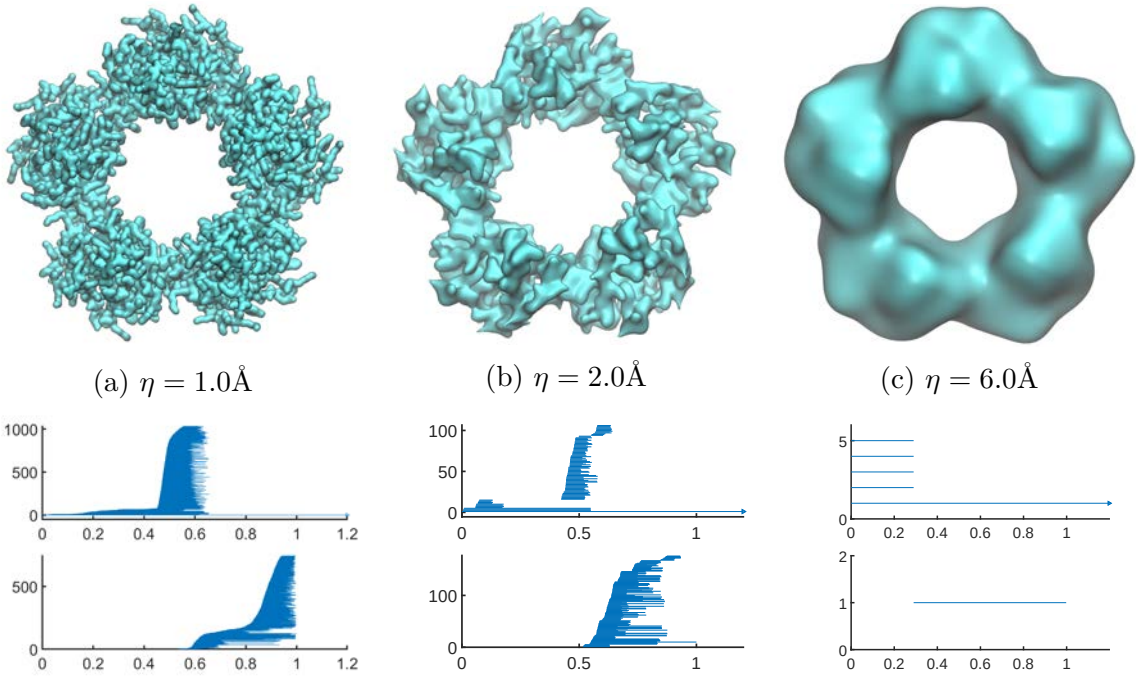


(a) $\eta = 0.3$ (b) $\eta = 0.5$ (c) $\eta = 0.7$

(d) $\eta = 1.0$ (e) $\eta = 2.0$ (f) $\eta = 4.0$

Figure 2.13: Illustration of multiresolution geometric analysis of the RNA of 4QG3 structure. The rigidity functions $\mu(\mathbf{r})$ are constructed from various η values. From (a) to (d), η values are set to 0.7, 1.0, 2.0, 4.0, respectively.

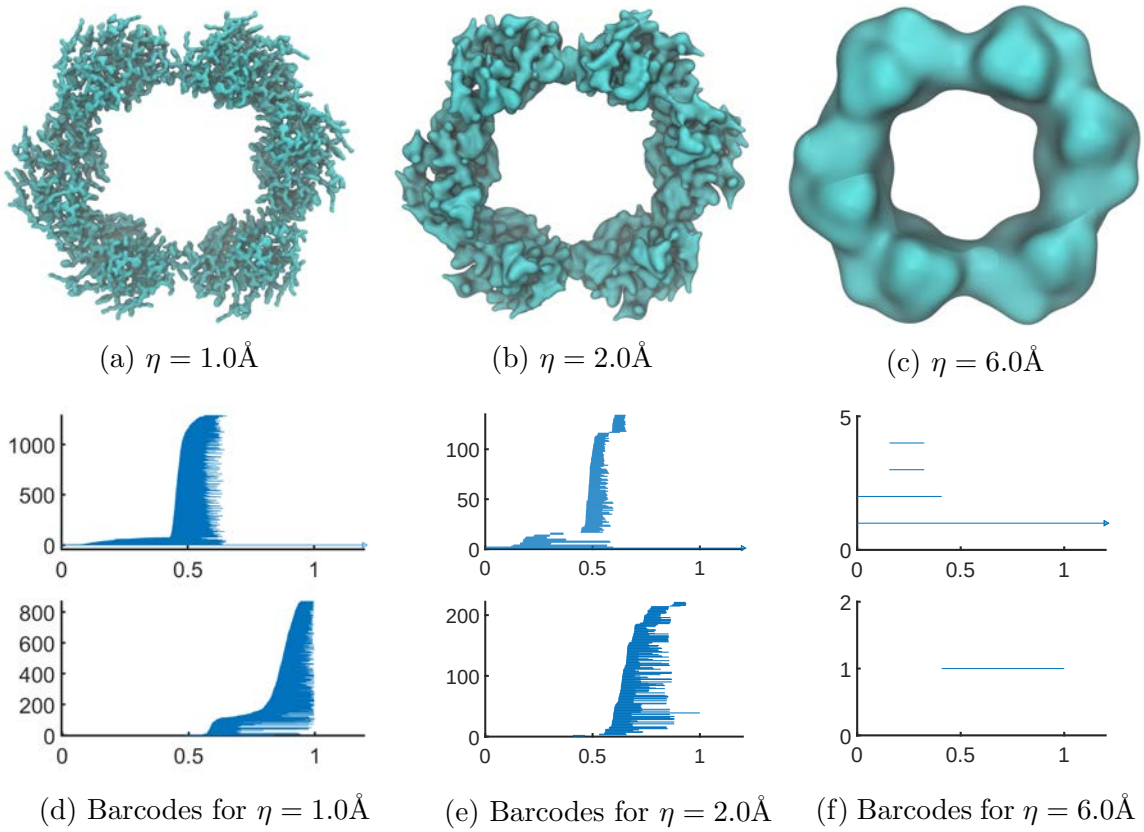
The last example shown in [5, 6] is the virus capsid structure 1DYL. It consists of 5705 atoms in its pentagon-shaped proteins and 6844 atoms each in its hexagonal-shaped proteins. It contains a total of 273780 atoms which makes the multiresolution persistence of whole structure computationally expensive under a single topological representation. However, the PDB contains structural information for single proteins and VMD/Chimera/Biovia has symmetric operations to generate the whole structure. Figure 2.14 and 2.15 shows the multiresolution persistence for a single pentagon-shaped complex and a single hexagonal-shaped complex. Similar to [5, 6], we set the grid size to 0.6Åfor the density map constructions for the single pentagon-shaped complex and the single hexagonal-shaped complex. The density maps are also being rescaled by using Equation (2.8) but barcodes with length of less than 0.05 are removed due to noise. This is because the barcodes in this structure become more noisy when the grid size becomes larger than 0.6Å. In Figure 2.14, the barcodes decreases to 5 symmetric β_0 bars in (f) as η increases. This shows that as the resolution gets poorer, the structure starts to form a symmetric pentagonal structure in (c). However, in Figure 2.15, the β_0 bars do not decrease to 6 symmetric bars. Hence, this shows the hexgonal structure is rather asymmetric as there are only 4 β_0 bars in (f), with two appearing earlier and two appearing later in filtration [5, 6]. The multiresolution persistence and 2D persistent homology of the whole structure can be found in [5, 6].



(a) $\eta = 1.0$Å (b) $\eta = 2.0$Å (c) $\eta = 6.0$Å

(d) Barcodes for $\eta = 1.0$Å (e) Barcodes for $\eta = 2.0$Å (f) Barcodes for $\eta = 6.0$Å

Figure 2.14: Multiresolution analysis of the pentagon-shaped protein in 1DYL. The rigidity functions $\mu(\mathbf{r})$ are constructed for $\eta = 1.0$, 2.0 and 6.0Å. The corresponding persistent barcodes are also shown respectively from (d)-(f).



(a) $\eta = 1.0$Å (b) $\eta = 2.0$Å (c) $\eta = 6.0$Å

(d) Barcodes for $\eta = 1.0$Å (e) Barcodes for $\eta = 2.0$Å (f) Barcodes for $\eta = 6.0$Å

Figure 2.15: Multiresolution analysis of the hexagon-shaped protein in 1DYL. The rigidity functions $\mu(\mathbf{r})$ are constructed for $\eta = 1.0$, 2.0 and 6.0Å. The corresponding persistent barcodes are also shown respectively from (d)-(f).

Chapter 3

Elastic Network Models

In this chapter, we discuss the use of elastic network models (ENMs) in normal mode analysis (NMA) of large biomolecules. This can also found in [1] where the new Multi-scale Virtual Particle (MVP) ENMs showed consistency with the traditional ENMs such as Gaussian Network Models (GNMs) and Anisotropic Network Models (ANMs). The GNM and ANM will first be studied followed by the extension of MVP-ENM to large biomolecules.

3.1 Gaussian Network Model

Definition 3.1.1. For a biomolecule that consists of N number of C_α atoms with coordinates $\mathbf{r}_1, \mathbf{r}_2, \cdots \mathbf{r}_N$, let r_c be the specified cutoff distance between every two C_α atoms of a biomolecule. The Laplacian matrix $\mathbf{L}$ is an $\mathbf{N}$ by $\mathbf{N}$ matrix that can be expressed as

$$\mathbf{L}_{ij} = \begin{cases} -1, & i \neq j \text{ and } r_{ij} \leq r_c. \\ 0, & i \neq j \text{ and } r_{ij} > r_c. \\ -\sum_{i \neq j}^{\mathbf{N}} \mathbf{L}_{ij}, & i = j. \end{cases}$$

Using the Laplacian matrix $\mathbf{L}$, we can compute the equilibrium correlation between fluctuations by the following equation:

$$\langle \Delta\mathbf{r}_i \cdot \Delta\mathbf{r}_j \rangle = \frac{3k_BT}{\gamma}(\mathbf{L}^{-1})_{ij}, \quad \forall i = 1, 2, \cdots, N. \tag{3.1}$$

where $\mathbf{L}^{-1}$ is the Moore-Penrose pseudo-inverse of $\mathbf{L}$. This is due to the first eigenvalue of $\mathbf{L}$ being zero, hence $\mathbf{L}$ is not invertible. However, we only require the diagonal entries of $\mathbf{L}^{-1}$ to calculate the predicted B-factors. In fact, the predicted B-factors are computed by the following formula:

$$B_i^{GNM} = \frac{8\pi^2}{3}\langle \Delta\mathbf{r}_i \cdot \Delta\mathbf{r}_i \rangle = \frac{8\pi^2 k_BT}{\gamma}(\mathbf{L}^{-1})_{ii}. \tag{3.2}$$

where we can get the values of $(\mathbf{L}^{-1})_{ii}$ via the eigenvalues and eigenvectors:

$$(\mathbf{L}^{-1})_{ii} = \sum_{k=2}^{N} \lambda_k^{-1}[v_k v_k^T]_{ii}. \tag{3.3}$$

3.2 Anisotropic Network Model

Previously, we discussed the Gaussian Network Models which allows us to use the pseudo-inverse of the Laplacian to predict B-factors of alpha carbons in biomolecules. In order to consider molecular dynamics, we implement the Anisotropic Network Models which allow us to perform Normal Mode Analysis (NMA) to analyse the molecular dynamics using the eigenmodes.

Definition 3.2.1 (Harmonic Potential)**.** Let $\mathbf{r}_i = (x_i, y_i, z_i)$, $\mathbf{r}_i^d = (x_i^d, y_i^d, z_i^d)$ for $i = 1, 2, \cdots, N$. The Harmonic Potential, also known as the ANM potential function is defined as the following:

$$V^{\text{ANM}} = \gamma \sum_{1 \le i \le j \le N} (r_{ij}^d - r_{ij})^2 f(r_{ij}) \approx \frac{\gamma}{2} \Delta \mathbf{R}^T \mathbf{H} \Delta \mathbf{R}. \tag{3.4}$$

where $f(r_{ij})$ is the same Heavside function defined in the earlier section and $\mathbf{H}$ is the $3N \times 3N$ Hessian matrix.

Compared to the GNM, a larger cutoff distance is used to yield generally better results [1]. The cutoff distance r_c in ANM is chosen to be between 10Å to 15Å. The Hessian matrix is then computed by second partial derivatives of $V_{ij}^{\text{ANM}} := V_{ij}$ as follows:

$$\frac{\partial^2 V_{ij}}{\partial x_i \partial y_j} = -\frac{1}{r_{ij}^2}(x_j - x_i)(y_j - y_i), \quad \forall i, j = 1, 2, \cdots, N, i \neq j \text{ and } r_{ij} \le r_c \tag{3.5}$$

and

$$\frac{\partial^2 V_{ij}}{\partial x_i \partial x_j} = -\frac{1}{r_{ij}^2}(x_j - x_i)^2, \quad \forall i, j = 1, 2, \cdots, N, i \neq j \text{ and } r_{ij} \le r_c. \tag{3.6}$$

Hence, we have

$$\mathbf{H}_{ij} = -\frac{1}{r_{ij}^2} \begin{bmatrix} (x_j - x_i)^2 & (x_j - x_i)(y_j - y_i) & (x_j - x_i)(z_j - z_i) \\ (y_j - y_i)(x_j - x_i) & (y_j - y_i)^2 & (y_j - y_i)(z_j - z_i) \\ (z_j - z_i)(x_j - x_i) & (z_j - z_i)(y_j - y_i) & (z_j - z_i)^2 \end{bmatrix}. \tag{3.7}$$

Similar to the Laplacian matrix in GNM, we assign the diagonal entries $\mathbf{H}_{ii}$ as the negative sums of the off diagonal 3×3 block matrices for ith column.

$$\text{i.e. } \mathbf{H}_{ii} = -\sum_{i \neq j} \mathbf{H}_{ij}, \quad \forall i = 1, 2, \cdots, N. \tag{3.8}$$

Similar to the GNM, we compute the equilibrium correlation between fluctuations by the following equation

$$\langle \Delta \mathbf{R}_i \cdot \Delta \mathbf{R}_j \rangle = \frac{3k_B T}{\gamma} (\mathbf{H}^{-1})_{ij}, \quad \forall i = 1, 2, \cdots, N. \tag{3.9}$$

where $\mathbf{H}^{-1}$ is also the pseudo-inverse of $\mathbf{H}$. Similar to the GNM once again, the Hessian matrix $\mathbf{H}$ here has rank $3N - 6$ hence it is not invertible. This is because the first 6

eigenvalues are zero due to the rigid body motions of the biomolecule. Hence, we consider the pseudo inverse

$$(\mathbf{H}^{-1})_{ii} = \sum_{k=7}^{3N} \lambda_k^{-1} [v_k v_k^T]_{ii}, \tag{3.10}$$

where $i = 7$ to N represents the index of non-trivial eigenmodes. The eigenvectors of these eigenmodes describe the vibrational direction and the relative amplitude in the different modes. Therefore, for the ANM, the predicted B-factors is

$$B_i^{ANM} = \frac{8\pi^2}{3} \sum_{j=3i-2}^{3i} \langle \Delta\mathbf{R}_i \cdot \Delta\mathbf{R}_j \rangle, \quad \forall i = 1, 2, \cdots, N. \tag{3.11}$$

3.3 MVP Model

The MVP model is an application of the multiscale topological representation of biomolecules in the earlier sections. Recall that the multiscale rigidity index of the point cloud data to be

$$\mu(\mathbf{r}, \eta) = \sum_{j=1}^{N} w_j \Phi(||\mathbf{r} - \mathbf{r}_j||; \eta), \tag{3.12}$$

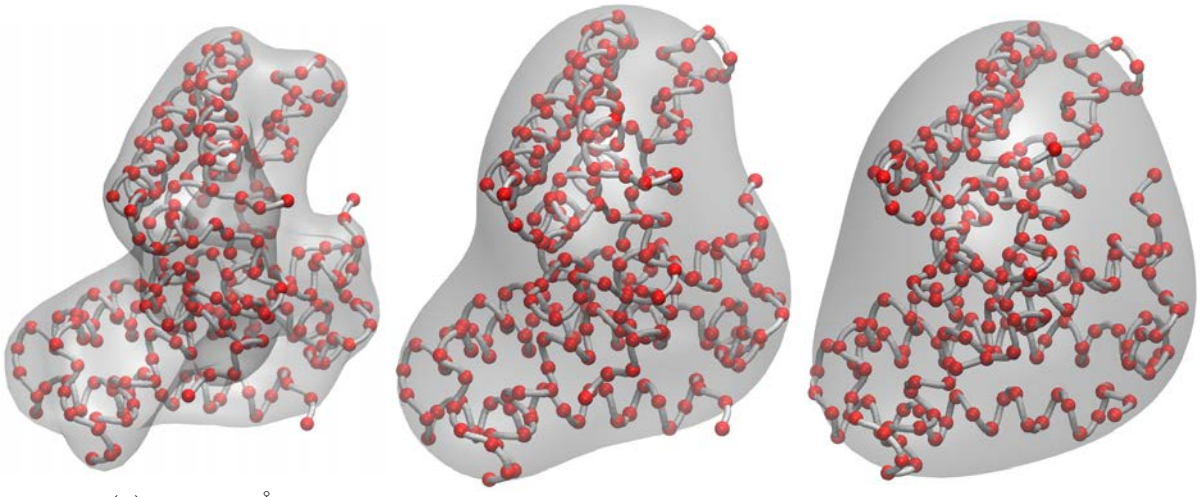
where w_j is the weight parameter of the jth atom and η is the same resolution parameter. For example, in the multiresolution topological analysis in the previous section, w_j is considered as the atomic number of the jth atom. For the MVP model, we only take into account the C_α coarse grained representation of biomolecules (i.e. only alpha carbons are extracted from the PDB.) and $w_j = 1$ for all $1 \leq j \leq N$. In the MVP model, our kernel function is still the generalized exponential kernel used in earlier sections.

$$\text{i.e. } \Phi(||\mathbf{r} - \mathbf{r}_j||; \eta, \kappa) = e^{-(||\mathbf{r}-\mathbf{r}_j||/\eta)^\kappa}, \quad \kappa > 0. \tag{3.13}$$

However, in this model, we rescale our density function in Equation (3.12) by

$$\mu(\mathbf{r}; \eta)^s = \frac{\mu(\mathbf{r}) - \mu_{\min}}{\mu_{\max} - \mu_{\min}}, \tag{3.14}$$

where $\mu_{\min}$ and $\mu_{\max}$ are the maximum and minimum of $\mu(\mathbf{r})$. In this section, we use the protein 2CCY as our main example for all the models. As mentioned previously, only the C_α atoms are extracted needed. we first visualize the density data of MVP model using the Gaussian kernel in Equation (3.13) with $\kappa = 2$ and 3 different η values, namely, $\eta = 5$Å, 10Å, 15Å. Figure 3.1 shows the biomolecular surfaces of 2CCY based on the 3 different η values.



(a) $\eta = 5.0$ Å (b) $\eta = 10.0$ Å (c) $\eta = 15.0$ Å

Figure 3.1: Illustration of density data of Protein 2CCY generated by the Gaussian Kernel with $\kappa = 2$ and $\eta = 5.0$ Å, 10.0 Å and 15.0 Å respectively. Isovalue is adjusted to 0.4.

Now, we can explain the rest of the MVP model. The rest of the MVP model is based on the idea of using a multiscale virtual particle model which means instead of directly applying the GNM or ANM to the atoms in the biomolecules, we construct a set of virtual particles based on the density data of biomolecule with a certain isovalue and then apply the ENM model. In this case, the virtual particles are constructed by the density data of the biomolecule. For a certain threshold range, we consider a virtual particle to be in the 2D biomolecular surface (see [1]). This approach allows the use of parameters such as η and C_α coarse grained representations which will vary the accuracies due to the structural representations. As the virtual particles are dependent on their structural representations, the spring parameter γ used in traditional ENMs is not suitable [1]. Hence, it was proposed in [1] that the new spring parameter would be

$$\gamma_{IJ} = \gamma(r_I, r_J, \Omega_I, \Omega_J, \mu(\mathbf{r})^s, \eta^{\text{MVP}}) = \gamma_1(\Omega_I, \Omega_J, \mu(\mathbf{r})^s) \cdot \gamma_2(\mathbf{r}_I, \mathbf{r}_J, \eta^{\text{MVP}}) \tag{3.15}$$

where $\mathbf{r}_I$, $\mathbf{r}_J$ are the centers of the I-th and J-th virtual particle respectively, Ω_I and Ω_J are the regular and irregular grid space respectively. Irregular grid spaces relates to the virtual particles near the boundary of the 2D density map which might be of irregular shape. The parameter γ_1 is part of the spring parameter that takes into account the mass or density contribution and γ_2 takes into account of the distance between any two virtual particles. For γ_1, in order to consider the contribution of mass or density, we integrate its density over the regular and irregular grid space by the following equation

$$\gamma_1(\Omega_I, \Omega_J, \mu(\mathbf{r})^s) = \left(1 + a \int_{\Omega_I} \mu(\mathbf{r})^s \, dr\right)\left(1 + b \int_{\Omega_J} \mu(\mathbf{r})^s \, dr\right), \tag{3.16}$$

where a and b are constants. In [1], the constants a and b were set to $\frac{1}{|\Omega_I|}$ and $\frac{1}{|\Omega_J|}$ respectively. By approximating both the regular and irregular virtual particles to a regular virtual particle under the entire grid space, we get the following approximation shown in [1].

$$\gamma_1(\Omega_I, \Omega_J, \mu(\mathbf{r})^s) \approx (1 + \mu(\mathbf{r}_I)^s)(1 + \mu(\mathbf{r}_J)^s). \tag{3.17}$$

In this case, the paper sets all the virtual particle centers r_I and r_J onto Cartesian grid points. As for γ_2, the term was chosen to be the generalised gaussian kernel in Equation

(3.13).

$$\gamma_2(\mathbf{r}_I, \mathbf{r}_J, \eta^{\text{MVP}}) = e^{-(||\mathbf{r}-\mathbf{r}_j||/\eta^{\text{MVP}})^\kappa}, \quad \kappa > 0. \tag{3.18}$$

With the MVP model, this allows us to introduce the MVP-GNM and MVP-ANM [1] in the section 4.4.

3.4 MVP-ENM

For the MVP-GNM, the new potential function would be defined as

$$V^{\text{MVP-GNM}} = \frac{1}{2}\Delta\mathbf{r}^T\mathbf{L}^{\text{MVP-GNM}}\Delta\mathbf{r}, \tag{3.19}$$

where $\mathbf{L}^{\text{MVP-GNM}}$ is the new Laplacian matrix that includes the new spring parameter γ_{IJ}. In fact,

$$\mathbf{L}^{\text{MVP-GNM}} = \begin{cases} -\gamma(r_I, r_J, \Omega_I, \Omega_J, \mu(\mathbf{r})^s, \eta^{\text{MVP}}) & I \neq J \\ -\sum_{I \neq J} \mathbf{L}_{IJ}^{\text{MVP-GNM}} & I = J \end{cases} \tag{3.20}$$

The rest of the formulas for the predicted B-factors and equilibrium correlations follows similarly as in section 4.1. However, notice that in MVP-GNM, the predicted B-factors would be calculated for the virtual particles and not the original atoms of biomolecule. In order to conduct comparison with the traditional GNM, the nearest neighbour interpolation method was used to interpolate the predicted values based on atoms coordinates. Essentially, in python, one could implement it using the single line of code:

```
griddata(vp_coords, test_inv, coords, method = "nearest")
```

Similar to Figure 3.1, we choose $\kappa = 2$ and test the MVP-GNM based on the 3 η values (5Å, 10Å and 15Å). The grid size is chosen to be 2.0Å and the normalized density data is filtered with $\mu(\mathbf{r})^s \geq 0.4$. The η^{MVP} is also chosen to be same as η. Figure 3.2 shows the comparison of GNM with MVP-GNM of 3 different η values. Figures 3.3 and 3.4 serve as similar comparisons for two other proteins, namely, 2ABH and 1AQB. Essentially, the results of MVP-GNM showed that with only the coarse grained representation and $\eta = 5\mathring{A}$ (i.e. finest scale resolution), the predicted B-factors are still strongly correlated to the experimental B-factors. The inaccuracies of the predicted B-factors can be attributed to the regions with extremely high B-factors [1]. Even at poorer resolutions such as $\eta = 10\mathring{A}$ and $15\mathring{A}$, the plots shows that the MVP-GNM still preserves the basic patterns of the B-factors [1]. Note that the cutoff distance differs the performance of PCCs for different proteins. Hence, it is not true that low η value results in higher PCC.

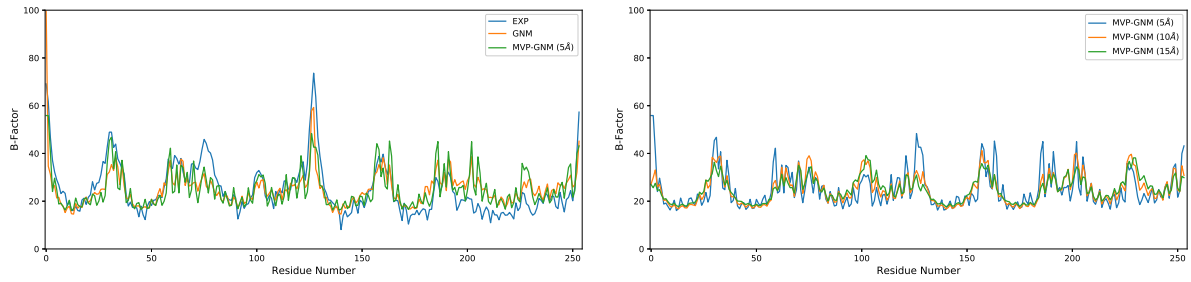


Figure 3.2: Comparison between B-Factor Prediction for 2CCY MVP-GNM Model, GNM Model and the Experimental B-Factors. Pearson Correlation Coefficient (PCC) for GNM is 0.739. PCCs for MVP-GNM with $\eta = 5\mathring{A}$, $10\mathring{A}$, $15\mathring{A}$ are 0.737, 0.583 and 0.484, respectively.

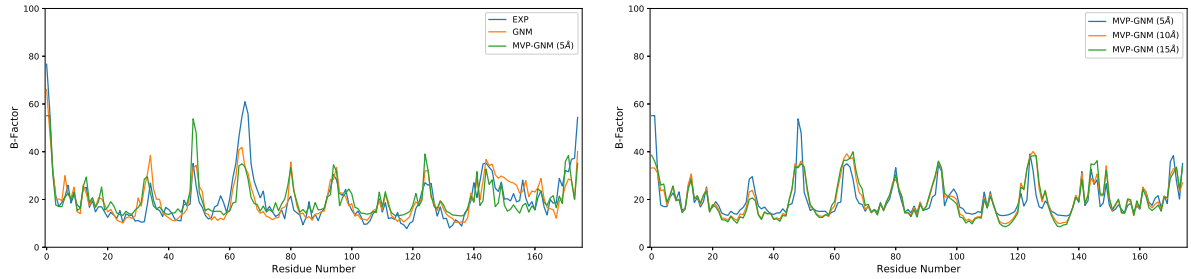


Figure 3.3: Comparison between B-Factor Prediction for 1AQB MVP-GNM Model, GNM Model and the Experimental B-Factors. PCC for GNM is 0.822. PCCs for MVP-GNM with $\eta = 5\mathring{A}$, $10\mathring{A}$, $15\mathring{A}$ are 0.742, 0.722 and 0.770, respectively.

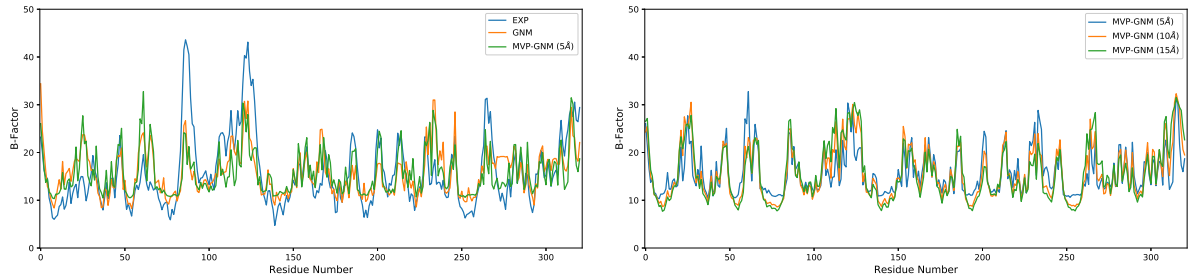


Figure 3.4: Comparison between B-Factor Prediction for 2ABH MVP-GNM Model, GNM Model and the Experimental B-Factors. PCC for GNM is 0.647. PCCs for MVP-GNM with $\eta = 5\mathring{A}$, $10\mathring{A}$, $15\mathring{A}$ are 0.634, 0.741 and 0.796, respectively.

For the MVP-ANM, the new potential function would then be

$$V^{\text{MVP-ANM}} = \frac{1}{2}\Delta\mathbf{R}^T\mathbf{H}^{\text{MVP-ANM}}\Delta\mathbf{R}, \tag{3.21}$$

where $\mathbf{H}^{\text{MVP-ANM}}$ is Hessian matrix with 3x3 off-diagonal matrices

$$\mathbf{H}_{IJ} = -\frac{\gamma_{IJ}}{r_{IJ}^2}\begin{bmatrix} (x_J - x_I)^2 & (x_J - x_I)(y_J - y_I) & (x_J - x_I)(z_J - z_I) \\ (y_J - y_I)(x_J - x_I) & (y_J - y_I)^2 & (y_J - y_I)(z_J - z_I) \\ (z_J - z_I)(x_J - x_I) & (z_J - z_I)(y_J - y_I) & (z_J - z_I)^2 \end{bmatrix}, \quad I \neq J, \tag{3.22}$$

and diagonal matrices

$$\mathbf{H}_{II}^{\text{MVP-ANM}} = -\sum_{I \neq J}\mathbf{H}_{IJ}^{\text{MVP-ANM}}. \tag{3.23}$$

The same MVP model spring parameter is used in the MVP-ANM. We follow the same B-factor prediction formulas from Section 4.2.

Remark 3.4.1. Note that MVP-ANM can be directly applied to NMA of Cryo-EM data [1] as compared to the ANM. However, with regards to B-factor prediction, the ANM has poorer results than the GNM.

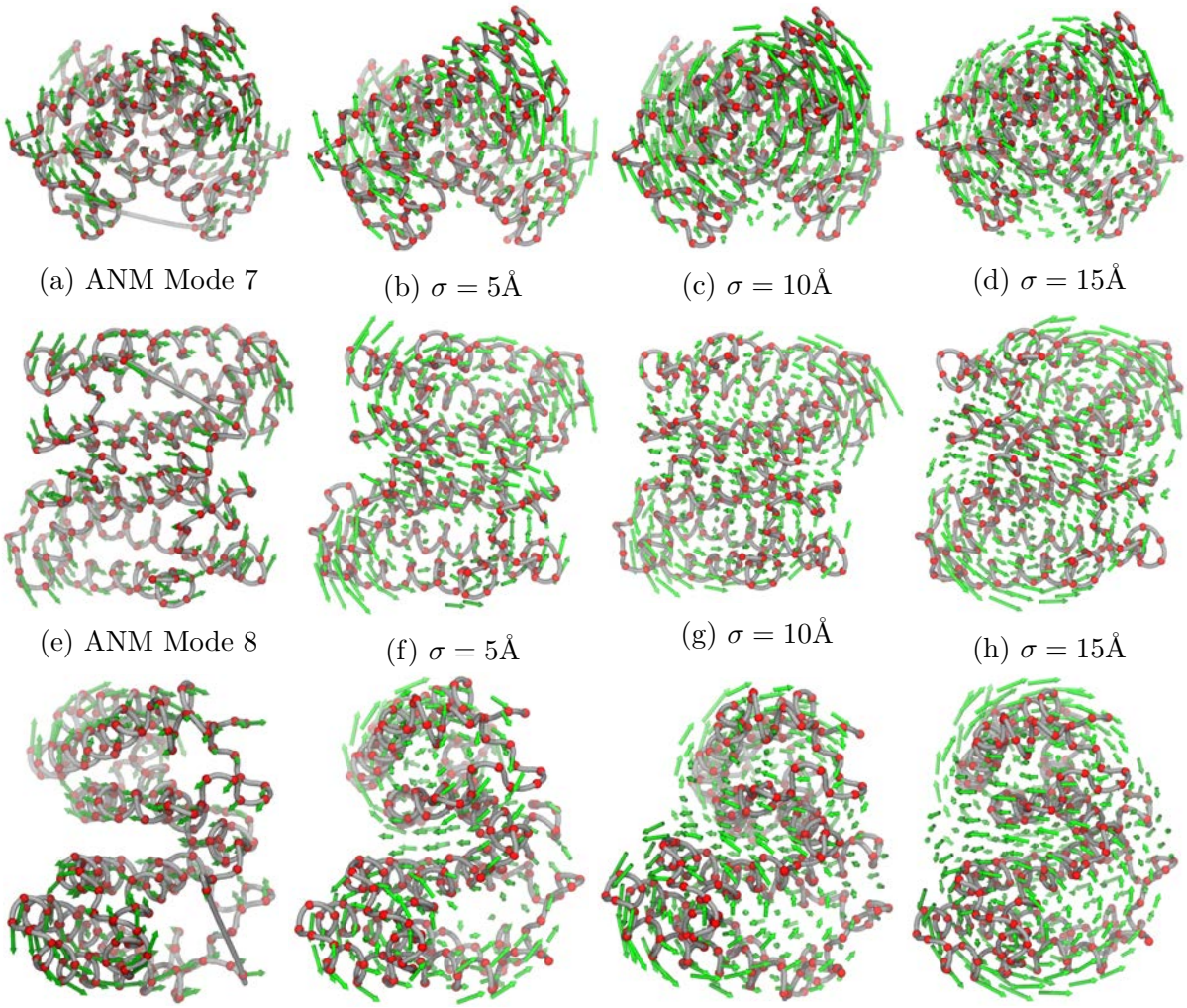
Similar to MVP-GNM, we set $\kappa = 2$ and use the three scale resolutions $\eta = 5\mathring{A}$, 10Å, 15Å. As the Hessian matrix is larger than the Laplacian matrices, we set the grid size to 5Å. Again, we also filter the normalised density values with $\mu(\mathbf{r})^s \geq 0.4$. Similar to the ANM, we observe the non-trivial eigenmodes of $\mathbf{H}^{-1}$ via Equation (3.10) In particular, we perform MVP-ANM on 2CCY. The non-trivial eigenmodes of 7 to 9 of ANM and MVP-ANM are shown in Figure 3.5. We observe that for 2CCY, the eigenmodes for MVP-ANM is highly consistent with the ANM in terms of rotational patterns. This further supports the MVP-GNM in showing that coarse grained representation still has great consistency in preserving the patterns. In [1], the eigenvectors were also reshaped to directional components. This is to perform interpolation of coordinates to compute the PCCs between the ANM eigenvectors and those from MVP-ANM for each eigenmode. In order to compute the PCCs, the eigenvectors are reshaped such that every three values are the X, Y and Z coordinates. Similar to MVP-GNM, we use the nearest neighbour interpolation to interpolate the X, Y and Z coordinates to the atom coordinates. The interpolated coordinates are then compared with the coordinates of ANM to compute PCC with respect to X, Y and Z components separately. Table 3.1 shows the average PCC for each of the three eigenmodes of 2CCY [1].

PDB ID	**Mode 7**	**Mode 8**	**Mode 9**
2CCY	0.880	0.669	0.756

Table 3.1: Average PCCs for Modes 7 to 9 of 2CCY [1]

Although it is shown in [1] that MVP-ANM can help reduce computational complexity for larger biomolecules and can applied in Cryo-EM data, there is a lack of mathematical explanation in why MVP-ANM is consistent in its results. There should be some mathematical reasoning to support the connection between MVP-ANM and molecular dynamics of biomolecules. Nevertheless, MVP-ANM showed that the molecular dynamics are captured even when coarse grained representation and low scale resolution is used.

However, note that for some biomolecules such as 2ABH, its eigenmodes can be irregular and more attention should be paid in order to observe the results correctly (see [1] for greater detail).



(a) ANM Mode 7 (b) $\sigma = 5$Å (c) $\sigma = 10$Å (d) $\sigma = 15$Å

(e) ANM Mode 8 (f) $\sigma = 5$Å (g) $\sigma = 10$Å (h) $\sigma = 15$Å

(i) ANM Mode 9 (j) $\sigma = 5$Å (k) $\sigma = 10$Å (l) $\sigma = 15$Å

Figure 3.5: Illustration of ANM and MVP-ANM for modes 7, 8 and 9 of 2CCY.

Chapter 4

Protein-Ligand Binding

Protein-ligand binding is fundamental to many biological processes in living organisms. The understanding of protein-ligand interactions is essential for drug design and protein design and has been a central issue in molecular biophysics, structural biology, and medicine. There are many factors that affects protein-ligand binding:

- Driven by Binding Energy Reduction. i.e. Intermolecular forces, such as ionic bonds, hydrogen bonds, hydrophobic effects and van der Waals (vdW) interactions.
- Main Focus of [2]: Driven by Flexibility Reductions or Rigidity Enhancements. These are potential mechanisms that have been neglected in current modelling and computations.

In the previous chapters, we reviewed some methods that apply persistent homology to understand underlying topological patterns in biomolecules. Recall that these topological patterns can be used as inputs to predict B-factors and other quantitative biological data. In this chapter, we review [2] in which the paper uses Random Forest and Gradient Boosting Trees to that Rigidity Strengthening models can predict binding affinities for Protein-Ligand binding as well.

4.1 Theory and Methodology

The paper [2] postulates that flexibility reduction or rigidity strengthening plays a unique role in protein-ligand binding. The methodology of the paper uses random forest algorithm for prediction of protein-ligand binding affinities. Recall that the Flexibility-Rigidity Index (FRI) is defined as the following:

Definition 4.1.1 (Flexibility-Rigidity Index (FRI))**.** Consider a bio-molecule having N atoms with coordinates $\{r_i | r_i \in \mathbb{R}^3, i = 1, 2, \cdots, N\}$. We denote $||r_i - r_j||$ as the Euclidean distance between the ith and jth atom. We denote r_i as the vdW radius of the ith atom and set $n_{ij} = \tau(r_i + r_j)$ as a scale to characterize the distance between the ith and jth atoms, where $\tau > 0$ is an adjustable parameter. The Atomic Rigidity Index (ARI) is then expressed as

$$\mu_i = \sum_{j=1}^{N} w_j \Phi_\tau(||r_i - r_j||)$$

and the Flexibility index (FI) is expressed as

$$f_i = \frac{1}{\mu_i}$$

where w_j are the particle-type dependent weights and are initialized as 1. Here Φ is a real-valued monotonically decreasing correlation function satisfying

- $\Phi(||r_i - r_j||) = 1$, as $||r_i - r_j|| \to 0$.
- $\Phi(||r_i - r_j||) = 0$, as $||r_i - r_j|| \to \infty$.

Again, we make use of the two commonly used FRI correlation functions.

- Generalized Exponential Functions

$$\Phi^E_{\kappa,\tau}(||r_i - r_j||) = e^{-(||r_i - r_j||/n_{ij})^\kappa}, \quad \kappa > 0.$$

- Generalized Lorentz Functions

$$\Phi^L_{\upsilon,\tau}(||r_i - r_j||) = \frac{1}{1 + (||r_i - r_j||/n_{ij})^\upsilon}, \quad \upsilon > 0.$$

In order to measure the element-specific protein-ligand rigidity index, we collect cross correlations by defining the following:

Definition 4.1.2 (Rigidity Index Based Scoring Functions (RI-Score))**.** Let α be the kernel index indicating either the exponential kernel ($\alpha = E$) or Lorentz kernel ($\alpha = L$). Correspondingly, let β be the kernel order index such that $\beta = \kappa$ when $\alpha = E$ and $\beta = \upsilon$ when $\alpha = L$. We also define a cutoff distance c for the vdW distance between the kth atom $\in X \in$ Protein and lth atom $\in Y \in$ Ligand where $X = \{C, N, O, S\}$ and $Y = \{C, N, O, S, P, F, Cl, Br, I\}$. Then we have

$$RI^\alpha_{\beta,\tau,c}(X - Y) = \sum_{k \in X \in \text{Pro}} \sum_{l \in Y \in \text{Lig}} \Phi^\alpha_{\beta,\tau}(||r_k - r_l||), \quad \forall ||r_k - r_l|| \leq c.$$

Hence, by the types of atoms in X and Y, we would have $9 \times 4 = 36$ distinct atom-atom combinations. In [2], three databases were used: PDBBind v2007, v2013 and v2016. In the implementation, the RI-Score was computed and appended to the quantitative values for each atom-atom combination. For example, for every $C - C$ combination, we sum up all the RI-Scores. Therefore, each vector generated for each complex in PDBBind dataset should be of size $36 \times 1|$. However, as an improved model, the use of multiscale just like in the previous chapters helped to improve their results. The multiscale is applied by considering various values of τ and constructing vectors of size $36 \times$ `len(\tau)` as a input vector to their model. For instance, by considering 3 different values of τ, the vector would be of size 108. This allows their model to train the input values from 3 different measurements. As we observe in [2], the application of multiscale once again showed that the model is able to beat the correlation coefficients of all the past results in 2017. The next section shows a series of plots by repeating the model in the paper.

4.2 Results

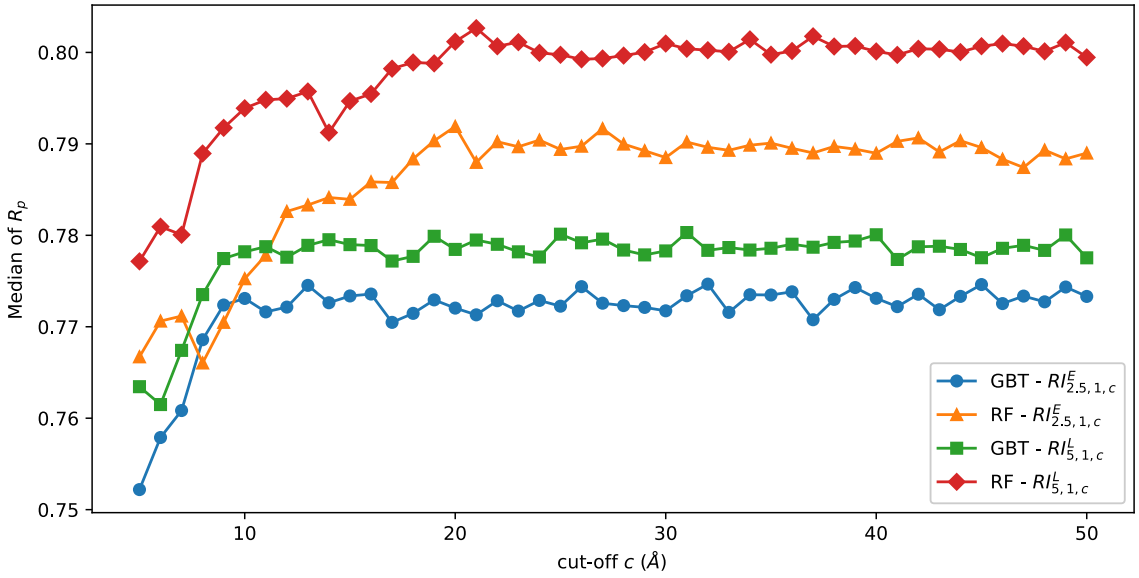
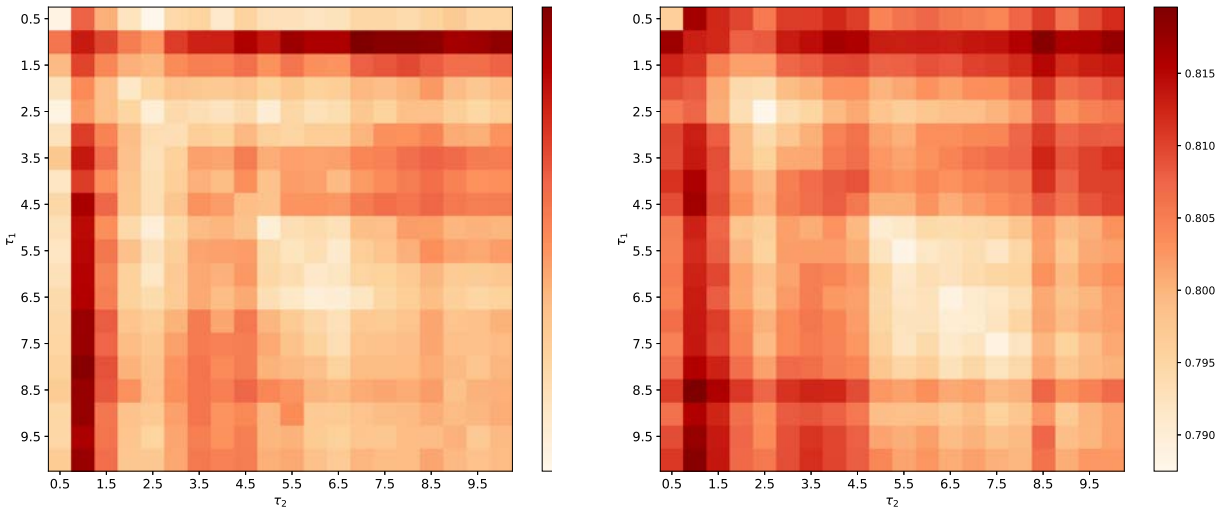


Figure 4.1: R_p values of $RI^E_{2.5,1,c}$ and $RI^L_{5,1,c}$ via GBT and RF



(a) (b)

Figure 4.2: Illustration of multiscale behaviour in Protein-Ligand Binding Predictions. Median of R_p values of 3-scale models on PDBBind v2007 core set are plotted against different values τ_1 and τ_2: (a) $RI^{EEE}_{2.5,\tau_1,c_1;2.5,\tau_2,c_2;40,5.5,18}$ using Random Forest; (b) $RI^{EEE}_{2.5,\tau_1,c_1;2.5,\tau_2,c_2;40,5.5,18}$ using Gradient Boosted Trees. τ_1 and τ_2 are varied from 0.5 to 20 with stepsize of 0.5 and cutoff distance c_i is $\max(12, 3.7\tau_i)$ for $i = 1, 2$.

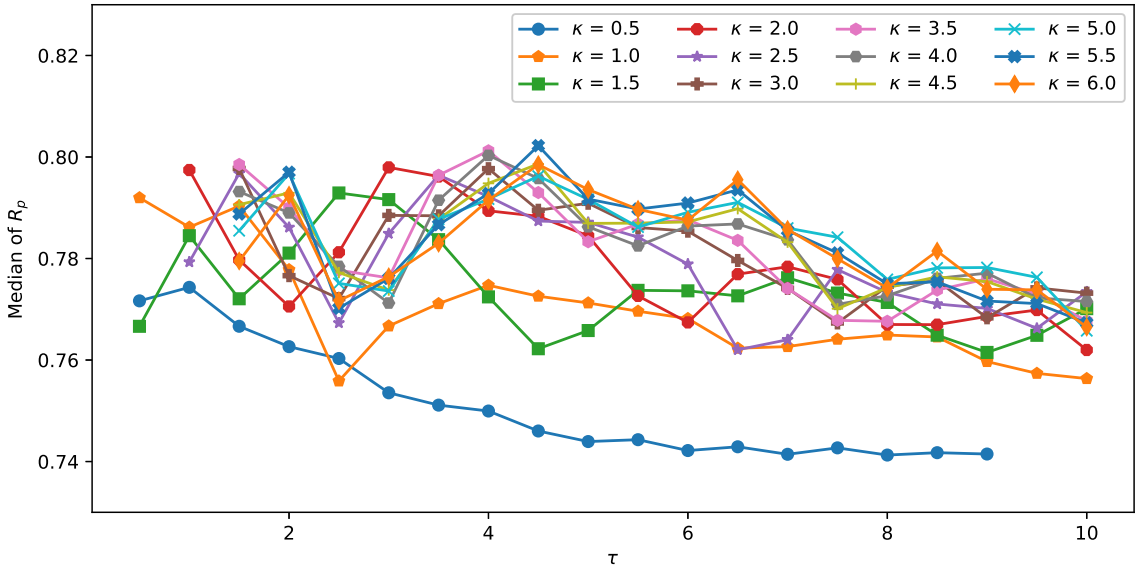


Figure 4.3: Pearson correlation coefficients (R_p) of $RI^E_{\kappa,\tau,40}$ are plotted against the choice of τ for PDBBind v2007 core set over a range of κ values using GBT

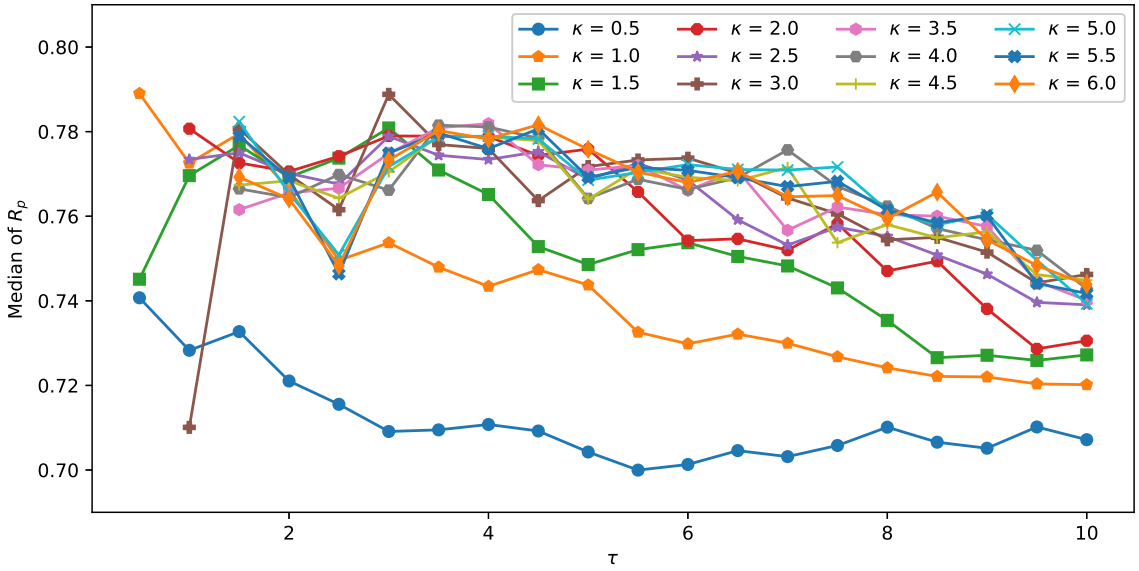


Figure 4.4: Pearson correlation coefficients (R_p) of $RI^E_{\kappa,\tau,40}$ are plotted against the choice of τ for PDBBind v2007 core set over a range of κ values using RF

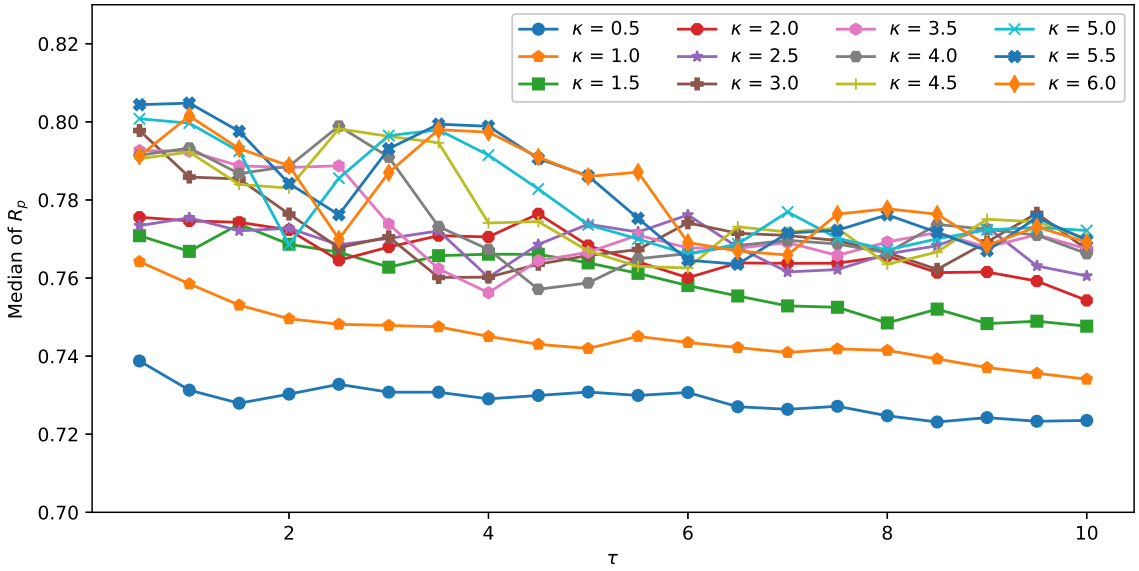


Figure 4.5: Pearson correlation coefficients (R_p) of $RI^L_{\kappa,\tau,40}$ are plotted against the choice of τ for PDBBind v2007 core set over a range of κ values using GBT

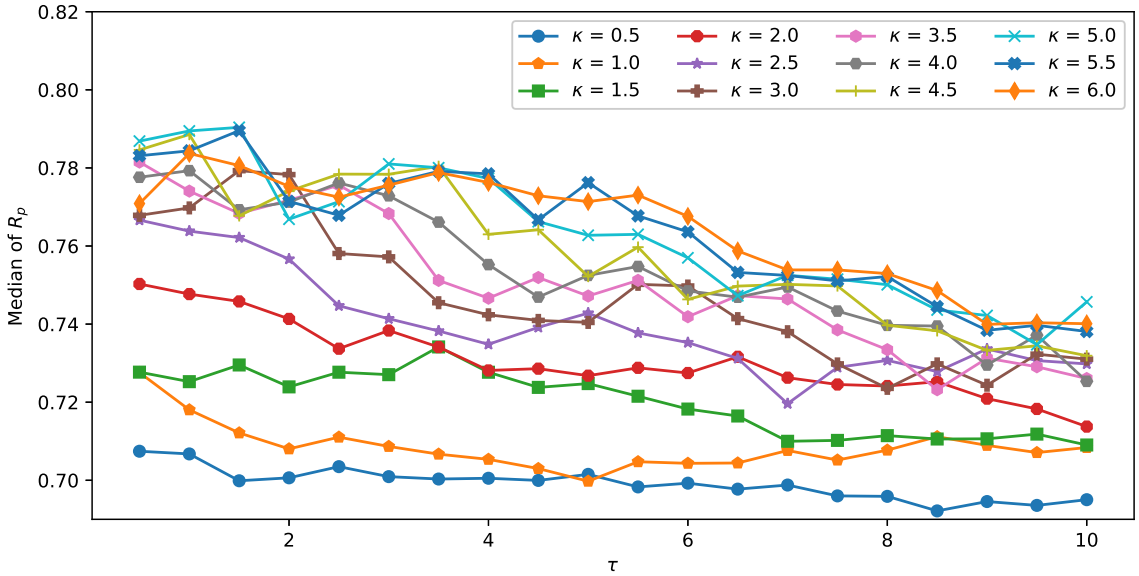


Figure 4.6: Pearson correlation coefficients (R_p) of $RI^L_{\kappa,\tau,40}$ are plotted against the choice of τ for PDBBind v2007 core set over a range of κ values using RF

Bibliography

[1] Kelin Xia, "Multiscale virtual particle based elastic network model (MVP-ENM) for biomolecular normal mode analysis" , Physical Chemistry Chemical Physics, 20(1), 658-669 (2018)

[2] Nguyen, D. D., Xiao, T., Wang, M., & Wei, G. W. (2017). Rigidity Strengthening: A Mechanism For Protein–ligand Binding. Journal of Chemical Information and Modeling, 57(7), 1715-1721.

[3] Nguyen, D. D., Xia, K., & Wei, G. W. (2016). Generalized flexibility-rigidity index. The Journal of chemical physics, 144(23), 234106.

[4] Kelin Xia and Guo-Wei Wei, "Persistent homology for cryo-EM data analysis", International Journal for Numerical Methods in Biomedical Engineering, 31(8), e02719(2015).

[5] Kelin Xia and Guo-Wei Wei, "Multiresolution topological simplification", Journal of Computational Biology, 22(9), 1-5 (2015).

[6] Xia, K., Zhao, Z., & Wei, G. W. (2015). Multiresolution topological simplification. arXiv preprint arXiv:1504.00033.

[7] Kelin Xia, Xin Feng, Yiying Tong and Guo-Wei Wei, "Persistent homology for the quantitative prediction of fullerene stability", Journal of Computational Chemistry, 36, 408-422(2015).

[8] Opron, K., Xia, K., & Wei, G. W. (2014). Fast and anisotropic flexibility-rigidity index for protein flexibility and fluctuation analysis. The Journal of chemical physics, 140(23), 06B617_1.

[9] Xia, K., & Wei, G. W. (2014). Persistent homology analysis of protein structure, flexibility, and folding. International journal for numerical methods in biomedical engineering, 30(8), 814-844.

[10] Zomorodian, A. J. (2001). Computing and comprehending topology: Persistence and hierarchical morse complexes. University of Illinois at Urbana-Champaign.

[11] Konstantin Mischaikow and Vidit Nanda. Morse Theory for Filtrations and Efficient Computation of Persistent Homology. Discrete & Computational Geometry, Volume 50, Issue 2, pp 330-353, September 2013.

[12] Maria, Clément, Jean-Daniel Boissonnat, Marc Glisse, and Mariette Yvinec. "The gudhi library: Simplicial complexes and persistent homology." In International Congress on Mathematical Software, pp. 167-174. Springer, Berlin, Heidelberg, 2014.

[13] Adams, Henry, Andrew Tausz, and Mikael Vejdemo-Johansson. "JavaPlex: A research software package for persistent (co) homology." In International Congress on Mathematical Software, pp. 129-136. Springer, Berlin, Heidelberg, 2014.