A COMPARISON OF CLUSTERING ALGORITHMS FOR THE STUDY OF ANTIBODY LOOP STRUCTURES

A Dissertation
Submitted to
the Temple University Graduate Board

In Partial Fulfillment
of the Requirements for the Degree
MASTER OF SCIENCE

---

by
Benjamin H. North
Diploma Date August 2017

Examining Committee Members:

Zoran Obradovic, Advisory Chair, CIS
Eduard Dragut, CIS
Slobodan Vucetic, CIS
Qiang Zeng, CIS
Roland Dunbrack, External Member, Fox Chase Cancer Center

ABSTRACT

Antibodies are the fundamental agents of the immune system. The CDRs, or Complementarity Determining Regions act as the functional surfaces in binding antibodies to their targets. These CDR structures, which are peptide loops, are diverse in both amino acid sequence and structure. In 2011, we performed a survey of CDR loop structures using the affinity propagation clustering algorithm of Frey and Dueck. With the growth of the number of structures deposited in the Protein Data Bank, the number of antibody CDRs has approximately tripled. In addition, although the affinity clustering in 2011 was successful in many ways, the methods used left too much noise in the data, and the affinity clustering algorithm tended to clump diverse structures together. This work revisits the antibody CDR clustering problem and uses five different clustering algorithms to categorize the data. Three of the clustering algorithms use DBSCAN but differ in the data comparison functions used. One uses the sum of the dihedral distances, while another uses the supremum of the dihedral distances, and the third uses the Jarvis-Patrick shared nearest neighbor similarity, where the nearest neighbor lists are compiled using the sum of the dihedral distances. The other two clustering methods use the k-medoids algorithm, one of which has been modified to include the use of pairwise constraints. Overall, the DBSCAN using the sum of dihedral distances and the supremum of the dihedral distances produced the best clustering results as measured by the average silhouette coefficient, while the constrained k-medoids clustering algorithm had the worst clustering results overall.

ACKNOWLEDGMENTS

I would like to thank my advisors, Roland Dunbrack and Zoran Obradovic, for their helpful discussions and guidance. Also, many thanks go out to the current and former members of the Dunbrack lab, especially Jared Adolf-Bryfogle, for his help in obtaining the antibody database. Also, thanks to Andreas Lehmann for his contributions to the database.

TABLE OF CONTENTS

LIST OF TABLES

Table Page

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Antibodies are a fundamental to the function of the immune system. Each antibody is a protein that binds to a specific antigen for targeting by the immune system (Lodish et al., 2007). The study of antibody structure is an important line of research for many reasons, including for the investigation of the biological processes of the immune system, the repurposing of antibodies as therapeutic agents or as in vitro binding or catalytic agents in the lab, and the study of protein structures in general. Typical antibodies consist of two types of subunits that participate in binding, the heavy chain variable (VH) domain and the light chain variable (VL) domain. These subunits are structurally similar to one another, each consisting of an immunoglobulin fold (Bork, Holm, & Sander, 1994). These variable domains consist of a fairly constant framework (which constitutes the bulk of the domain) and a set of three hypervariable regions per domain. These hypervariable regions, also called the Complementarity Determining Regions (CDRs), are peptide loops that are highly variable in both amino acid sequence and in three-dimensional structure, hence their name. In this work they are called both "CDRs" and "antibody loops" interchangeably. They are responsible for the binding activity of antibodies to each antibody's target, called an epitope. CDRs consist of six different types, three for the light variable chain (CDR types L1, L2, and L3), and three for the heavy variable chain (CDR types H1, H2, and H3).

There are several reasons for revisiting the CDR loop structure categorization problem. One good reason is the increase in the size of the antibody structure database. There were approximately 1300 CDR structures per CDR type in the Protein Data Bank for our 2011 analysis, which is a recent comprehensive analysis; as of January 2017, there are approximately 4,000 structures per CDR type (North, Lehmann, & Dunbrack, 2011). One goal of this project is to categorize as many of the CDR structures available as possible. Also, given the results of the 2011 CDR study, there are still improvements that can be made on the analysis. The outlier removal used for the 2011 study was not strong enough, and so clusters that contain members CDR loops with different conformations are present. This problem of several conformations in a single cluster needs to be resolved. One possibility is a more stringent procedure for outlier removal than that used in the 2011 study. Another possibility is the use of a clustering algorithm that explicitly includes outlier designation and removal as part of its operation. Still another option is to perform a clustering that is restricted from grouping too many different conformations into the same cluster. And, finally, one could change the distance function to improve the conformational purity of the clustering, by biasing the clustering towards one where each cluster is dominated by a single conformation. One final goal to improve on the 2011 CDR study is to improve the measurement of the quality of a given clustering. These measures could include scientific information about structural biology (for instance, that alpha-right regions should not be grouped with alpha-left regions), or could be based solely on the clustered data itself. Possibilities for these potential improvements are discussed below.

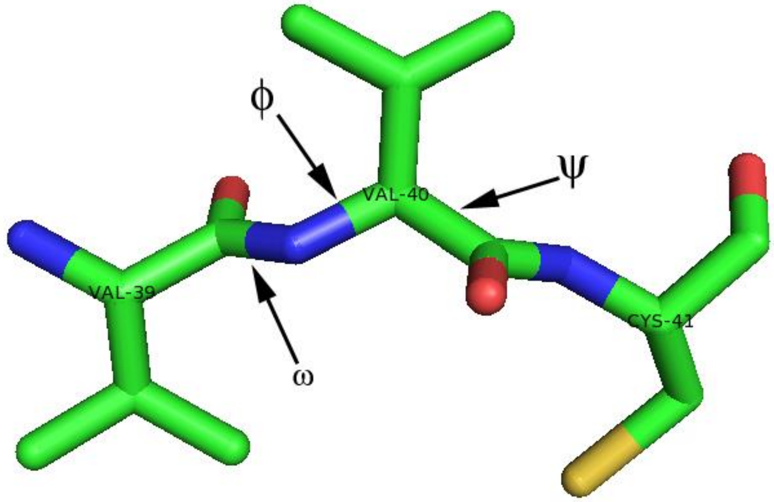

Figure 1-1. Peptide Backbone Dihedral Angles. Rotations about these three backbone dihedral angles are the only degrees of freedom for a peptide backbone with an ideal bond geometry. Omega (ω), which is the dihedral angle for the peptide bond, is the most restrictive of the three, with values either near 0° (a *cis* conformation) or 180° (a *trans* conformation).

In order to compare the results of antibody structural studies, an understanding of what makes a good peptide conformational cluster is necessary. To understand peptide conformations, one needs to be familiar with peptide dihedral angles. Each amino acid residue has three different dihedral angles, phi (φ), psi (ψ), and omega (ω), that together determine the path of a peptide backbone (see Figure 1-1). To first approximation, these three dihedral angles (per amino acid residue position) are the only degrees of freedom for peptide backbone conformations. With a given set of these values over the entire peptide, the backbone conformation is fixed. However, although phi, psi, and omega are dihedral angles (with possible values ranging from -180° to 180°), in practical terms the range of values that they may take on is much more limited (Creighton, 1993). Omega is the peptide dihedral angle with the most restricted set of possible values due to the partial double bond character of the peptide bond. Its values are either near 0° or near 180°, i.e. the four atoms covalently bonded to the carbonyl carbon atom and nitrogen atom participating in the peptide bond are in a single plane, with the alpha carbon atoms of the adjacent amino acids either near each other (which constitutes a *cis* peptide conformation) or away from one another (which is a *trans* conformation). The *trans* conformation is much more energetically favorable except in the case of proline amino acid residues, for which both *cis* and *trans* conformations are common. The range of common values for the phi and psi dihedral angles are context dependent both in terms of the type of amino acid and that of its neighbors, as well as the overall structural context of the peptide chain at large. However, a typical amino acid can take on a set of (φ,ψ) values as shown in Figure 1-2. This is a Ramachandran plot (Ramachandran & Sasisekharan, 1968). This contour plot of conformational density for a generic amino acid has local maxima at the alpha-right (α_R), beta (β), and alpha-left (α_L) regions. The relatively unpopulated region in the lower right quadrant of the Ramachandran plot is labeled the gamma (γ) region. Except for the gamma region, these regions' names are from the peptide conformations that occur if several amino acids in a row take on those particular (φ,ψ) values (that of a right-handed alpha helix, a left-handed alpha helix, and a beta strand, respectively). Roughly speaking, each of these possible discrete conformation definitions describes a different structural context for that amino acid. Each conformation has a different approximate bond geometry for each of an amino

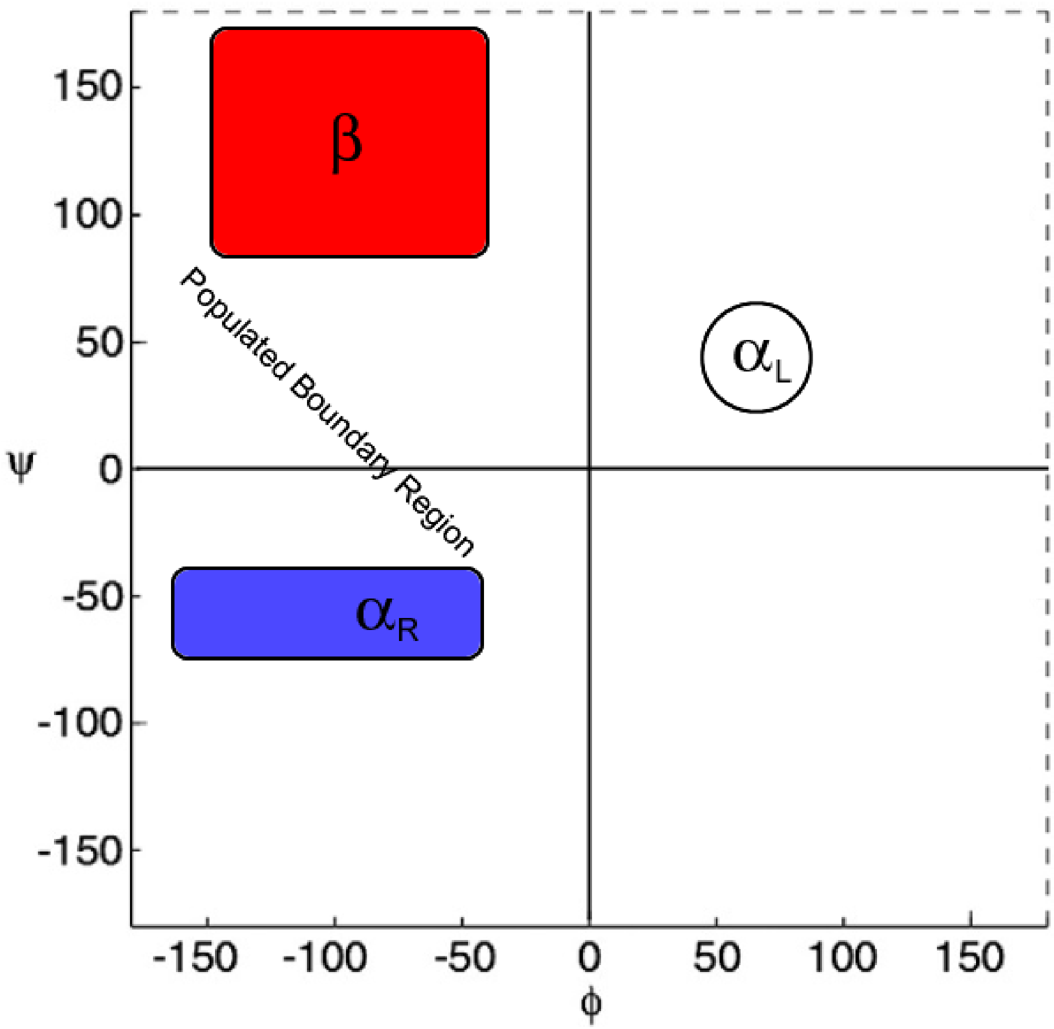

Figure 1-2. A Typical Ramachandran Plot. This is a contour plot of the population density for a typical amino acid (Ramachandran *et al.* 1968). The most commonly occupied areas in the plot are in color (in blue for the alpha-right conformation and in red for the beta conformation). Less common, but also seen, is the alpha-left conformation near (60°, 45°). Notice that the labeled boundary region between alpha-right and beta is moderately occupied. The lower right quadrant of the plot, the gamma region, is very rarely occupied by most amino acid types.

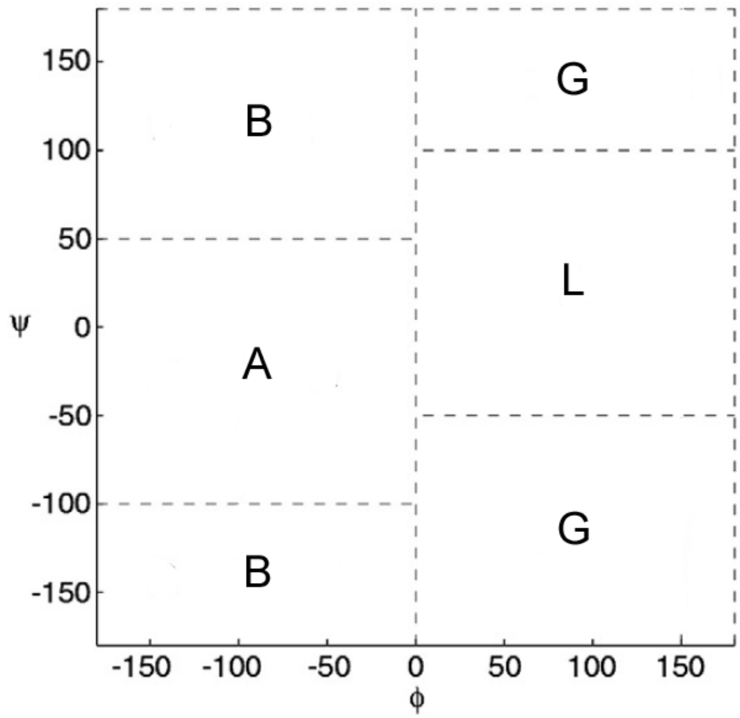

Figure 1-3. Regions of the Ramachandran Plot. This is a partitioning of the Ramachandran plot into regions, each labeled with a letter. All letters but G match a common conformation within their borders.

acid's moieties, which in turn describes how that amino acid interacts with its environment. For instance, hydrogen bonds, which are non-covalent bonds that are quite important in biochemical interactions, are strongly directional in nature and so would depend on the (φ,ψ) values of the amino acid in question (Creighton, 1993). Given the restricted nature of the (φ,ψ) values of a typical amino acid, the area of the Ramachandran plot can be partitioned into regions, with each region named for the peptide structure that is most prevalent within it. The partitioning scheme used here is shown in Figure 1-3. Its possible values are: A, for the alpha-right region; B, for the beta region; L, for the alpha-left region; and, G, for the gamma region. For this work, the Ramachandran classification scheme used is called the ABLG geometric classification alphabet. For the formation of cannot-link clustering constraints, as discussed in the Methods section, the A and B regions are united as the X region, forming the XLG geometric classification alphabet. Using these definitions, the conformation of a given CDR can be described by the sequence of geometric regions that its (φ,ψ) pairs inhabit. For instance, a peptide that consists of two amino acid residues in an alpha-right conformation followed by an amino acid residue in a beta conformation could be described by the geometric sequence AAB. Using this description of a peptide conformation gives the means to quickly compare two different peptide conformations and so provides a metric for judging the quality of a peptide conformational clustering. An ideal conformational cluster, which would be represented by exactly one unique structure, would necessarily have only one geometric sequence among its CDR members. Since each conformational cluster of more than one protein structure has at least some spread to its conformational values, there will also be some variability in geometric sequence even within a single conformational cluster. However, with a tightly clustered group, one geometric sequence should predominate (or maybe a few geometric sequences, if at least one of the residue positions of the cluster is centered on one of the geometric regional boundaries). The geometric sequence definitions in Figure 1-3 are useful for investigating the relative conformational purity for a given cluster as well as giving a geometric sequence label to a given cluster.

CHAPTER 2

REVIEW OF LITERATURE

2.1 Previous Work on CDR Structures

Numerous previous studies of antibody CDR structures have been conducted, starting with the landmark papers by Chothia *et al.* in the 1980s and 1990s (Al-Lazikani, Lesk, & Chothia, 1997; Chothia & Lesk, 1987; Martin & Thornton, 1996; Morea, Tramontano, Rustici, Chothia, & Lesk, 1998). Chothia and his coworkers were the first to identify the existence of "canonical structures," which are a small number of three dimensional shapes that most antibody CDRs exhibit. In other words, even though there is a large number of antibody CDRs with a very large number of possible loop conformations, the ensemble of structures they form is much more restricted than the set of possible shapes they could take on given their length. Chothia originally stated this hypothesis in his 1987 paper, stating "most of the hypervariable regions in immunoglobulins have one of a small discrete set of main-chain conformations that we call 'canonical structures,'" and went on to state that these canonical structures are typically caused by certain set amino acid residues at a few positions within each CDR (Chothia & Lesk, 1987). For that first 1987 paper, only a few antibody structures were available and Chothia *et al.* grouped them into clusters by hand by both structure and sequence, with each cluster representing, in their words, a "canonical structure." Each subsequent major antibody structural survey followed the same essential concepts (Al-Lazikani et al., 1997; Kuroda, Shirai, Kobori, & Nakamura, 2008; Martin & Thornton, 1996; Oliva, Bates, Querol, Avilés, & Sternberg, 1998; Shirai, Kidera, & Nakamura, 1999). The last paper by Chothia and coworkers on antibody CDR structural classification, in 1997, used only 17 high-resolution protein structures (Al-Lazikani et al., 1997). Out of the following papers, of particular importance is the paper by Martin and Thornton in 1996, which found much more evidence for the canonical structures hypothesis (Martin & Thornton, 1996). In addition, rather than the ad hoc clustering of Chothia and coworkers, Martin and Thornton used a quantitative clustering approach using peptide backbone conformations. To represent peptide conformations, they used a vector for each sine and cosine value for each of the backbone dihedral angles (φ, ψ, and ω), so the conformation of a 10 residue peptide would be represented by a 60-dimensional vector in their scheme, with each vector having a value between +1 and -1, inclusive. They then perform a hierarchical agglomerative clustering using Ward's minimum variance method for a preliminary clustering (Ward, 1963). For each cluster, they divide the error value by the number of dimensions to get a parameter that is comparable across different clustering runs (of peptides of different length). To that normalized parameter E, a cutoff value of 0.06 is applied which provides a stopping point for the agglomerative clustering. The preliminary clustering is followed by a cluster merging step using root-mean-square deviation (RMSD) to merge clusters whose member medoids have low RMSDs to one another, where each cluster's medoid is defined to be the CDR loop with the smallest SSE to the other members of its cluster. Although this RMSD-based cluster merging joins clusters whose alpha carbon atoms overlap closely when superposed, the clusters it joins may be quite different in conformation, which can be a considerable disadvantage when searching for "canonical structures." Overzealous clustering aside, their project was a significant improvement in part due to the much larger number of antibody structures in their study in comparison to the Chothia studies (57 structures were used in the Martin and Thornton study). Martin and Thornton used the loop length as well as the CDR type as part of their nomenclature and classification scheme, which is also a useful addition. CDR loops of different lengths, even if of the same type, are necessarily going to have a different structural context. So, the antibody CDR classification problem becomes the structural classification of each of the observed antibody CDR type-lengths, with each CDR falling into one of these type-length groups.

Given that the above description of peptide structures and Ramachandran plots gives a good basis for recognizing Chothia's "canonical structures," all that is necessary to test his hypothesis is a large enough sample of antibody structural data and a proper mechanism for handling conformational outliers, i.e. noise. In such a database, dense clustering in conformational space should occur for a given CDR type-length if Chothia is correct, as subsequent works by others attest (Al-Lazikani et al., 1997; Chothia & Lesk, 1987; Kuroda et al., 2008; Martin & Thornton, 1996; Morea et al., 1998; Oliva et al., 1998). In 2011, we performed just such a survey (North et al., 2011). Rather than the 17 antibody structures used by Al-Lazikani *et al.* (1997) or the 57 structures from Martin and Thornton (1996), our 2011 study used a database of 923 antibody structures. In order to minimize the amount of noise in the data, that large database was significantly pruned of entries that were deemed to be likely to have poor quality structures. See Table 2-1 for the size of the database of the 2011 study. Note that the removal of poor quality

Table 2-1

Size of the 2011 North et al. Database

	L1	L2	L3	H1	H2	H3
Initial database	1334	1334	1334	1262	1262	1262
Poor quality structures	447	440	444	404	392	455
Redundant CDRs	577	582	578	525	529	487
Outliers	9	1	1	3	4	13
Included in 2011 survey	301	311	311	330	337	307

Only the structures counted in the Outliers line were removed due to having different conformations than the rest of the database.

structures should reduce the number of outliers in the data, making the "canonical structures" described by Chothia and coworkers easier to find. Rather than counting in terms of full antibody structures, Table 2-1 refers to the number of antibody chains involved at each step. Each antibody structure may consist of multiple chains (which are typically, but not always, nearly identical copies of the same VH and VL domains). After collecting antibody structures from the Protein Data Bank, ill-suited structures due to poor structure quality are omitted. First, any chain from an x-ray structure with a resolution worse than 2.8 Å is omitted. Any structure with a resolution worse than this would become so poor that the reported dihedral angle values of the peptide backbone would not be reliable. Next, chains that have backbone atom B-factors greater than 80 (or have unreported B-factors, with a value of zero) are also removed. Each atom in an x-ray structure has a reported B-factor, or temperature factor, that indicates how much variation there is in that atom's position due to motion (which includes thermal motion, hence the name "temperature factor") (Creighton, 1993). The higher the B-factor, the less the certainty present in that atom's position in the reported structure. Next, any CDR for which backbone atoms were missing from the structure were excluded, since it would be impossible to calculate the backbone dihedral angles without all of the backbone atoms of the loop being present. Next, there is a cutoff in a calculated conformational energy term. Again, CDR loops with unusual structures should have a high energy and hence should be filtered out by this step, thus reducing the number of outliers in the clustering. The conformational energy was calculated from the Ramachandran probability distributions for each of the twenty common naturally occurring amino acids that our lab previously published (Ting et al., 2010). Next, CDR structures that had a non-proline residue with a *cis* conformation, which is very unfavorable, were removed. The previous filtering steps except for the structure resolution cutoff remove only a handful of structures. The structure resolution cutoff removed the most at roughly a quarter of the initial database. After these filters, so-called redundant structures were filtered out of the database. If two or more chains have the same amino acid sequence for all three CDRs, they were considered to be redundant. In that case, only the best quality structure (first measured by resolution, then by B-factor, then by conformational energy) was included in the clustering database. Nearly half of the original structures are culled this way. Finally, conformational outliers are removed. For the 2011 study, a conformational outlier is one that has a backbone dihedral angle value that is at least 90° away from the corresponding dihedral angle value for each other CDR of that CDR type-length. There are very few conformational outliers in the database (at most 13, for CDR H3), which demonstrates that either the conformational outlier criterion is a very lenient one (allowing for a very large degree of structural variability) or that the conformational data are very tightly clustered over all of the CDR type-lengths. There were two different motivations for the database filtering procedures of the 2011 work. One motivation is to get an accurate depiction of the relative populations of the different CDR conformational classes. This motivation was the impetus behind the removal of so-called redundant structures. Note that some antibodies structures have been solved multiple times (for instance, to examine that antibody's changes in conformation upon binding to its epitope), while others have been solved only once. The removal of redundant structures should resolve that problem and the resulting error it would cause in calculating the relative frequencies of the different antibody CDR conformations. The other motivation for the database filtering is to remove as much noise as possible before clustering, which uses the assumption that poor quality structures are the cause of the bulk of the noise. This pre-clustering processing procedure is especially important given that the choice of clustering algorithm for the 2011 survey does not have any mechanism for outlier removal.

The 2011 survey divides the CDR data into groups based upon both CDR type-length and the *cis/trans* peptide conformations of each residue position (North et al., 2011). Then, the affinity propagation clustering algorithm is applied using only the peptide conformations (dihedral angles) for its conformational clustering (Frey & Dueck, 2007). Affinity propagation clustering is an improvement over the hand-picked clustering by Chothia and coworkers and the clustering of Martin and Thornton (Al-Lazikani et al., 1997; Chothia & Lesk, 1987; Martin & Thornton, 1996). One reason for the improvement is that neither Chothia and coworkers nor Martin and Thornton used any sort of data filtering for outlier removal because they did not have enough data to do so. Also, the improvement over Martin and Thorton's clustering is in part due to the removal of the post-clustering cluster merging due to low RMSD while ignoring the dissimilarity in peptide conformations between cluster medoids. Before clustering, the pairwise conformational distance for each CDR loop within each group is calculated. First, the distance between the corresponding dihedral angles at the same residue position for two different CDR loops is computed using the formula below:

$$d(\theta_1, \theta_2) = 2(1 - \cos(\theta_1 - \theta_2)) \qquad \text{Eqn. 2.1}$$

This value for the distance, *d*, can be interpreted geometrically as the squared distance of the chord on a unit circle connecting the vectors (cos θ_1, sin θ_1) and (cos θ_2, sin θ_2), where θ_1 and θ_2 are the angles whose distance is being measured (North et al., 2011). Then, for two CDR loops *i* and *j* of the same type, length, and *cis-trans* peptide conformations, the distance between them is the sum of the angular distances of each of their (φ,ψ) backbone dihedral angle pairs:

$$D(i,j) = \sum_{r=1}^{k} d(\varphi_{ir}, \varphi_{jr}) + d(\psi_{ir}, \psi_{jr}) \qquad \text{Eqn. 2.2}$$

The above equation, which is the L_1 norm for the dihedral distance, is used to generate the pairwise distance matrix for the group. This distance matrix is used as the input for the affinity clustering algorithm of Frey and Dueck (Frey & Dueck, 2007). The affinity algorithm takes similarities as input, so the negative of the distance *D(i.j)* is used for $(i \neq j)$:

$$s(i,j) = -D(i,j) \qquad \text{Eqn. 2.3}$$

The self-similarities are equal to a constant times the average value of the non-self-similarities. For the 2011 work, this constant was fixed to be equal to 1:

$$s_{self} = \frac{2}{n(n-1)} \sum_i \sum_{j>i} s(i,j) \qquad \text{Eqn. 2.4}$$

The constant factor for Equation 2.4 is a scaling parameter that sets the clustering tendency of the algorithm and is the only adjustable parameter for the input.

Affinity propagation clustering employs a message passing system to perform the clustering, which can be thought of as a means of conveying clustering information (Frey & Dueck, 2007). After processing, each CDR loop is assigned an exemplar, which is the CDR loop that best represents that CDR loop (and might be the same CDR loop, or might not). Also, there are two matrices of pairwise values for each set of CDR loops to be clustered, which are the responsibilities *r(i,k)* and the availabilities *a(i,k)*. At the start of the algorithm, all availabilities are set to zero. Then, for each step of the algorithm, first the responsibilities are updated. Each responsibility *r(i,k)* is a message passed from point *k* to point *i* and indicates how well *k* would serve as an exemplar (according to point *i*):

$$r(i,k) \leftarrow s(i,k) - \max_{k' \neq k}\{a(i,k') + s(i,k')\} \qquad \text{Eqn 2.5}$$

Once the responsibilities are updated, then the availabilities are updated. Each availability *a(i,k)* is a message passed from potential exemplar point *k* to point *i* and is a measure of how well point *k* would serve as an exemplar for point *i*. For $k \neq i$:

$$a(i,k) \leftarrow \min \left\{0, r(k,k) + \sum_{i' \notin \{i,k\}} \max \{0, r(i',k)\}\right\} \qquad \text{Eqn 2.6}$$

For the self-availability,

$a(k,k) \leftarrow \sum_{i' \neq k} \max \ \{0, r(i',k)\}$ Eqn 2.7

The exemplar for each point *i* is then calculated:

$Exemplar(i) = arg\max_k\{a(i,k) + r(i,k)\}$ Eqn 2.8

The point-exemplar relationships are used to generate an undirected graph where each link connects a point to its exemplar. The connected components of that graph are the clusters. The algorithm is terminated when it has run at least ten iterations and the cluster assignments have been constant for at least four iterations (North et al., 2011). After the affinity propagation clustering is complete, a second clustering step is performed. First, for each cluster, the CDR loop with the smallest median distance to the other CDR loops of its cluster is found and designated to be the representative CDR loop for that cluster. Then, each of the representative loops for CDRs of the same CDR type-length and *cis-trans* conformation are compared. For any given pair of clusters, if their representative CDR loops differ in dihedral distance by less than 65° for each of their (φ,ψ) dihedral angle pairs, then those two clusters are merged into one. Note that no structures are labeled as outliers during the clustering process, so any labeling of outliers is performed during pre-processing before the clustering algorithm is run.

The results of the structural clustering of the 2011 paper were good, but could be improved. Also, some CDR type-lengths have a greater clustering tendency than others. The CDR type-lengths with less of a clustering tendency tended to fare poorly using the data pruning and clustering procedure of the 2011 paper because many structures that should have been labeled as outliers were not designated as such. Here are the clustering

Table 2-2

Conformations of CDR L1-11 from North et al. 2011

	Size	1	2	3	4	5	6	7	8	9	10	11
Cluster 1	76	B	B	A	B	B	A[a]	G[a]	A	Z	B	B
Cluster 2	55	B	B	A	B[a]	B	A	L	L	Z	B	B
Cluster 3	5	B	B[a]	B[a]	L[b]	A[a]	A	A	B	B[a]	B	B

Conformation Z is on the border of A and B

[a]Single outlier present

[b]Multiple outliers present

results of two different CDR type-lengths from the 2011 work to illustrate that point. To start, here is the example of CDR loop type-length L1-11, which is of type L1, is eleven amino acids long, and is comprised of all *trans* residues. Residues are assumed to be in the *trans* conformation unless stated otherwise, and clusters are numbered in descending order of size, so that the most populous cluster is cluster 1, and the next most populous is cluster 2, and so on. This nomenclature will be followed throughout the paper. CDR loop type-length L1-11 is an example of a CDR group that is well clustered in the 2011 analysis. See Figure 2-1 for a Ramachandran plot of the structurally variable residues of L1-11. There are two large clusters, clusters 1 and 2, and a small cluster, cluster 3. Clusters 1 and 2 are well separated from each other and are tightly clustered. Cluster 1 had a few outliers. Here, the ABLG conformational classification is useful, as shown in Table 2-2. For instance, clusters 1 and 2 have the same ABLG alphabet until residue 7. There, cluster 1 is in the G conformation (with one outlier in the A region) while cluster 2 is in the L region. For residue 8, cluster 1 is in the A region while cluster 2 is in the L region. Then, for residues 9, 10, and 11, both clusters 1 and 2 are in the B region. For all

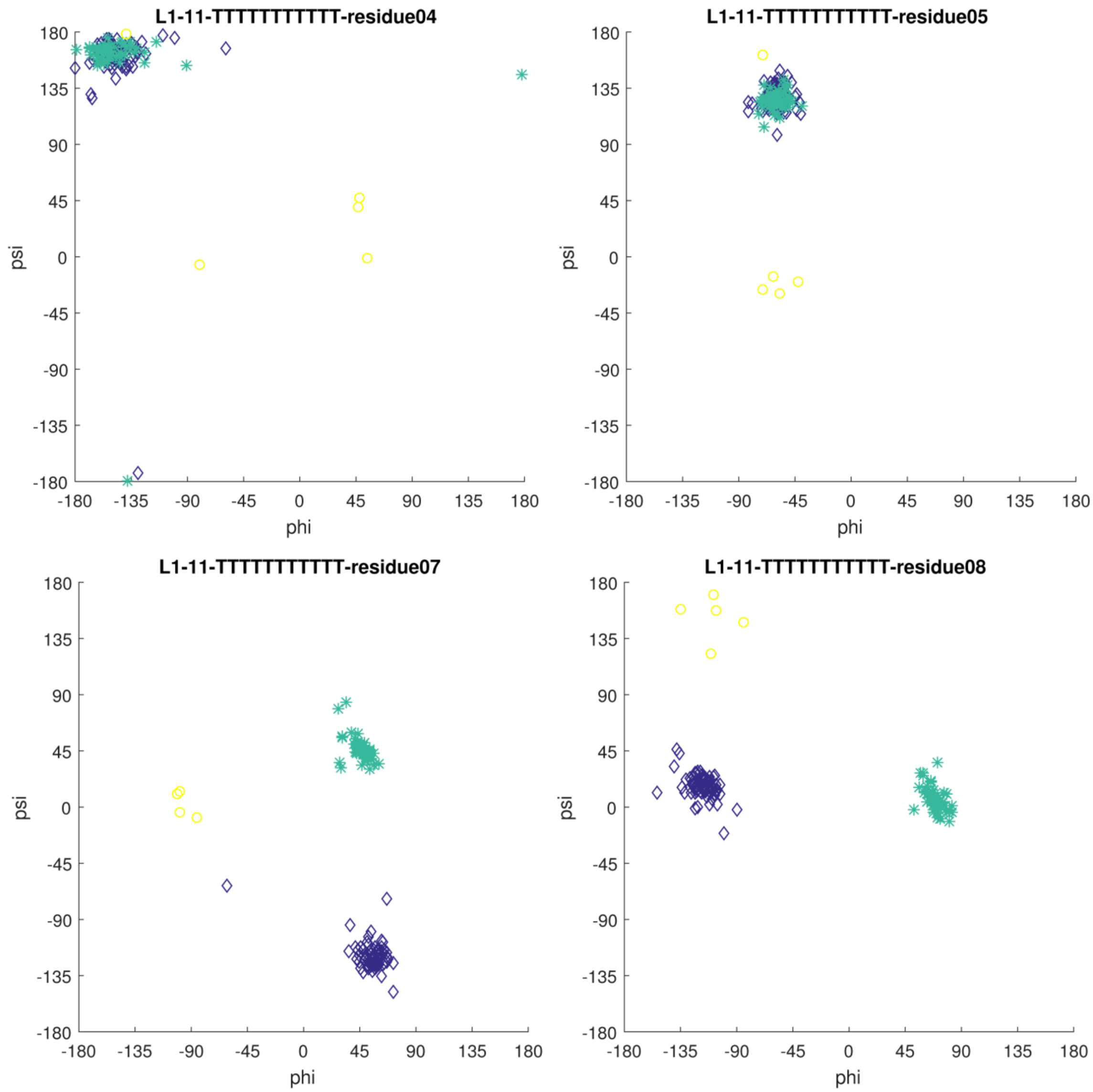

Figure 2-1. Clustering of L1-11 in North *et al.* 2011. These are the four most diverse residue positions in terms of conformation for CDR type-length L1-11. Each Ramachandran plot is labeled with its residue position, with the *cis-trans* conformation of each peptide labeled, showing that all are in a *trans* conformation. Cluster 1 is shown as purple diamonds, while cluster 2 is shown as green asterisks. The tiny (5 member) cluster 3 is shown as yellow circles. Clusters 1 and 2 are well clustered with few outliers, the worst being the cluster 1 member in residue 7 near (-50°, -50°). Cluster 3 is much more scattered with two evident outliers in residue 4 (one in the A region, and the other in the B region) and another outlier value for residue 5 (in the B region).

of clusters 1 and 2, the worst outlier is the data point of cluster 1 in the A region at residue 7. Overall, though, this is a very good clustering for clusters 1 and 2. Cluster 3, however, is much worse. It consists of only 5 CDR loops, and even with such a small cluster there are still several residue positions where cluster 3 CDR loops take on multiple conformations. For residue 2, one CDR of cluster 3 is in the G region while the rest are in the B region. For residue 3, four CDRs of cluster 3 are in the B region (although considerably spread out around it) while one is in the A region. Residue 4 is a particularly bad position for cluster 3, with three CDR loops in the L region, one in the A region, and one in the B region. For residue 5, one of the cluster 3 CDR loops is in the B region while the rest are in the A region. For the rest of the residues, all of the CDR loops of cluster 3 are in the same region for each position, although for residue 9 they are in the same region (B), but one CDR loop is far away from the others. Given that cluster 3 has both very few members and a high tendency to have outliers, it likely should never have been included in the clustering in the first place. Ideally, all of its members should have been marked as outliers. A CDR type-length that fared worse in the 2011 paper is H1-13. The affinity propagation clustering algorithm found eleven clusters, although the cluster H1-13-1 (cluster 1 for H1-13) has by far the most of the CDR loops of that type and length, with 87 percent of the CDR loops of that type-length, with the other clusters having at most two percent of the CDR loops of that type length (North et al., 2011). Cluster 1 has outliers at positions 4, 7, 8, and 9. Residue 4, shown in Figure 2-2, is the most scattered position for cluster 1, with the majority of the CDR loops in the L region, but several in the G region (both above and below the L region, which is quite a bit of spread to the data). For residue 7, cluster 1 is almost entirely in the A region, but has two

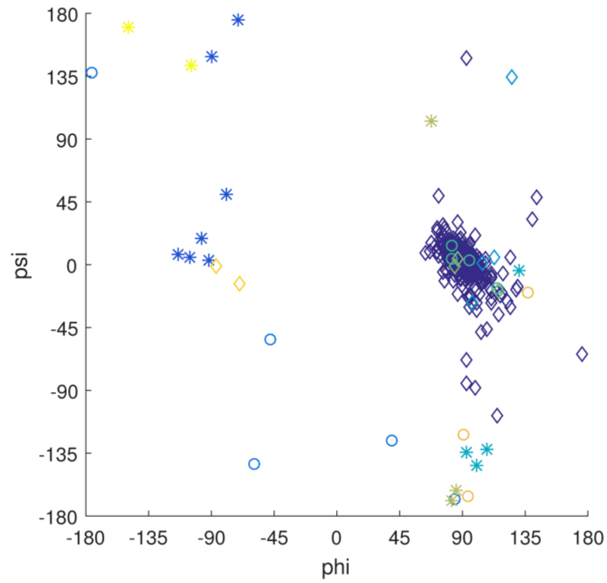

Figure 2-2. Clustering of H1-13 in North et al. 2011. This Ramachandran plot of the fourth residue of CDR type-length H1-13 shows that the lack of stringent outlier removal and the use of the affinity clustering were not adequate for clustering this CDR type-length. Cluster 1, the largest, is shown in purple diamonds, while the other clusters are shown in various different colors.

members in the L region (one bordering with the G region). For residue 8, most of cluster 1 is in the A region, but one CDR loop is in the B region. Overall, cluster 1 has some scatter, but is reasonably well clustered. The other, less populated clusters of CDR type-length H1-11 have less well defined clusters, and many have so few members that they should likely be considered outliers rather than clusters. Cluster 2, shown as blue asterisks, is a good example and is the most numerous of the less populated clusters. At residue 3, cluster 2 is reasonably well clustered in the A region. For residue 4, cluster 2 is split between the A and B regions, with four CDR loops in the A region, one on the border between the A and B regions, and two in the B region. Residue 5 also shows that cluster 2 contains noise, with most of cluster 2 in the B region, but one of its CDR loops is in the A region and another in the G region. For the rest of the residue positions, cluster 2 is reasonably well clustered, with its members all in the same geometric region. Even so, it is clear that cluster 2 of H1-13 has too many outliers in too many positions to be considered a proper cluster, and the subsequent clusters of H1-13 are even smaller. From the suffuse data points of clusters 2 through 11, it is clear that not enough was done in the 2011 paper to remove outliers from the H1-13 CDR loop type-length.

2.2 Alternative Methods for Assigning and Evaluating Peptide Clusters

Rather than trying to improve the pre-processing of CDR loop structures prior to clustering, one option for improving the structural clustering is to include outlier detection in the clustering algorithm itself. One clustering algorithm that does this is DBSCAN (Ester, Kriegel, Sander, & Xu, 1996; Tan, Steinbach, & Kumar, 2006). DBSCAN is a density-based clustering algorithm that, based upon local density, labels each data point as a core point, a border point, or an outlier. There are two input parameters for DBSCAN, Eps, which is compared to the link strength, and MinPts, which is an integer. Consider the data as a graph where each vertex is a data point from the original data, and data points are linked to each of the others with a link strength equal to the distance (or the similarity, depending on the comparison function being used) between the pair linked. Then, the graph is sparsified by removing all links with a distance greater than Eps (or, in the case of the link strength being equal to the similarities between points, links with a similarity less than Eps are removed). In the DBSCAN residual graph, each data point with at least as many neighbors (points that are adjacent to the reference point in the graph) as the value of the MinPts parameter is labeled as a core point. Once the core points are labeled, every unlabeled point that is adjacent to a core point is labeled as a border point. After that, any unlabeled points that are left are designated to be outlier points. To find the clusters of a DBSCAN clustering, take the DBSCAN residual graph and remove all vertices but the core points. The connected components of the resulting graph are the clusters from that DBSCAN clustering run. Then, take each border point and assign it to the cluster of the closest core point. Outlier points, of course, are not assigned to a cluster. DBSCAN has a number of advantages and disadvantages: while it is very good at clustering in the presence of noise, and is good at eliminating outliers, it can struggle when dealing with clusters of different density or with high dimensional clustering problems (Tan et al., 2006). In the context of clustering CDR loop structures, DBSCAN may have problems with some of the CDR type-lengths of longer loops due to the dimensionality. Once the CDRs are also sorted by length and *cis-trans* peptide bond types into groups, then there are two dimensions per residue position to the clustering problem, which are the (φ,ψ) backbone dihedral angles. However, given how the CDR loop types are defined to begin and end, many of the residues at the beginning and at the ends of loops have the same conformation for each of the different loop types, being typically in the B region. This effect reduces the overall dimensionality of the problem. However, particularly long loops, such as some of the H3 CDRs, have many variable positions and hence are more difficult to cluster (Kuroda et al., 2008; Morea et al., 1998; North et al., 2011; Oliva et al., 1998; Shirai et al., 1999; Weitzner, Dunbrack, & Gray, 2015).

The issue of high dimensionality in the CDR loop structure clustering problem could be dealt with in a number of ways. If noise in a specific dimension or dimensions causes a problem, then dimensionality reduction techniques could be used. However, the larger problem in CDR loop structure clustering is ensuring that the differences in each residue position is recognized in the final clustering, rather than ignoring a residue position that has more scattered data points than other residue positions. This fact, coupled with the desire to keep the distance or similarity function to be readily physically interpretable, makes the various dimensionality reduction techniques less appealing. Rather than using dimensionality reduction, one possibility is to use a distance function that alleviates the curse of dimensionality while still penalizing the formation of clusters with very diverse conformations. For instance, one option is the supremum distance, which is the L_{max} or L_{∞} norm. The supremum distance is just the maximum distance for a single attribute (dihedral angle, in this case) present among all of the attributes. The distance, as is true for the other dihedral angle distances in this work, is calculated using Equation 2.1. If such a distance is capable of distinguishing the different clusters that are relatively close to one another, it should serve well as a distance function since it will be very capable at keeping clusters with residues in geometric regions from being put into the same cluster. This latter fact is due to the large difference in dihedral angle for the typical positions that residues take within the Ramachandran plot (Ramachandran & Sasisekharan, 1968). Refer to points in Figure 1-2 and note the large distances between the populated areas of different geometric regions. The supremum distance will cause loops inhabiting those densely populated wells in different geometric regions to have some of the largest distances overall among all of the pairwise distances of the group, and so will virtually guarantee that such points will not end up in the same clusters. Another option for a point pairwise comparative function that could improve DBSCAN is using a shared nearest neighbor (SNN) similarity used in the Jarvis-Patrick clustering algorithm as the similarity function (Ertöz, Steinbach, & Kumar, 2002; Jarvis & Patrick, 1973; Tan et al., 2006). To calculate the SNN similarity between two points, first set the length *k* of the nearest neighbor list. The value of *k* must be less than the size of the group being clustered. Then, find the *k* closest neighbors for each point. Finally, for any pair of data points, draw a link between them if they are on each other's nearest neighbor list, with the strength of the link being the number of nearest neighbors they share on the nearest neighbor list of length *k*. Then, apply the DBSCAN clustering algorithm to the resulting graph. Set the parameter Eps to be a similarity threshold, and remove any link that falls below the Eps threshold. Then, label points as core, border, or outlier points as described above for DBSCAN with the MinPts parameter as another input value. The SNN-DBSCAN clustering algorithm is better than DBSCAN based on Euclidean distances at handling data sets with high dimensionality (Ertöz et al.). Consequently, if high dimensionality is a problem for some of the CDR type-lengths, then the SNN-DBSCAN algorithm may improve the clustering results.

An alternative method to a clustering algorithm that eliminates outliers is a clustering algorithm that uses biophysical information to guide its actions. There is a considerable body of literature focused on the problem of clustering with the addition of outside information, which is variously called semi-supervised learning, clustering using pairwise constraints, and clustering using instance level constraints (Basu, Banerjee, &

Mooney; Basu, Bilenko, & Mooney, 2004; Davidson & Basu, 2007; Schwenker & Trentin, 2014; Wagstaff & Cardie, 2000; Wagstaff et al., 2001). In semi-supervised learning, knowledge of a proper classification (from domain knowledge, for example) can be used to improve clustering. In this subfield, clustering without such knowledge is called unsupervised learning. Among the founding papers of this field are two by Wagstaff and coworkers (Wagstaff & Cardie, 2000; Wagstaff et al., 2001). In the 2000 paper, Wagstaff *et al.* proposes the idea of pairwise constraints. They suggest that clustering accuracy can be improved and the number of iterations required for convergence can be reduced through the addition of a few constraints. Rather than constraint in the form of class labels for selected data points, Wagstaff *et al.* suggest the use of two different kinds of pairwise constraints: must-link, or ML, constraints, such that ML(a,b) means that points a and b must be in the same cluster; and, cannot-link, or CL, constraints, such that CL(a,b) means that points a and b cannot be in the same cluster. Note that ML constraints are transitive: if ML(a,b) and ML(b,c), then it must be that ML(a,c). In addition, ML constraints have implication for CL constraints as well. If point a and point b have a CL constraint between them, then any point with a ML constraint with point a cannot be in a cluster with point b, and therefore is in a CL constraint with point b. With the conformational geometric regions in the geometric classification alphabet defined in Figure 1-3, a quick method of generating CL and ML pairwise constraints throughout each CDR type-length group is possible. First, by comparing Figure 1-2 with Figure 1-3, it is clear that there is some possibility of a peptide conformation at the boundary between regions A and B. Therefore, forming a CL constraint on the basis of loop x having a residue in region A while loop y has a residue at the same position in region B would have the potential to restrict relatively similar CDR loops from being in the same cluster. However, this potential overlap in regions is not as much as a problem between the other possible pairs of the geometric alphabet. So, this could be the rule for CL formation: for any two CDR loops *x* and *y*, if for any amino acid residue position of the loop the geometric (ABLG) classification differs for any corresponding residue pair and one of the two residues in question is in the L or G regions, create the cannot-link relation CL(*x*,*y*). This reduced alphabet for finding cannot-link (CL) constraints is called the XLG geometric alphabet. For the must-link relation, a higher degree of selectivity is desirable to avoid combining clusters that have the same geometric (ABLG) sequence but still differ enough from one another that they should be put into different clusters. So, the rule to form a must-link constraint is this: if two CDR loops have matching geometric sequences for each of their corresponding residue positions and each of their corresponding backbone dihedral angles are within 15° of each other, then they should be connected with a ML constraint. Wagstaff used pairwise instance level constraints to modify the K-means algorithm to include constraints, which they named the COP-KMEANS algorithm (MacQueen, 1967; Wagstaff et al., 2001). First, they add ML and CL constraints to the data. Then, they initialize the K-means algorithm as normal, choosing the positions of *k* of the points to be cluster centers. Next, they modify the assignment of points in the original k-means algorithm. For the original k-means algorithm, each point is added to the cluster of the closest cluster center. In COP-KMEANS, it is added to the cluster of the closest cluster center for which adding the point to that cluster will not violate any of the pairwise constraints. If this is impossible, the COP-KMEANS algorithm returns failure, indicating that the clustering is impossible. Assuming that each of the data points has been assigned to a cluster center, then the cluster centers are updated by calculating the average position of the cluster members for each cluster and assigning those average positions to be the new cluster centers. Then the algorithm repeats itself until it converges (i.e. the cluster assignments no longer change). One difficulty with using this algorithm on the CDR loop characterization problem is that the average of two possible conformations is often an impossible conformation. For instance, the midway point in the Ramachandran plot between an alpha-right and an alpha-left conformation is a relatively uninhabited region of the plot. Rather than using k-means as the template for a constrained clustering algorithm, then, k-medoids is used as the template algorithm (Kaufman & Rousseeuw, 1987). K-mediods differs from k-means in that rather than using average cluster centers, it nominates an exemplar or medoid data point for each cluster. The medoid is chosen such that it has the smallest dissimilarity the other members of the cluster out of all of the data points belonging to that cluster. Other than the Wagstaff method, which uses what are called hard constraints because they must be obeyed, there are other methods that use so-called soft constraints because, while the constraints are used to improve the results of the clustering, they are not required to be obeyed. Because the constraints used in this work have been chosen to be limited in scope it is preferable to require that the constraints be obeyed. Therefore, COP-KMEANS will be modified to use k-medoids for this work rather than using the probabilistic methods of Basu *et al.*

Finally, there is the question of how to measure the clustering quality to compare the results of different clustering algorithms on the same data sets. This work will use three of them. Two clustering quality metrics, the average silhouette coefficient and the normalized average distance, are based purely on the nature of the clustering results. The silhouette for each point *i* is computed using the following expression:

$$s_i = (b_i - a_i)/\max\ \{a_i, b_i\} \qquad \text{Eqn. 2.9}$$

where a_i is the average distance from point *i* to all other points in the same cluster, while b_i is the average distance from point *i* to the closest cluster that point *i* does not belong to (i.e., the cluster that point *i* does not belong to that has the smallest average distance to *i*) (Rousseeuw, 1987). The silhouette coefficient has a maximum value of 1, indicating that the data point is a member of a very tight cluster and the next closest cluster is very far away, and has a minimum value of -1. Any value less than zero indicates that the data point is closer on average to the next closest cluster than it is on average to the members of its own cluster. Although the silhouette is undefined for clusters of one data point and for clustering runs with only one cluster as a result, I have chosen to define such situations to have a silhouette coefficient of zero, indicating that there is no available evidence that any given point should be in the cluster to which it is assigned or to a different cluster. The average silhouette coefficient is just the average value of the silhouette coefficient of each point. This average can be over each cluster separately, to give the distinguishability of each cluster, or can be an overall average for all of the data points, which quantifies the quality of the clustering of that CDR type-length. The next clustering quality measure is the normalized average distance for a cluster. To calculate the normalized average distance, first find for each cluster the data point with the smallest normalized average distance to each other member in the cluster. This point is the mediod, and the normalized average distance is its average distance to each member in the cluster divided by the number of backbone dihedral angles for that CDR loop-length. It is a measure of the scatter in the data for each cluster. Finally, there is the geometric classification purity of a cluster. For each cluster, the geometric region of each residue for each data point is calculated to give a geometric sequence where each letter represents the conformation of each residue position for that data point as described above. The unique strings present for each cluster are then listed with the percentage of data points of that cluster that have that geometric sequence. Tightly grouped clusters should have only a few geometric sequences make up the majority of the cluster. Another means of using this measure is to calculate an entropy term using the geometric sequence fractions of the cluster in question. Here the Shannon entropy of the geometric sequence of a CDR cluster, using the base 2 logarithm, is used:

$$H = -\sum_i p_i \log_2 p_i \qquad \text{Eqn. 2.10}$$

where i is an index over the conformational strings present for the cluster and p_i is the fraction of data points of that cluster with that geometric sequence (Shannon & Weaver, 1998). If all of the CDR structures of a cluster have the same geometric sequence, then the entropy will be zero. As the cluster is spread out among more geometric sequences, the entropy increases.

CHAPTER 3

METHODS

For this work, two different data sets are used. The first is the CDR loop set used for the 2011 paper, while the second uses the current database from the PyIgClassify website (Adolf-Bryfogle, Xu, North, Lehmann, & Dunbrack, 2015; North et al., 2011). PyIgClassify uses the same framework to update its antibody information that the 2011 paper used for its database generation, so the CDR definitions (i.e. where each CDR loop type begins and ends) are consistent. Since 2011, the size of the database has grown by approximately a factor of four. As of January 2017, there are 4,066 L1 CDR structures, 4,089 L2 CDR structures, 4,089 L3 CDR structures, 4,251 H1 CDR structures, 4,314 H2 CDR structures, and 4,145 H3 CDR structures. Any structure with missing backbone atoms from the model have been removed from the dataset prior to the count listed above. Unlike the 2011 dataset, no structures have been removed due to structure quality or redundancy. Instead, the goal is to classify as many structures as possible while testing the ability of each clustering algorithm to deal with noise.

Two different clustering algorithms were implemented in C# for this project, DBSCAN and k-medoids (Ester et al., 1996; Kaufman & Rousseeuw, 1987). For DBSCAN, there are three different comparison functions used. Two are distance functions. One is the same distance function as used in the 2011 paper, which is the sum of the backbone dihedral angle distances as shown in Equation 2.2. The other is the supremum, or the L_{max} norm, for which only the largest dihedral angle distance is used.

The third comparison function is the (Jarvis-Patrick) SNN similarity (Ertöz et al., 2002; Jarvis & Patrick, 1973). There are two input parameters to DBSCAN, Eps and MinPts. To find the optimal values for these, a grid search is performed for each CDR type-length over different values for Eps and MinPts. For each grid point, a target percentage of links to keep in the simulation and a target percentage of core points are set. Then, the algorithm sets the value of Eps such that at least as large a fraction of pairwise links as the target links percentage is kept, while trying to keep the kept links percentage as close to the target percentage as possible. If the only possibility for doing this is to keep all of the links, then the optimizer returns a failure code for that grid point. Once Eps is set, MinPts is set in an analogous way, except that its target is the fraction of data points to be core points rather than the fraction of links to keep. This procedure is similar to that used by Ertöz *et al.* for the examples in their work (Ertöz et al., 2002). For the DBSCAN clustering runs for both the 2011 and 2017 datasets a grid of link fractional and core point fractional values from 0.2 to 0.8 with a step value of 0.05 are used for the parameter optimization grid. For SNN-DBSCAN, the same grid points are used for the 2011 dataset analysis, which uses 20 as a value of k, the length of the nearest neighbors list. For the 2017 dataset, SNN-DBSCAN uses the same grid point system for the parameter k equal to 20, but extends the grid point system for a value of k of 100, for which the Eps grid was extended to include values from 0.01 to 0.20 with a step value of 0.01. The value of Eps, MinPts, and k, if applicable, are chosen for each CDR type-length to maximize the average value of the silhouette coefficient. These optimal values of the input parameters are then used to generate the clustering.

To test the use of constraints as defined in the introduction, the k-mediods equivalent to COP-KMEANS is used. Figure 3-1 features an outline of the algorithm, called Constrained K-medoids. One complication of Constrained K-medoids for this problem is that there is a CL constraint for any CDR loop pair that differs in their conformational strings by an L or a G. To accommodate each of the minimum necessary number of clusters, first a pseudo-conformational string is calculated for each CDR loop for a given CDR type-length, where the geometric alphabet used is L, G, and X for any dihedral pair not in the L or G regions. Also, each CDR loop is added to a pool of potential initial cluster medoids. Then, each CDR loop is added to a hash table where the key is the pseudo-conformational string and the value is a list of CDR loops whose conformations match that string. Then, for each pseudo-conformational key in the hash table, a loop is chosen at random from the list of CDR loops associated with that key to be the medoid. Then, the chosen CDR loop and all CDR loops connected by a ML constraint to the chosen CDR loop are removed from the pool of CDR loops that could serve as potential medoids. If the number of pseudo-conformational keys is more than *k*, the number of clusters, then the program returns with a failure code since it is impossible to simultaneously satisfy all of the constraints. If *k* is larger than the number of pseudo-conformational strings, then the rest of the medoids are chosen at random from the overall pool of potential medoids. Again, whenever a CDR loop is chosen to be a medoid, both it and all CDR loops with a ML constraint to it are removed from the pool of potential medoids to avoid possibly violating a ML constraint. Now that the medoids for each cluster have been chosen, each unassigned CDR loop is matched against each medoid. The unassigned CDR loop is put in the cluster of the closest medoid whose

Constrained K-Medoids

1. Choosing initial *k* cluster medoids
   a) For each CDR loop *x*: Add *x* to the *pool* of possible initial medoids
   b) For each unique value *v* of (XLG) geometric sequence in the data set:
      i) If no more available unassigned cluster medoids, return fail
      ii) Choose CDR *x* at random from list of CDRs with geometric sequence *v*
         (1) Make *x* a medoid
         (2) Remove *x* from pool of possible medoids
         (3) **Remove all CDRs *y* from *pool* such that ML(*x,y*)**
   c) For each unassigned cluster medoid
      i) Choose CDR *x* at random from *pool* as medoid
      ii) Remove *x* from *pool* of possible initial medoids
      iii) Remove all CDRs *y* from *pool* such that ML(*x,y*)
2. While clustering has not converged:
   a) For each unassigned CDR loop *x*, assign it to the cluster of the closest medoid **that does not violate a constraint. If no possible cluster assignment, return fail**
   b) Update medoids: for each cluster, assign CDR with smallest sum of intra-cluster distances as medoid

Figure 3-1. Constrained K-Medoids Algorithm For Peptide Clustering. In the outline above, the portions in bold are those added to the original k-medoids algorithm to force this version of k-medoids to obey a list of pairwise constraints (Kaufman & Rousseeuw, 1987). The CL constraints are based on the reduced XLG alphabet for geometric sequences: if two CDR loops differ in geometric sequence at a location where one CDR has an L or and G, then there is a CL constraint between them. Similarly, if two CDR loops have the same full alphabet (ABLG) geometric sequence, they are connected by a ML constraint if each of their backbone dihedral angles is within 15 degrees of each other. The clustering converges when both steps 2a and 2b do not change either the CDR cluster assignments or the medoid assignments. Constrained k-Medoids is analogous to the COP-KMEANS algorithm of Wagstaff *et al.* (2001).

cluster does not contain a CL constraint to that CDR loop. After its assignment, then all CDR loops with a ML constraint to that CDR loop are also assigned to that cluster. This process is repeated until all of the CDR loops have been assigned to a cluster.

Both k-medoids with constraints and without constraints have been used on both the 2011 and 2017 datasets. For k-medoids with constraints, the number of clusters is varied from the minimum number (the number of unique pseudo-conformational strings present) to 19 larger than that minimum number, and for k-medoids without constraints, the number of clusters is varied from 1 to 20. Also, for each value of k the algorithm is run 20 times with starting random seeds from 1 to 20. The value of k and random seed that has the highest average silhouette coefficient is used for the final cluster analysis.

CHAPTER 4

RESULTS

The results of the clustering studies are shown here organized into sections by chain type, with the first section being for light variable (VL) domains and the second being for heavy variable (VH) domains. Comparisons to the 2011 affinity clustering study are shown first, followed by a similar analysis on the full January 2017 dataset. CDR type-lengths with fewer than 50 members are excluded. Each clustering run has a different run type label which is listed under the Run Info column. For the original results of the 2011 affinity propagation clustering, the label is 2011AFF. For the DBSCAN clustering using the sum of dihedral distances (L_1 norm), the label is DBSCAN. For the DBSCAN clustering using the supremum distance, the label is LMAX. For the SNN-DBSCAN clustering using the shared nearest neighbors (Jarvis-Patrick) similarity, the label is SNN. For the k-medoids clustering runs, the runs that use constraints are labeled KMC and the runs that do not use constraints (and so are just the standard k-medoids algorithm) are labeled KM. If the Run Info column is blank, then that line of the table denotes the properties of the CDR type-length as a whole, rather than those of a particular clustering. As for the cluster labels, they are listed in the Cluster column. For each cluster label, first the CDR loop type (L1, L2, L3, H1, H2, or H3) is listed, then the length of the loop, then a list of *cis*-conformation positions (if any), and finally the numeric cluster label. If no cluster label is listed, then that line of the table refers to that CDR type-length as a whole. Clusters are numbered in size order, from largest to smallest. After the fifth cluster, all clusters that are less than ten percent of the total size of their CDR type-length are omitted.

There are four different types of results tables in this work, of which two are included in the Results chapter (the main results tables and the silhouette tables). The other two table types (the amino acid consensus sequence tables and the tables listing the percentage of outliers) are in Appendices A and B, respectively. The main results tables contain basic information about the CDR type-length groups, their clusters, and their conformations. The first two columns, Run Info and Cluster, are described in the paragraph above. If a Run Info type field is blank, then that line refers to the properties of the listed CDR type-length as a whole, rather than that of a specific cluster. The third column, Size, is the number of members of that cluster. The next, "Avg Cl Dist," is the normalized average distance from the cluster medoid to the members of its cluster. It is normalized in the sense that it is the quotient of the average distance to the medoid and the number of non-omega (i.e., phi and psi) backbone dihedral angles, which is twice the length of the CDR loop. Since this value is normalized this way, the results of CDR type-lengths of different lengths can be compared with this statistic. The next column, "Primary Conf," is the most common geometric sequence of that cluster, and the column after that, "Conf Pct.," is the percentage of that cluster that has that most common conformation. For the "Primary Conf" column, the letters correspond to the conformational regions defined in the Introduction in Figure 1-3. All uppercase letters represent *trans* peptide bond conformations, while all lowercase letters represent *cis* peptide bond conformations. The last column of the main results tables is the "GSE" column, which lists the geometric sequence information entropy as described in Equation 2.10. A lower value implies a greater degree of conformational purity. For the silhouette tables, the first column, "CDR," is the CDR type-length. Each of the other columns is labeled with one of the "Run Info" types as described above, and each column's entries lists the average silhouette coefficient for clustering runs of that type. A silhouette value of zero indicates that there is only one cluster for that clustering run type and CDR type-length. As for the data in the appendices, Appendix A contains the consensus sequence and other information about each cluster. The first three columns of Appendix A are identical to those of the main results table as described above. The next column, "Consensus Seq," is the consensus sequence for that cluster. For each residue position, if a single amino acid type constitutes more than 90 percent of the CDR residues of that position, then it is represented by its one-letter abbreviation in uppercase. If the most prevalent amino acid for that position is between 20 and 90 percent of the total for that position, then it is shown in lowercase. All positions for which no amino acid type is at least 20 percent of the residues for that position are shown by a dash. The next column, "No. Unique Seqs," is the number of unique amino acid sequences present in that cluster. Finally, the "No. UGS" column lists the number of unique geometric sequences present for that cluster. Appendix B has the table listing the percentage of outliers out of the total number of CDR loops for each CDR type-length for the clustering runs that use DBSCAN. The first column, "CDR," lists the CDR type-length, while the other columns are labeled by the DBSCAN run type as described above (using the L_1 norm distance, the supremum distance, and finally the SNN similarity). Appendix C lists the input parameters for DBSCAN and DBSCAN-SUP in terms of the fractions of links kept to set

Eps and the fraction of data points that are core points to set MinPts. Appendix D lists input parameters for SNN, k-medoids, and k-medoids with constraints. For SNN, it lists the fraction of links kept, the fraction of data points that are core points, and the length of the nearest neighbor list. For k-medoids and k-medoids with constraints, it lists the number of clusters, k.

4.1 **The Light Chain Variable (VL) Domain CDRs**

The average silhouette coefficients for each of the CDRs considered for variable light chain (VL) domains are in Table 4-1. First, the results from the clustering of the 2011 North *et al.* dataset are shown, and then those of the 2017 dataset. In Table 4-2, the main results for the clustering of L1 CDRs using the 2011 dataset are shown, while the results for CDR L1 for the 2017 dataset are shown in Table 4-3. For CDR L1, the overall clustering results are in agreement for the composition of the larger clusters: for each CDR length-type for all of the clusters with more than 10 percent of the data, each clustering technique resulted in the same two top clusters for that CDR length-type. For instance, for L1-10, each of the clustering techniques used has the most prevalent cluster, cluster 1, having a majority geometric sequence of BBABBBABBB, while cluster 2 for each of the runs has a majority geometric sequence of BBABBBBLBB. DBSCAN (using the L1 norm; hereafter referred to as just "DBSCAN") and DBSCAN using the supremum distance (hereafter referred to as "DBSCAN-SUP") both have only one geometric sequence per cluster for their two clusters of L1-10, and therefore both clusters for them have a geometric sequence entropy of zero. k-medoids with constraints has a geometric sequence entropy of 0.06 for L1-10-1 and zero for L1-10-2. K-medoids

Table 4-1

Average Silhouette for the VL CDRs

CDR	DBSCAN	LMAX	SNN	KMC	KM	2011AFF
2011 Dataset						
L1-11	0.93	0.93	0.92	0.69	0.86	0.86
L1-16	0.55	0.51	0.42	0.50	0.81	0.00
L2-8	0.66	0.66	0.74	0.24	0.88	0.80
L3-9-cis7	0.73	0.72	0.48	0.82	0.80	0.82
2017 Dataset						
L1-10	0.89	0.89	0.85	0.72	0.83	
L1-11	0.91	0.91	0.92	0.40	0.63	
L1-12	0.92	0.91	0.88	0.54	0.71	
L1-13	0.93	0.93	0.92	0.69	0.75	
L1-14	0.95	0.95	0.94	0.37	0.71	
L1-15	0.85	0.85	0.81	0.26	0.73	
L1-16	0.84	0.84	0.81	0.29	0.73	
L1-17	0.79	0.75	0.77	0.61	0.71	
L2-8	0.82	0.76	0.36	0.16	0.80	
L3-5	0.69	0.68	0.62	0.45	0.36	
L3-8	0.96	0.96	0.92	0.76	0.74	
L3-9-cis7	0.74	0.73	0.56	0.37	0.87	
L3-9	0.89	0.88	0.74	0.31	0.54	
L3-10	0.83	0.79	0.72	0.60	0.40	
L3-11	0.83	0.85	0.85	0.40	0.57	
L3-12	0.96	0.96	0.57	0.71	0.65	

Run types (column labels) are defined at the start of the Results chapter.

Table 4-2

CDR L1 Clustering Results for the 2011 Dataset

Run Info[a]	Cluster	Size	Avg Cl Dist[b]	Primary Conf	Conf Pct.	GSE
	L1-11	134	0.44	BBABBAGABBB	54	1.42
2011AFF	L1-11-1	74	0.07	BBABBAGABBB	97	0.21
2011AFF	L1-11-2	55	0.05	BBABBALLBBB	98	0.13
2011AFF	L1-11-3	5	0.56	BBBLAAABBBB	40	1.92
DBSCAN	L1-11-1	45	0.03	BBABBAGABBB	98	0.15
DBSCAN	L1-11-2	31	0.02	BBABBALLBBB	100	0.00
LMAX	L1-11-1	41	0.03	BBABBAGABBB	100	0.00
LMAX	L1-11-2	33	0.03	BBABBALLBBB	100	0.00
SNN	L1-11-1	52	0.04	BBABBAGABBB	98	0.14
SNN	L1-11-2	40	0.03	BBABBALLBBB	100	0.00
KMC	L1-11-1	74	0.07	BBABBAGABBB	97	0.21
KMC	L1-11-2	52	0.04	BBABBALLBBB	100	0.00
KMC	L1-11-3	2	0.07	BBABBALLBBB	100	0.00
KMC	L1-11-4	2	0.75	BBBBBAABBBB	50	1.00
KMC	L1-11-5	2	0.02	BBBLAAABBBB	100	0.00
KM	L1-11-1	74	0.07	BBABBAGABBB	97	0.21
KM	L1-11-2	55	0.05	BBABBALLBBB	98	0.13
KM	L1-11-3	5	0.56	BBBLAAABBBB	40	1.92
	L1-16	64	0.18	BBABBABBAALBBBBB	80	1.36
2011AFF	L1-16-1	64	0.18	BBABBABBAALBBBBB	80	1.36
DBSCAN	L1-16-1	29	0.04	BBABBABBAALBBBBB	100	0.00
DBSCAN	L1-16-2	3	0.01	BBABBBBBAALBBBBB	100	0.00
LMAX	L1-16-1	31	0.04	BBABBABBAALBBBBB	100	0.00
LMAX	L1-16-2	4	0.02	BBABBBBBAALBBBBB	75	0.81
SNN	L1-16-1	36	0.04	BBABBABBAALBBBBB	100	0.00
SNN	L1-16-2	4	0.06	BBABBABBAALBBBBB	50	1.00
KMC	L1-16-1	14	0.03	BBABBABBAALBBBBB	100	0.00
KMC	L1-16-2	8	0.03	BBABBABBAALBBBBB	100	0.00
KMC	L1-16-3	5	0.04	BBABBABBAALBBBBB	100	0.00
KMC	L1-16-4	4	0.03	BBABBBBBAALBBBBB	100	0.00
KMC	L1-16-5	3	0.04	BBABBABBAALBBBBB	100	0.00
KM	L1-16-1	63	0.16	BBABBABBAALBBBBB	81	1.26
KM	L1-16-2	1	0.00	BBABBABBBLBBBBBB	100	0.00

[a]If blank, refers to entire CDR type-length. Details on columns on pages 35-36

[b]Average Normalized Distance to cluster medoid

Table 4-3

CDR L1 Clustering Results for the 2017 Dataset

Run Info	Cluster	Size	Avg Cl Dist	Primary Conf	Conf Pct.	GSE
	L1-10	229	0.58	BBABBBABBB	59	1.49
DBSCAN	L1-10-1	86	0.05	BBABBBABBB	100	0.00
DBSCAN	L1-10-2	41	0.04	BBABBBBLBB	100	0.00
LMAX	L1-10-1	93	0.06	BBABBBABBB	100	0.00
LMAX	L1-10-2	37	0.05	BBABBBBLBB	100	0.00
SNN	L1-10-1	128	0.08	BBABBBABBB	100	0.00
SNN	L1-10-2	80	0.12	BBABBBBLBB	94	0.38
KMC	L1-10-1	135	0.10	BBABBBABBB	99	0.06
KMC	L1-10-2	75	0.09	BBABBBBLBB	100	0.00
KMC	L1-10-3	5	0.01	BBABBLBBBB	100	0.00
KMC	L1-10-4	4	0.19	BBABBBBGBB	100	0.00
KMC	L1-10-5	4	0.05	BBABBLBBBB	100	0.00
KM	L1-10-1	139	0.11	BBABBBABBB	96	0.28
KM	L1-10-2	80	0.12	BBABBBBLBB	94	0.38
KM	L1-10-3	10	0.40	BBABBLBBBB	90	0.47
	L1-11	1839	0.60	BBABBAGABBB	55	2.71
DBSCAN	L1-11-1	798	0.04	BBABBAGABBB	97	0.20
DBSCAN	L1-11-2	184	0.03	BBABBALLBBB	100	0.00
LMAX	L1-11-1	847	0.04	BBABBAGABBB	97	0.21
LMAX	L1-11-2	234	0.04	BBABBALLBBB	100	0.00
SNN	L1-11-1	201	0.03	BBABBALLBBB	100	0.00
SNN	L1-11-2	12	0.01	BBABBAGABBB	100	0.00
KMC	L1-11-1	986	0.06	BBABBAGABBB	94	0.44
KMC	L1-11-2	359	0.08	BBABBALLBBB	93	0.46
KMC	L1-11-3	84	0.10	BBBLAAABBBB	90	0.45
KMC	L1-11-4	57	0.20	BBABBAGABBB	75	1.25
KMC	L1-11-5	55	0.14	BBAAAAABBBB	75	1.01
KM	L1-11-1	1209	0.21	BBABBAGABBB	84	1.34
KM	L1-11-2	427	0.18	BBABBALLBBB	82	1.22
KM	L1-11-3	203	0.95	BBBLAAABBBB	36	3.27
	L1-12	295	0.67	BBABBBBLLBBB	34	2.80
DBSCAN	L1-12-1	80	0.03	BBABBBBLLBBB	99	0.10
DBSCAN	L1-12-2	47	0.05	BBABBBBAAABB	100	0.00
LMAX	L1-12-1	83	0.04	BBABBBBLLBBB	99	0.09
LMAX	L1-12-2	56	0.06	BBABBBBAAABB	100	0.00

Table 4-3 *(continued)*						
Run Info	Cluster	Size	Avg Cl Dist	Primary Conf	Conf Pct.	GSE
SNN	L1-12-1	62	0.08	BBABBBBAAABB	92	0.40
SNN	L1-12-2	56	0.04	BBABBBBLLBBB	98	0.13
KMC	L1-12-1	116	0.17	BBABBBBAAABB	85	0.86
KMC	L1-12-2	91	0.06	BBABBBBLLBBB	100	0.00
KMC	L1-12-3	24	0.09	BBABBBAALBBB	100	0.00
KMC	L1-12-4	17	0.10	BBBAAAAABBBB	94	0.32
KMC	L1-12-5	10	0.10	BBABBBBLLBBB	90	0.47
KM	L1-12-1	115	0.16	BBABBBBAAABB	86	0.90
KM	L1-12-2	107	0.10	BBABBBBLLBBB	93	0.44
KM	L1-12-3	29	0.22	BBABBBAALBBB	83	0.90
KM	L1-12-4	18	0.23	BBBAAAAABBBB	89	0.61
KM	L1-12-5	14	0.34	BBABBAGAAABB	64	1.48
	L1-13	256	0.73	BBBAAAAAABBBB	57	2.34
DBSCAN	L1-13-1	94	0.04	BBBAAAAAABBBB	99	0.09
DBSCAN	L1-13-2	32	0.03	BBABGBAAABBBB	100	0.00
LMAX	L1-13-1	105	0.05	BBBAAAAAABBBB	98	0.14
LMAX	L1-13-2	33	0.03	BBABGBAAABBBB	100	0.00
SNN	L1-13-1	115	0.06	BBBAAAAAABBBB	97	0.17
SNN	L1-13-2	28	0.05	BBABGBAAABBBB	100	0.00
KMC	L1-13-1	152	0.10	BBBAAAAAABBBB	95	0.32
KMC	L1-13-2	41	0.06	BBABGBAAABBBB	98	0.17
KMC	L1-13-3	19	0.04	BBABBBBLLABBB	100	0.00
KMC	L1-13-4	11	0.04	BBBAAAAAALBBB	100	0.00
KMC	L1-13-5	7	0.02	BBBLGBAAABBBB	100	0.00
KM	L1-13-1	186	0.23	BBBAAAAAABBBB	79	1.43
KM	L1-13-2	70	0.62	BBABGBAAABBBB	57	1.65
	L1-14	367	1.48	BBAAGBBBAAALBB	37	3.92
DBSCAN	L1-14-1	115	0.06	BBAAGBBBAAALBB	98	0.13
DBSCAN	L1-14-2	47	0.04	BBBAAAAAABBBBB	38	1.96
LMAX	L1-14-1	115	0.06	BBAAGBBBAAALBB	97	0.17
LMAX	L1-14-2	46	0.04	BBBAAAAAABBBBB	37	2.04
SNN	L1-14-1	110	0.06	BBAAGBBBAAALBB	98	0.13
SNN	L1-14-2	32	0.06	BBBAAAAAABBABB	38	2.13
KMC	L1-14-1	121	0.09	BBAAGBBBAAALBB	88	0.60
KMC	L1-14-2	85	0.11	BBBAAAAAABABBB	34	2.50

Table 4-3 *(continued)*						
Run Info	Cluster	Size	Avg Cl Dist	Primary Conf	Conf Pct.	GSE
KMC	L1-14-3	37	0.13	BBAAGBBBAAALBB	76	1.10
KMC	L1-14-4	25	0.13	BBBAAAAAABAABB	60	0.97
KMC	L1-14-5	18	0.12	BBBAAAAAALBBBB	83	0.80
KM	L1-14-1	201	0.69	BBBAAAAAABABBB	21	4.40
KM	L1-14-2	166	0.16	BBAAGBBBAAALBB	81	1.13
	L1-15	205	0.38	BBABBABBLLBBBBB	57	2.70
DBSCAN	L1-15-1	108	0.08	BBABBABBLLBBBBB	94	0.44
DBSCAN	L1-15-2	17	0.05	BBABBABBAABBBBB	100	0.00
LMAX	L1-15-1	107	0.08	BBABBABBLLBBBBB	93	0.44
LMAX	L1-15-2	13	0.04	BBABBABBAABBBBB	100	0.00
SNN	L1-15-1	114	0.09	BBABBABBLLBBBBB	90	0.60
SNN	L1-15-2	5	0.22	BBABBABBAABBBBB	60	0.97
KMC	L1-15-1	50	0.09	BBABBABBLLBBBBB	92	0.47
KMC	L1-15-2	50	0.08	BBABBABBLLBBBBB	86	0.78
KMC	L1-15-3	25	0.13	BBABBABBLLBBBBB	72	1.32
KMC	L1-15-4	25	0.11	BBABBABBAABBBBB	100	0.00
KMC	L1-15-5	17	0.32	BBABBABBLLBBBBB	53	1.00
KM	L1-15-1	203	0.36	BBABBABBLLBBBBB	57	2.64
KM	L1-15-2	2	1.16	BBAGBAABAGABBBG	50	1.00
	L1-16	516	0.18	BBABBABBAALBBBBB	82	1.51
DBSCAN	L1-16-1	433	0.07	BBABBABBAALBBBBB	94	0.38
DBSCAN	L1-16-2	7	0.01	BBABBBBBBLLBBBBB	100	0.00
LMAX	L1-16-1	422	0.07	BBABBABBAALBBBBB	94	0.39
LMAX	L1-16-2	7	0.01	BBABBBBBBLLBBBBB	100	0.00
SNN	L1-16-1	371	0.06	BBABBABBAALBBBBB	94	0.36
SNN	L1-16-2	8	0.12	BBABBBBBBLLBBBBB	75	0.81
KMC	L1-16-1	347	0.06	BBABBABBAALBBBBB	97	0.24
KMC	L1-16-2	58	0.16	BBABBABBAALBBBBB	69	1.39
KMC	L1-16-3	22	0.12	BBABBABBAALBBBBB	86	0.70
KMC	L1-16-4	14	0.26	BBABBABBBLLBBBBB	50	1.00
KMC	L1-16-5	14	0.20	BBABBABBAALBBBBB	93	0.37
KM	L1-16-1	515	0.18	BBABBABBAALBBBBB	82	1.50
KM	L1-16-2	1	0.00	BBAGBABGAAGBBBBB	100	0.00
	L1-17	249	0.18	BBABBABBAAALBBBBB	83	1.30
DBSCAN	L1-17-1	152	0.06	BBABBABBAAALBBBBB	100	0.00

without constraints has the most mixed set of geometric sequences with an entropy of 0.28 for cluster L1-10-1. For cluster L1-10-2, interestingly, SNN-DBSCAN and k-medoids without constraints resulted in the same clustering. Each is also relatively spread out with a geometric sequence entropy of 0.38, with their top geometric sequence being 94 percent of the cluster. This is a high value, but still less than the other clustering runs. Finally, only the k-medoids runs, both of which are without outlier removal, have more than two clusters. K-medoids without constraints has only one more cluster, L1-10-3, that is much more spread out than the others. It has a geometric sequence entropy of 0.47 and a normalized average distance to the medoid of 0.4, which is about three times that of the larger clusters. After the two largest clusters, k-medoids without constraints has more clusters than k-medoids. They are lower in sequence entropy and in normalized average distance from the medoid, but they are also very small at five members or fewer.

The other CDRs of L1 mostly follow the same pattern of L1-10. The largest clusters for each of the clustering runs are the same for each CDR length-type. As for the silhouette coefficients, there are some variations. For CDR type-lengths L1-10, L1-12, L1-13, L1-14, L1-15, and L1-16, DBSCAN and DBSCAN-SUP are the best, both of which perform approximately as well as each other according to average silhouette values. Next is SNN-DBSCAN, although it is not far behind (the exception is L1-17, where the silhouette ranking of SNN-DBSCAN is better than DBSCAN-SUP). K-medoids without constraints is next in performance as judged by silhouette, although as seen in L1-10, it has a higher geometric sequence entropy than k-medoids with constraints. Finally k-medoids with constraints is the worst of the performers in terms of average silhouette coefficient, with its values ranging from middling to terrible. L1-15 and L1-16 are particularly bad for k-medoids with constraints, with overall average silhouette values of 0.26 and 0.29. The affinity clustering of the 2011 dataset has about the same performance as k-medoids without constraints for L1-11, while for L1-16, affinity clustering placed all of the data points into a single cluster. L-11 is interesting in that it is the only L1 CDR type for which SNN-DBSCAN has the highest average silhouette coefficient. Also of note is that for the 2011 dataset, cluster L1-11-3 of the affinity clustering is removed as outliers by all of the clustering runs using DBSCAN, as expected. Cluster L1-11-3 of the affinity clustering is much more diffuse than the other clusters of L-11 by the affinity clustering.

For CDR L2, CDR type-length L2-8 predominates, with none of the other CDR type-lengths having enough members to meet the threshold for display. The data for the CDR type-length L2-8 clustering results are shown in Table 4-4, with the results of the analysis of the 2011 dataset shown first, followed by those of the 2017 dataset. For the 2011 dataset, k-medoids without constraints has the highest average silhouette coefficient at 0.88, followed by the affinity clustering at 0.80. SNN-DBSCAN is next at 0.74, followed by DBSCAN and DBSCAN-SUP at 0.66. Finally, k-medoids with constriants has the lowest average silhouette coefficient at a very low 0.24. For the larger 2017 dataset, DBSCAN, DBSCAN-SUP, and k-medoids without constraints are all relatively on par with one another, with DBSCAN being the best with an average silhouette coefficient at 0.82. k-medoids with constraints is again the worst performer at 0.16. Oddly, SNN-DBSCAN has a much worse result when clustering the 2017 dataset than the 2011 dataset, with an average silhouette of 0.36. The main clustering results in Table 4-4 also has some unexpected results. For the 2011 dataset, the largest cluster for each clustering run, L2-8-1, has a majority geometric sequence of BLLABBBB. K-medoids with constraints is the only clustering method that has a geometric sequence entropy of zero, while the others are considerably higher, with the majority geometric sequence being around 67 percent (for k-medoids) to 70 to 71 percent (for DBSCAN, DBSCAN-SUP, SNN-DBSCAN, and affinity clustering). K-medoids with constraints is the only clustering method that has more than one major cluster for L2-8, with its L2-8-2 having 73 members. That cluster has a geometric sequence of BLGABBBB for 97 percent of its members which results in a geometric sequence entropy of 0.18. As for the 2017 dataset, the oddest result is that the SNN-DBSCAN average silhouette is so low in comparison to the 2011 dataset. The reason becomes clearer when examining the outlier percentages. For the 2017 dataset, SNN-DBSCAN marks fully 88 percent of the members of L2-8 as outliers. This is very high, and is much higher than the outlier percentages for DBSCAN (28 percent) and DBSCAN-SUP (13 percent). Also, these outliers removed by SNN-DBSCAN are mostly from the most common geometric sequence of L2-8, which is BLLABBBB. So, unlike all of the other clustering methods, SNN-DBSCAN has the geometric sequence BLLABBBA as a majority for L2-8-1, while all of the others have BLLABBBB. For the other clustering methods, DBSCAN, DBSCAN-SUP, and k-medoids all consist mostly of one large cluster, cluster L2-8-1, and much smaller clusters after that. For DBSCAN and DBSCAN-SUP, its second largest cluster is very close to its largest for the 2011 dataset clustering of L2-8, which is why their average silhouette coefficient values are relatively low. K-medoids has both the largest geometric sequence

Table 4-4

CDR L2 Clustering Results

Run Info	Cluster	Size	Avg Cl Dist	Primary Conf	Conf Pct.	GSE
2011 Dataset						
	L2-8	295	0.11	BLLABBBB	67	1.30
2011AFF	L2-8-1	279	0.06	BLLABBBB	70	0.93
2011AFF	L2-8-2	9	0.10	BLLABBBA	67	1.22
2011AFF	L2-8-3	3	0.37	BBLLBBBB	67	0.92
2011AFF	L2-8-4	2	0.10	BGAABBBB	100	0.00
2011AFF	L2-8-5	2	0.45	ABAABBBB	50	1.00
DBSCAN	L2-8-1	186	0.03	BLLABBBB	71	0.87
DBSCAN	L2-8-2	10	0.01	BLLABBBB	90	0.47
LMAX	L2-8-1	193	0.03	BLLABBBB	70	0.88
LMAX	L2-8-2	10	0.01	BLLABBBB	90	0.47
SNN	L2-8-1	141	0.03	BLLABBBB	71	0.87
SNN	L2-8-2	4	0.06	BLGABBBA	50	1.50
KMC	L2-8-1	155	0.04	BLLABBBB	100	0.00
KMC	L2-8-2	73	0.06	BLGABBBB	97	0.18
KMC	L2-8-3	12	0.02	BLLABBBB	100	0.00
KMC	L2-8-4	8	0.10	BLGABBBB	100	0.00
KMC	L2-8-5	7	0.15	BLLABBBA	86	0.59
KM	L2-8-1	291	0.09	BLLABBBB	68	1.19
KM	L2-8-2	4	0.77	BGAABBBB	50	1.50
2017 Dataset						
	L2-8	4038	0.23	BLLABBBB	61	1.95
DBSCAN	L2-8-1	2880	0.06	BLLABBBB	72	0.86
DBSCAN	L2-8-2	21	0.02	BLGABBBA	57	0.99
LMAX	L2-8-1	3420	0.08	BLLABBBB	70	0.93
LMAX	L2-8-2	107	0.08	BLLABBBA	60	0.97
SNN	L2-8-1	132	0.10	BLLABBBA	60	1.17
SNN	L2-8-2	69	0.27	BGAABBBB	65	1.21
SNN	L2-8-3	61	0.02	BLLABBBB	62	0.96
SNN	L2-8-4	48	0.01	BLLABBBB	88	0.54
SNN	L2-8-5	13	0.02	BLLABBBB	77	0.78

Table 4-4

(2017 dataset results, continued)

Run Info	Cluster	Size	Avg Cl Dist	Primary Conf	Conf Pct.	GSE
KMC	L2-8-1	2221	0.06	BLLABBBB	100	0.03
KMC	L2-8-2	1062	0.10	BLGABBBB	98	0.15
KMC	L2-8-3	100	0.17	BLLABBBB	100	0.00
KMC	L2-8-4	99	0.17	BLLABBBA	91	0.67
KMC	L2-8-5	94	0.30	BGAABBBB	68	1.43
KM	L2-8-1	3951	0.19	BLLABBBB	62	1.77
KM	L2-8-2	87	0.90	ABAABBBB	31	3.59

entropy, at 1.77, and the largest average normalized distance to medoid at 0.19 out of all of the clustering methods. As in the 2011 dataset, for the 2017 dataset k-medoids with constraints has a large second cluster, L2-8-2, with 1,062 members. Clusters L2-8-1 and L2-8-2 for k-medoids with constraints are both tightly clustered, with low average normalized distances to their medoids.

CDR type L3 has only one CDR type-length, L3-9-cis7, with enough data from the 2011 dataset to be included. However, the 2017 dataset has far more CDR type-lengths with enough data. The data for CDR L3 are shown in Tables 4-5 and 4-6. They include L3-5, L3-8, L3-9 (without *cis* peptide bonds), L3-10, L3-11, and L3-12. In terms of the average silhouette coefficient for the 2017 dataset, L3-5, L3-8, L3-9, and L3-10 have DBSCAN and DBSCAN-SUP with the best values, then SNN-DBSCAN, and finally the k-medoids and k-medoids with constraints as the worst. For L3-11, SNN-DBSCAN is tied with DBSCAN-SUP with the highest average silhouette coefficient, while for L3-12, SNN-DBSCAN has the lowest average silhouette coefficient out of all

Table 4-5

CDR L3 Clustering Results for the 2011 Dataset

Run Info	Cluster	Size	Avg Cl Dist	Primary Conf	Conf Pct.	GSE
	L3-9-cis7	219	0.15	BBAABBbBB	93	0.54
2011AFF	L3-9-cis7-1	209	0.10	BBAABBbBB	97	0.22
2011AFF	L3-9-cis7-2	8	0.14	BBABGBbBB	88	0.54
2011AFF	L3-9-cis7-3	2	0.28	BBAABBaLB	50	1.00
DBSCAN	L3-9-cis7-1	176	0.07	BBAABBbBB	99	0.09
DBSCAN	L3-9-cis7-2	6	0.01	BBABGBbBB	100	0.00
LMAX	L3-9-cis7-1	168	0.06	BBAABBbBB	99	0.05
LMAX	L3-9-cis7-2	6	0.01	BBABGBbBB	100	0.00
SNN	L3-9-cis7-1	118	0.05	BBAABBbBB	100	0.00
SNN	L3-9-cis7-2	6	0.06	BBAABBbBB	100	0.00
SNN	L3-9-cis7-3	5	0.07	BBAABBbBB	100	0.00
SNN	L3-9-cis7-4	3	0.06	BBAABBbBB	100	0.00
KMC	L3-9-cis7-1	209	0.10	BBAABBbBB	97	0.22
KMC	L3-9-cis7-2	8	0.14	BBABGBbBB	88	0.54
KMC	L3-9-cis7-3	1	0.00	BBAABBaLB	100	0.00
KMC	L3-9-cis7-4	1	0.00	BBAABBaGB	100	0.00
KM	L3-9-cis7-1	217	0.14	BBAABBbBB	94	0.46
KM	L3-9-cis7-2	2	0.28	BBAABBaLB	50	1.00

Table 4-6

CDR L3 Clustering Results for the 2017 Dataset

Run Info	Cluster	Size	Avg Cl Dist	Primary Conf	Conf Pct.	GSE
	L3-5	99	0.06	BBGAB	100	0.00
DBSCAN	L3-5-1	43	0.02	BBGAB	100	0.00
DBSCAN	L3-5-2	6	0.01	BBGAB	100	0.00
LMAX	L3-5-1	47	0.02	BBGAB	100	0.00
LMAX	L3-5-2	6	0.01	BBGAB	100	0.00
SNN	L3-5-1	43	0.02	BBGAB	100	0.00
SNN	L3-5-2	4	0.03	BBGAB	100	0.00
KMC	L3-5-1	95	0.05	BBGAB	100	0.00
KMC	L3-5-2	4	0.04	BBGAB	100	0.00
KM	L3-5-1	74	0.05	BBGAB	100	0.00
KM	L3-5-2	25	0.06	BBGAB	100	0.00
	L3-8	236	0.94	BBAABGBB	48	2.50
DBSCAN	L3-8-1	71	0.02	BBAABGBB	100	0.00
DBSCAN	L3-8-2	24	0.02	BBAABBBB	100	0.00
LMAX	L3-8-1	73	0.03	BBAABGBB	100	0.00
LMAX	L3-8-2	27	0.02	BBAABBBB	100	0.00
LMAX	L3-8-3	13	0.01	BBABGALB	100	0.00
SNN	L3-8-1	101	0.05	BBAABGBB	100	0.00
SNN	L3-8-2	7	0.02	BBAABBBB	100	0.00
KMC	L3-8-1	116	0.09	BBAABGBB	98	0.13
KMC	L3-8-2	59	0.12	BBAABBBB	95	0.34
KMC	L3-8-3	18	0.07	BBABGALB	89	0.50
KMC	L3-8-4	10	0.29	BBAAABBB	60	1.30
KMC	L3-8-5	9	0.46	BBBAABBB	56	1.35
KM	L3-8-1	119	0.11	BBAABGBB	96	0.32
KM	L3-8-2	59	0.13	BBAABBBB	95	0.37
KM	L3-8-3	23	0.39	BBABGALB	70	1.07
KM	L3-8-4	14	0.63	BBAAABBB	36	2.16
KM	L3-8-5	11	0.33	BBBBABBB	45	1.68
	L3-9-cis7	2269	0.16	BBAABBbBB	92	0.66
DBSCAN	L3-9-cis7-1	2009	0.10	BBAABBbBB	97	0.19
DBSCAN	L3-9-cis7-2	26	0.06	BBABGBbBB	100	0.00
LMAX	L3-9-cis7-1	1989	0.10	BBAABBbBB	97	0.18
LMAX	L3-9-cis7-2	23	0.05	BBABGBbBB	100	0.00
SNN	L3-9-cis7-1	979	0.06	BBAABBbBB	100	0.04

Table 4-6

(continued)

Run Info	Cluster	Size	Avg Cl Dist	Primary Conf	Conf Pct.	GSE
SNN	L3-9-cis7-2	19	0.02	BBAABBbBB	100	0.00
SNN	L3-9-cis7-3	6	0.05	BBAABBbBB	100	0.00
KMC	L3-9-cis7-1	1877	0.09	BBAABBbBB	97	0.20
KMC	L3-9-cis7-2	162	0.47	BBAABBbBB	46	1.85
KMC	L3-9-cis7-3	102	0.24	BBAABBbBB	99	0.08
KMC	L3-9-cis7-4	85	0.21	BBAABBbBB	93	0.46
KMC	L3-9-cis7-5	20	0.17	BBAABBaLB	80	0.72
KM	L3-9-cis7-1	2266	0.16	BBAABBbBB	92	0.65
KM	L3-9-cis7-2	3	1.35	BBALLGbBB	33	1.59
	L3-9	568	1.27	BBAABABBB	31	3.85
DBSCAN	L3-9-1	141	0.13	BBAABABBB	96	0.28
DBSCAN	L3-9-2	87	0.07	BBBBGABBB	59	1.18
LMAX	L3-9-1	148	0.14	BBAABABBB	97	0.21
LMAX	L3-9-2	91	0.08	BBBBGABBB	56	1.37
SNN	L3-9-1	115	0.13	BBBBGABBB	50	1.94
SNN	L3-9-2	41	0.23	BBBBLLBBB	76	1.23
KMC	L3-9-1	104	0.18	BBAABABBB	91	0.53
KMC	L3-9-2	92	0.23	BBAABABBB	87	0.67
KMC	L3-9-3	53	0.20	BBBBAABBB	94	0.37
KMC	L3-9-4	51	0.06	BBBBGABBB	96	0.28
KMC	L3-9-5	44	0.13	BBBBLLBBB	95	0.31
KM	L3-9-1	315	0.67	BBBBGABBB	24	3.43
KM	L3-9-2	253	0.58	BBAABABBB	70	2.24
	L3-10	257	1.09	BBBBAALBBB	33	4.14
DBSCAN	L3-10-1	81	0.18	BBBBAALBBB	100	0.00
DBSCAN	L3-10-2	14	0.07	BBBBAAAGBB	93	0.37
LMAX	L3-10-1	81	0.19	BBBBAALBBB	100	0.00
LMAX	L3-10-2	16	0.09	BBBBAAAGBB	94	0.34
LMAX	L3-10-3	11	0.06	BBBBBLLBBB	100	0.00
SNN	L3-10-1	78	0.26	BBBBAALBBB	90	0.73
SNN	L3-10-2	2	0.70	BBBAGLLBBB	50	1.00
SNN	L3-10-3	2	0.13	BBBAGLLBBB	50	1.00
KMC	L3-10-1	60	0.14	BBBBAALBBB	98	0.12
KMC	L3-10-2	27	0.13	BBBBAALBBB	93	0.38
KMC	L3-10-3	25	0.23	BBBBAAAGBB	100	0.00

Table 4-6 *(continued)*						
Run Info	Cluster	Size	Avg Cl Dist	Primary Conf	Conf Pct.	GSE
KM	L3-10-1	122	0.46	BBBBAALBBB	70	1.72
KM	L3-10-2	40	0.98	BBBAGLLBBB	30	2.89
KM	L3-10-3	35	0.69	BBBBAAAGBB	60	1.75
KM	L3-10-4	28	0.80	BBBBAABBBB	32	2.82
KM	L3-10-5	21	0.70	BBBABAABBB	57	2.12
	L3-11	289	0.68	BBBBAAALBBB	60	2.72
DBSCAN	L3-11-1	161	0.14	BBBBAAALBBB	96	0.26
DBSCAN	L3-11-2	13	0.04	BBBBAABABBB	100	0.00
LMAX	L3-11-1	152	0.12	BBBBAAALBBB	95	0.27
LMAX	L3-11-2	13	0.04	BBBBAABABBB	100	0.00
SNN	L3-11-1	112	0.10	BBBBAAALBBB	96	0.22
SNN	L3-11-2	4	0.33	BBBBABABABB	75	0.81
KMC	L3-11-1	47	0.04	BBBBAAALBBB	98	0.15
KMC	L3-11-2	21	0.10	BBBBAAALBBB	90	0.45
KMC	L3-11-3	20	0.13	BBBBAAALBBB	95	0.29
KMC	L3-11-4	18	0.08	BBBBAAALBBB	100	0.00
KMC	L3-11-5	17	0.05	BBBBAAALBBB	100	0.00
KM	L3-11-1	243	0.44	BBBBAAALBBB	72	1.82
KM	L3-11-2	46	1.15	BBBBABABABB	22	3.55
	L3-12	108	1.61	BBBBAABBBBAB	18	3.75
DBSCAN	L3-12-1	15	0.07	BBBBAABBBBAB	100	0.00
DBSCAN	L3-12-2	13	0.03	BBBABBLBABBB	100	0.00
LMAX	L3-12-1	16	0.09	BBBBAABBBBAB	100	0.00
LMAX	L3-12-2	13	0.03	BBBABBLBABBB	100	0.00
LMAX	L3-12-3	10	0.04	BBAABABLBABA	100	0.00
LMAX	L3-12-4	8	0.00	BBBBABAALBBB	100	0.00
LMAX	L3-12-5	7	0.04	BBBBABAALBBB	100	0.00
SNN	L3-12-1	28	0.65	BBBBABAALBBB	50	2.18
SNN	L3-12-2	7	0.60	BBBBAABBBBAB	71	1.15

Table 4-6

(continued)

Run Info	Cluster	Size	Avg Cl Dist	Primary Conf	Conf Pct.	GSE
KMC	L3-12-1	18	0.13	BBBBAABBBBAB	94	0.31
KMC	L3-12-2	17	0.30	BBBBABAALBBB	94	0.32
KMC	L3-12-3	13	0.03	BBBABBLBABBB	100	0.00
KMC	L3-12-4	12	0.20	BBBBAAAGBBAB	100	0.00
KMC	L3-12-5	10	0.04	BBAABABLBABA	100	0.00
KM	L3-12-1	25	0.46	BBBBABAALBBB	64	1.70
KM	L3-12-2	20	0.15	BBBBAABBBBAB	95	0.29
KM	L3-12-3	14	0.79	BBGBAABAAGBB	50	2.06
KM	L3-12-4	14	0.05	BBBABBLBABBB	93	0.37
KM	L3-12-5	12	0.37	BBBBABGBBBBB	58	1.78
KM	L3-12-6	12	0.20	BBBBAAAGBBAB	100	0.00

of the clustering methods tested. For the most prevalent CDR type-length, L3-9-cis7, k-medoids with constraints is tied for the highest average silhouette coefficient for the 2011 dataset, but has the lowest average silhouette coefficient for the 2017 dataset. There are 28 unique geometric sequences present in the 2017 dataset of L3-9-cis7, while there are only 8 such geometric sequences in the 2011 dataset, which may help to explain why the performance of k-medoids with constraints is so much worse for the 2017 dataset, since it has to have a separate cluster to cover each unique geometric sequence of the reduced XLG alphabet. As for the clustering results, for each of the clustering methods for each L3 CDR type-length except for L3-9 and L3-12, the most common geometric sequence for the dataset is also the most common geometric sequence for the largest cluster. L3-9 has two exceptions to that rule. The CDR type-length as a whole has the geometric sequence BBAABABBB as its most prevalent at 31 percent. However, both SNN-

DBSCAN and k-medoids (without constraints) has BBBBGABBB as the most prevalent geometric sequences for their largest L3-9 clusters (L3-9-1). SNN-DBSCAN and k-medoids also both have a large geometric sequence entropy for L3-9-1, at 1.9 and 3.4, respectively, while those of L3-9-1 for DBSCAN, DBSCAN-SUP, and k-medoids with constraints are much lower. However, the difference between SNN-DBSCAN and k-medoids for L3-9-1 is that the k-medoids cluster L3-9-1 is much more scattered than that of SNN-DBSCAN, with a normalized average distance of 0.67 to the medoid for k-medoids versus a value of 0.13 for SNN-DBSCAN. Despite the fact that they share their most common geometric sequence, the ability of SNN-DBSCAN to designate outliers causes it to have tighter clusters than k-medoids when clustering the same regions of a dataset. As for L3-12, SNN-DBSCAN and k-medoids both have a most prevalent geometric sequence of BBBBABAALBBB, while the rest of the clustering methods share their most prevalent geometric sequence with that of the CDR type-length L3-12 in general. Also note that the cluster L3-12-1 for SNN-DBSCAN and k-medoids both have a high geometric sequence entropy (2.2 and 1.7) and normalized average distance to their medoids (0.65 for SNN-DBSCAN and 0.46 for k-medoids), indicating fairly spread out clusters.

4.2 The Heavy Chain Variable (VH) Domain CDRs

As for the light antibody chains, the average silhouette coefficients are collected together into a single group, shown as Table 4-7. For H1-13, the only CDR type-length of CDR type H1 that has enough data to show from the 2011 dataset, DBSCAN, DBSCAN-SUP, and k-medoids are all nearly tied for the largest average silhouette

Table 4-7

Average Silhouette for VH CDRs

CDR	DBSCAN	LMAX	SNN	KMC	KM	2011AFF
2011 Dataset						
H1-13	0.78	0.77	0.52	0.44	0.75	0.68
H2-9	0.52	0.44	0.50	0.84	0.86	0.85
H2-10	0.87	0.86	0.84	0.73	0.69	0.75
H3-12	0.71	0.72	0.35	0.95	0.51	0.26
2017 Dataset						
H1-13	0.78	0.74	0.23	0.14	0.67	
H1-14	0.92	0.91	0.55	0.48	0.55	
H1-15	0.87	0.86	0.85	0.49	0.74	
H2-9	0.80	0.52	0.79	0.15	0.84	
H2-10	0.85	0.85	0.63	0.30	0.61	
H2-12	0.46	0.44	0.85	0.39	0.86	
H3-5	0.98	0.98	0.85	0.84	0.80	
H3-6	0.92	0.93	0.84	0.78	0.70	
H3-7	0.97	0.97	0.73	0.82	0.69	
H3-8	0.98	0.98	0.83	0.92	0.71	
H3-9	0.66	0.66	0.56	0.56	0.50	
H3-10	0.66	0.64	0.68	0.50	0.42	
H3-11	0.86	0.86	0.68	0.61	0.44	
H3-12	0.30	0.13	0.65	0.54	0.29	
H3-13	0.75	0.77	0.61	0.51	0.32	
H3-14	0.93	0.93	0.80	0.59	0.42	
H3-15	0.69	0.12	0.40	0.72	0.40	
H3-16	0.71	0.13	0.32	0.61	0.35	
H3-17	0.51	0.39	0.52	0.73	0.36	
H3-18	0.64	0.26	0.50	0.81	0.53	
H3-19	0.93	0.93	0.35	0.82	0.58	
H3-20	0.86	0.71	0.76	0.77	0.50	
H3-21	0.97	0.94	0.86	0.84	0.68	
H3-22	0.98	0.98	0.55	0.89	0.80	
H3-26	1.00	1.00	0.87	0.85	0.85	

coefficient. The affinity clustering is next, followed by SNN-DBSCAN, and finally k-medoids with constraints, as usual, has the smallest average silhouette coefficient. For the 2017 dataset, DBSCAN and DBSCAN-SUP are again nearly tied for the largest silhouette coefficients of all the clustering methods tested. Next is k-medoids, and last are SNN-DBSCAN and k-medoids with constraints, which both have low average silhouette coefficients. There are two other CDR type-lengths for the 2017 dataset of CDR type H1, H1-14 and H1-15. For both of them, DBSCAN and DBSCAN-SUP are nearly tied for the largest average silhouette coefficient, while SNN-DBSCAN and k-medoids are in the middle and k-medoids with constraints has the lowest average silhouette coefficient. The cluster results and information on the geometric sequences of CDR type H1 are listed in Table 4-8. For each CDR type-length, the most common geometric sequence for the CDR type-length is also the most common CDR type-length for the largest cluster for each of the clustering methods. For H1-13-1, each of the clustering methods has a low geometric sequence entropy and average normalized distance to medoid except for k-medoids, which has a larger geometric sequence entropy than the rest at 2.1, and an average normalized distance to the medoid of 0.32 for the 2017 dataset. The removal of outliers by DBSCAN is apparent in Figure 4-1, which shows the DBSCAN clustering results for residue 4 of H1-13. Compare that to Figure 2-2 to see how much this has improved over the affinity clustering results. For cluster 2, the 2011 dataset does not have enough data to have large enough clusters to get meaningful data. For the 2017 dataset, H1-13-2 has the same geometric sequence as H1-13-1 for SNN-DBSCAN and k-medoids with constraints, which is also the most common geometric sequence for the cluster as a whole (BBBLBBAAABBBB). DBSCAN and

Table 4-8

CDR H1 Clustering Results

Run Info	Cluster	Size	Avg Cl Dist	Primary Conf	Conf Pct.	GSE
2011 Dataset						
	H1-13	303	0.30	BBBLBBAAABBBB	82	1.60
2011AFF	H1-13-1	266	0.14	BBBLBBAAABBBB	94	0.56
2011AFF	H1-13-2	7	0.48	BBAABBAAABBBB	43	2.13
2011AFF	H1-13-3	5	0.64	BBBGAAAAABBBB	40	1.92
2011AFF	H1-13-4	4	0.36	BBBLBBABLBBBB	50	1.50
2011AFF	H1-13-5	4	0.32	BBBGBAAABABBB	50	1.50
DBSCAN	H1-13-1	232	0.11	BBBLBBAAABBBB	99	0.07
DBSCAN	H1-13-2	2	0.06	BBBLBBAABGBBB	100	0.00
DBSCAN	H1-13-3	2	0.04	BBBGBAAABABBB	100	0.00
LMAX	H1-13-1	243	0.11	BBBLBBAAABBBB	99	0.07
LMAX	H1-13-2	2	0.06	BBBLBBAABGBBB	100	0.00
LMAX	H1-13-3	2	0.04	BBBGBAAABABBB	100	0.00
SNN	H1-13-1	91	0.05	BBBLBBAAABBBB	100	0.00
SNN	H1-13-2	19	0.07	BBBLBBAAABBBB	100	0.00
KMC	H1-13-1	255	0.12	BBBLBBAAABBBB	98	0.21
KMC	H1-13-2	12	0.81	BBBGBBAAABBBB	33	2.59
KMC	H1-13-3	6	0.38	BBAABBAAABBBB	50	1.79
KMC	H1-13-4	4	0.48	BBBLBBABLBBBB	50	1.50
KMC	H1-13-5	3	0.18	BBBLBBAABGBBB	100	0.00
KM	H1-13-1	289	0.23	BBBLBBAAABBBB	86	1.22
KM	H1-13-2	14	1.23	BBBGBAAABABBB	14	3.52
2017 Dataset						
	H1-13	3716	0.40	BBBLBBAAABBBB	73	2.47
DBSCAN	H1-13-1	2330	0.09	BBBLBBAAABBBB	99	0.09
DBSCAN	H1-13-2	60	0.05	BBBLBBABLBBBB	100	0.00
DBSCAN	H1-13-3	43	0.03	BBBGBBBAABBBB	100	0.00
LMAX	H1-13-1	2684	0.11	BBBLBBAAABBBB	97	0.23
LMAX	H1-13-2	69	0.07	BBBLBBABLBBBB	100	0.00
LMAX	H1-13-3	50	0.05	BBBGBBBAABBBB	100	0.00
SNN	H1-13-1	780	0.05	BBBLBBAAABBBB	100	0.00
SNN	H1-13-2	115	0.04	BBBLBBAAABBBB	100	0.00
SNN	H1-13-3	84	0.08	BBBLBBAAABBBB	99	0.09
SNN	H1-13-4	51	0.04	BBBLBBAAABBBB	100	0.00
SNN	H1-13-5	24	0.49	BBBLBABBBABBB	63	1.73

Table 4-8

(2017 Dataset continued)

Run Info	Cluster	Size	Avg Cl Dist	Primary Conf	Conf Pct.	GSE
KMC	H1-13-1	1591	0.10	BBBLBBAAABBBB	97	0.23
KMC	H1-13-2	285	0.15	BBBLBBAAABBBB	92	0.62
KMC	H1-13-3	233	0.11	BBBLBBAAABBBB	94	0.39
KMC	H1-13-4	215	0.13	BBBLBBAAABBBB	95	0.43
KMC	H1-13-5	151	0.17	BBBLBBAAABBBB	85	1.07
KM	H1-13-1	3525	0.32	BBBLBBAAABBBB	77	2.06
KM	H1-13-2	191	1.23	BBBAAAABLBBBB	15	4.70
	H1-14	180	0.75	BBBLBBAAABGBBB	23	3.78
DBSCAN	H1-14-1	57	0.05	BBBLBBAAABGBBB	58	1.09
DBSCAN	H1-14-2	12	0.03	BBBLBBAABLBBBB	100	0.00
LMAX	H1-14-1	59	0.06	BBBLBBAAABGBBB	58	1.09
LMAX	H1-14-2	13	0.03	BBBLBBAABLBBBB	100	0.00
LMAX	H1-14-3	9	0.04	BBBLBBAAABABBB	100	0.00
LMAX	H1-14-4	8	0.00	BBBLBABAAABBBB	100	0.00
LMAX	H1-14-5	5	0.02	BBBLBBAABBBBBB	80	0.72
SNN	H1-14-1	151	0.56	BBBLBBAAABGBBB	27	3.43
SNN	H1-14-2	23	0.45	BBBLBABAAABBBB	74	1.16
KMC	H1-14-1	24	0.04	BBBLBBAAABGBBB	83	0.65
KMC	H1-14-2	19	0.07	BBBLBBAAAGGBBB	84	0.63
KMC	H1-14-3	15	0.18	BBBLBBAABBBBBB	93	0.35
KMC	H1-14-4	14	0.05	BBBLBABAAABBBB	100	0.00
KMC	H1-14-5	13	0.07	BBBLBBAAABABBB	85	0.62
KM	H1-14-1	75	0.13	BBBLBBAAABGBBB	55	1.59
KM	H1-14-2	38	0.42	BBBLBBAABBBBBB	37	2.12
KM	H1-14-3	23	0.45	BBBLBABAAABBBB	74	1.16
KM	H1-14-4	20	0.40	BBBLBBAABGBBBB	40	2.42
KM	H1-14-5	18	0.31	BBBLBBAAABABBB	61	1.83
	H1-15	213	0.47	BBBLBBAABBLBBBB	58	2.80
DBSCAN	H1-15-1	96	0.07	BBBLBBAABBLBBBB	100	0.00
DBSCAN	H1-15-2	9	0.00	BBBLBBAABAABBBB	100	0.00
LMAX	H1-15-1	100	0.08	BBBLBBAABBLBBBB	97	0.19
LMAX	H1-15-2	9	0.00	BBBLBBAABAABBBB	100	0.00

Table 4-8 *(2017 Dataset continued)*						
Run Info	Cluster	Size	Avg Cl Dist	Primary Conf	Conf Pct.	GSE
SNN	H1-15-1	116	0.08	BBBLBBAABBLBBBB	97	0.22
SNN	H1-15-2	3	0.37	BBBLGBABBAABBBB	67	0.92
KMC	H1-15-1	38	0.04	BBBLBBAABBLBBBB	97	0.18
KMC	H1-15-2	23	0.04	BBBLBBAABBLBBBB	100	0.00
KMC	H1-15-3	14	0.09	BBBLBBAABAABBBB	93	0.37
KMC	H1-15-4	13	0.23	BBBLBBAABAABBBB	85	0.77
KMC	H1-15-5	12	0.03	BBBLBBAABBLBBBB	100	0.00
KM	H1-15-1	211	0.45	BBBLBBAABBLBBBB	58	2.74
KM	H1-15-2	2	1.02	BABLLBALBABABBB	50	1.00

DBSCAN-SUP for H1-13-2 have the same most common geometric sequence (BBBLBBABLBBBB). DBSCAN, DBSCAN-SUP, and SNN-DBSCAN all have only a single geometric sequence present and so have a geometric sequence entropy of zero. SNN-DBSCAN also removes 69 percent of the total H1-13 dataset as outliers. K-medoids with constraints has a marginally high geometric sequence entropy for H1-13-2 at 0.62, while that of k-medoids without constraints is quite high at 4.70. H1-13-2 for k-medoids without constraints also has a very large average normalized distance to medoid of 1.23. For H1-14, each of the clustering methods gave the geometric sequence BBBLBBAAABGBBB as the most common, which is also the most common geometric sequence for that CDR type-length. All of the clustering methods have a larger geometric sequence entropy and average normalized distance to medoid than is normal, indicating that the cluster is relatively diffuse in comparison to other CDR type-lengths. SNN-DBSCAN is particularly varied with a geometric sequence entropy of 3.43 and an

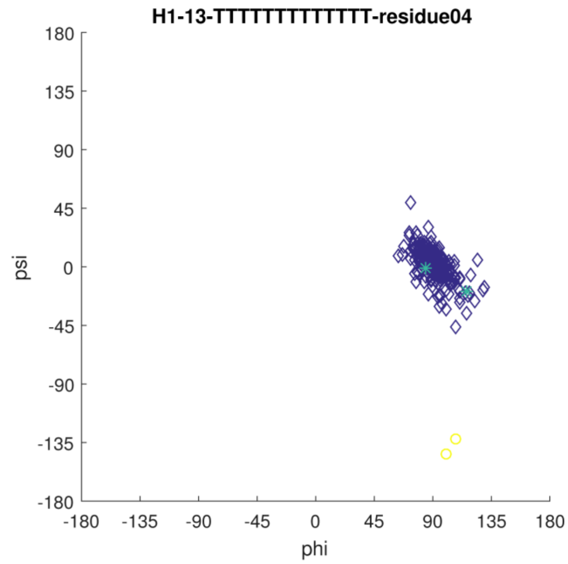

Figure 4-1. The DBSCAN Clusters of CDR H1-13. This is the Ramachandran plot of the DBSCAN clustering for H1-13 from the data of the 2011 North *et al.* study. This is the same CDR type-length and residue position as that of Figure 2-2, which depicts the affinity clustering results for the same data. For this clustering, the first cluster is shown as purple diamonds, the second as green asterisks, and the third as yellow circles. Note that almost all of the diffuse data points from Figure 2-2 have been removed as outliers by DBSCAN.

average normalized distance to medoid of 0.56. DBSCAN and DBSCAN-SUP, conversely, have tighter clusters but remove many more data points as outliers. DBSCAN removed 62 percent of the data for H1-14 as outliers and DBSCAN-SUP removed 48 percent of the H1-14 data as outliers. SNN-DBSCAN removed only 3 percent of the H1-14 data as outliers, so its lackluster performance in clustering H1-14 is understandable. For CDR type-length H1-15, each of the largest clusters for all of the clustering methods has the same most common geometric sequence, BBBLBBABBLBBBB. Each of the clustering methods has a low average distance to medoid and geometric sequence entropy except for k-medoids, which has a geometric sequence entropy of 2.7 and a normalized average distance to medoid of 0.45.

CDR type H2 has one of the more structurally variable of the non-H3 CDR type-lengths, H2-10. The 2011 dataset has two different CDR type-lengths, H2-9 and H2-10, while the 2017 dataset adds a third CDR type-length, H2-12. Clustering results for CDR type H2 are shown in Table 4-9. For CDR type-length H2-9 in the 2011 dataset, the average silhouette coefficient is high for k-medoids clustering (both with and without constraints) and the affinity clustering, while it is low for all of the DBSCAN-using methods, with a value of 0.52 for DBSCAN and 0.44 for DBSCAN-SUP. K-medoids with constraints has an average silhouette coefficient of 0.84, which is much better than its usual scores seen previously. This is an unusual behavior in comparison to most of the rest of the CDR types, where usually DBSCAN and DBSCAN-SUP perform quite well. For the 2017 dataset, k-medoids is the best, while DBSCAN has a better average silhouette coefficient at 0.80 than it does for the 2011 data set. DBSCAN-SUP also performs comparatively poorly for H2-9, with an average silhouette coefficient of 0.52. K-medoids clustering with constraints performs very poorly for the 2017 dataset, with an average silhouette coefficient of 0.15. For H2-10, the results for both the 2011 and 2017 datasets are more typical of the other CDR type-lengths: DBSCAN and DBSCAN-SUP have the largest average silhouette coefficients, with SNN-DBSCAN being next in the ranking, then affinity clustering (for the 2011 dataset), and then the k-medoids clustering (both with and without constraints). For H2-12, k-medoids has the largest average silhouette coefficient, with SNN-DBSCAN being almost as high, with values of 0.86 and

Table 4-9

CDR H2 Clustering Results

Run Info	Cluster	Size	Avg Cl Dist	Primary Conf	Conf Pct.	GSE
2011 Dataset						
	H2-9	81	0.17	BBBAALBBB	91	0.64
2011AFF	H2-9-1	77	0.10	BBBAALBBB	96	0.30
2011AFF	H2-9-2	2	0.15	BBBLLABBB	100	0.00
2011AFF	H2-9-3	2	0.47	BBBBLLBBB	50	1.00
DBSCAN	H2-9-1	42	0.04	BBBAALBBB	100	0.00
DBSCAN	H2-9-2	5	0.02	BBBAALBBB	100	0.00
LMAX	H2-9-1	46	0.05	BBBAALBBB	100	0.00
LMAX	H2-9-2	4	0.02	BBBAALBBB	100	0.00
SNN	H2-9-1	43	0.04	BBBAALBBB	100	0.00
SNN	H2-9-2	6	0.06	BBBAALBBB	100	0.00
KMC	H2-9-1	77	0.10	BBBAALBBB	96	0.30
KMC	H2-9-2	2	0.15	BBBLLABBB	100	0.00
KMC	H2-9-3	1	0.00	BBBBLLBBB	100	0.00
KMC	H2-9-4	1	0.00	BBBABGBBB	100	0.00
KM	H2-9-1	79	0.13	BBBAALBBB	94	0.49
KM	H2-9-2	2	0.15	BBBLLABBB	100	0.00
	H2-10	227	0.49	BBBAAALBBB	67	1.88
2011AFF	H2-10-1	155	0.13	BBBAAALBBB	98	0.17
2011AFF	H2-10-2	42	0.17	BBBAALABBB	90	0.64
2011AFF	H2-10-3	11	0.14	BBBBGALBBB	73	0.85
2011AFF	H2-10-4	7	0.33	BBBBLLABBB	57	1.66
2011AFF	H2-10-5	3	0.14	BBBAALBLBB	100	0.00
DBSCAN	H2-10-1	109	0.08	BBBAAALBBB	100	0.00
DBSCAN	H2-10-2	17	0.05	BBBAALABBB	100	0.00
LMAX	H2-10-1	113	0.08	BBBAAALBBB	100	0.00
LMAX	H2-10-2	24	0.06	BBBAALABBB	100	0.00
SNN	H2-10-1	105	0.08	BBBAAALBBB	100	0.00
SNN	H2-10-2	31	0.08	BBBAALABBB	100	0.00
KMC	H2-10-1	152	0.12	BBBAAALBBB	100	0.00
KMC	H2-10-2	40	0.12	BBBAALABBB	95	0.29
KMC	H2-10-3	8	0.10	BBBBGALBBB	100	0.00
KMC	H2-10-4	5	0.21	BBBBLLABBB	80	0.72
KMC	H2-10-5	4	0.40	BBBABLLBBB	75	0.81
KM	H2-10-1	171	0.21	BBBAAALBBB	89	0.80

Table 4-9 *(continued)*						
Run Info	Cluster	Size	Avg Cl Dist	Primary Conf	Conf Pct.	GSE
2017 Dataset						
KM	H2-10-2	56	0.42	BBBAALABBB	68	1.98
	H2-9	944	0.22	BBBAALBBB	91	0.84
DBSCAN	H2-9-1	801	0.10	BBBAALBBB	100	0.05
DBSCAN	H2-9-2	8	0.04	BBBLLABBB	100	0.00
LMAX	H2-9-1	792	0.10	BBBAALBBB	100	0.01
LMAX	H2-9-2	9	0.04	BBBAALBBB	100	0.00
LMAX	H2-9-3	8	0.04	BBBLLABBB	100	0.00
SNN	H2-9-1	815	0.11	BBBAALBBB	99	0.08
SNN	H2-9-2	15	0.55	BBBBLLBBB	67	1.38
SNN	H2-9-3	6	0.16	BBBABABBB	100	0.00
KMC	H2-9-1	575	0.09	BBBAALBBB	100	0.02
KMC	H2-9-2	89	0.11	BBBAALBBB	99	0.09
KMC	H2-9-3	53	0.19	BBBAALBBB	85	0.74
KMC	H2-9-4	45	0.12	BBBAALBBB	100	0.00
KMC	H2-9-5	44	0.14	BBBAALBBB	100	0.00
KM	H2-9-1	936	0.20	BBBAALBBB	91	0.76
KM	H2-9-2	8	0.97	BBBAALAGL	50	2.00
	H2-10	2982	0.63	BBBAAALBBB	54	2.52
DBSCAN	H2-10-1	1223	0.10	BBBAAALBBB	100	0.00
DBSCAN	H2-10-2	539	0.07	BBBAALABBB	100	0.00
LMAX	H2-10-1	1233	0.10	BBBAAALBBB	100	0.00
LMAX	H2-10-2	477	0.07	BBBAALABBB	100	0.00
SNN	H2-10-1	1015	0.09	BBBAAALBBB	100	0.00
SNN	H2-10-2	116	0.04	BBBAALABBB	100	0.00
SNN	H2-10-3	109	0.24	BBBBGALBBB	59	1.25
SNN	H2-10-4	51	0.05	BBBAALABBB	100	0.00
SNN	H2-10-5	39	0.05	BBBAALABBB	100	0.00
KMC	H2-10-1	1029	0.10	BBBAAALBBB	100	0.02
KMC	H2-10-2	718	0.12	BBBAALABBB	97	0.30
KMC	H2-10-3	299	0.17	BBBAAALBBB	97	0.24
KMC	H2-10-4	142	0.18	BBBAAALBBB	93	0.53
KMC	H2-10-5	130	0.19	BBBAAALBBB	94	0.40
KM	H2-10-1	1977	0.30	BBBAAALBBB	82	1.43
KM	H2-10-2	1005	0.37	BBBAALABBB	73	2.05

Table 4-9						
(2017 Dataset, continued)						
Run Info	Cluster	Size	Avg Cl Dist	Primary Conf	Conf Pct.	GSE
	H2-12	222	0.20	BBBBAAALLBBB	91	0.75
DBSCAN	H2-12-1	119	0.05	BBBBAAALLBBB	100	0.00
DBSCAN	H2-12-2	10	0.02	BBBBAAALLBBB	100	0.00
LMAX	H2-12-1	144	0.06	BBBBAAALLBBB	100	0.00
LMAX	H2-12-2	11	0.02	BBBBAAALLBBB	100	0.00
SNN	H2-12-1	189	0.08	BBBBAAALLBBB	99	0.05
SNN	H2-12-2	2	0.01	BBBBBAAGBBBB	50	1.00
KMC	H2-12-1	65	0.03	BBBBAAALLBBB	100	0.00
KMC	H2-12-2	53	0.05	BBBBAAALLBBB	100	0.00
KMC	H2-12-3	30	0.04	BBBBAAALLBBB	100	0.00
KMC	H2-12-4	18	0.07	BBBBAAALLBBB	100	0.00
KMC	H2-12-5	12	0.04	BBBBAAALLBBB	100	0.00
KM	H2-12-1	212	0.12	BBBBAAALLBBB	96	0.39
KM	H2-12-2	10	1.44	GBBAALBABBBB	30	2.45

0.85. DBSCAN and DBSCAN-SUP do not perform as well, with values of 0.46 and 0.44. Finally, k-medoids with constraints is the worst performer with an average silhouette coefficient of 0.39. For cluster H2-9-1 for both the 2011 and 2017 datasets, the largest cluster for each of the clustering methods for H2-9, each has the same most common geometric sequence, which is BBBAALBBB. This is also the most common geometric sequence for CDR type-length H2-9 as a whole. Although k-medoids has a high average silhouette coefficient for H2-9, it also has the highest geometric sequence entropy for the 2017 dataset (0.76) and the largest normalized average distance to medoid (0.2). Interestingly, for DBSCAN and DBSCAN-SUP, the vast majority of the data points are in H2-9-1, the first cluster. As for why they have such poor average silhouette coefficient values, refer to Figure 4-2, which shows a Ramachandran plot for H2-9 at

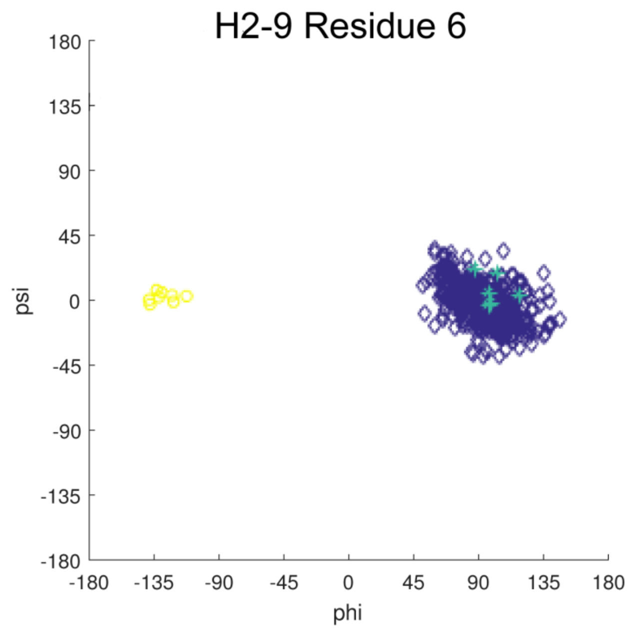

Figure 4-2. The DBSCAN-SUP Clusters of CDR H2-9. This is a Ramachandran plot of CDR type-length H2-9 residue 6 after being clustered by DBSCAN using the supremum distance as described in the Methods section. The largest cluster, cluster 1, is depicted with purple diamonds. The second largest cluster, cluster 2, is shown as green asterisks overlapping cluster 1. Cluster 3 is shown as yellow circles. Cluster 2 overlaps cluster 1 for the DBSCAN-SUP clustering of H2-9 for each residue, lowering its average silhouette coefficient.

residue position 6 for the 2017 dataset as clustered by DBSCAN-SUP. There are three clusters shown there, with the largest by far being H2-9-1, which is depicted by the purple diamonds. Next in size is H2-9-2, in green asterisks, while the smallest cluster is H2-9-3, shown as yellow circles. Notice that the members of H2-9-2 overlap perfectly with those of H2-9-1. This is also true for the other residue positions of H2-9, so the two clusters are indistinguishable. This is why DBSCAN-SUP has such poor average silhouette values for H2-9. For H2-10, all of the largest clusters for each of the clustering runs have the same most common geometric sequence, BBBAAALBBB. For H2-10-1, only k-medoids has a large geometric sequence entropy, with a value of 1.4. However, it still has a moderate value of the normalized average distance to mediod at 0.3. For H2-10-2, the second largest cluster, each clustering method returned a most common geometric sequence of BBBAALABBB, showing the consistency in results between the different clustering methods. CDR type-length H2-12 is interesting in that SNN-DBSCAN outperforms DBSCAN and DBSCAN-SUP, which is unusual for these datasets. Again, the most common geometric sequence for the CDR type-length, BBBBAAALLBBB, is also the most common geometric sequence for the largest cluster for each clustering method. Given the past behavior of the clustering methods, this result is expected. However, notice that the second largest cluster for DBSCAN and DBSCAN-SUP, H2-12-2, has the same geometric sequence as H2-10-1, while the second largest cluster for H2-12 for SNN-DBSCAN has a geometric sequence of BBBBBAAGBBBB, which is different. So, the two largest clusters for H2-12 for DBSCAN and DBSCAN-SUP are too close to one another and so reduce the value of the average silhouette coefficient.

CDR type H3 is the most diverse of the CDR types in terms of sequence, structure, and even the distribution of loop lengths. This variability renders their clustering to be difficult, if not impossible. The 2011 dataset does not have any H3 CDR type-lengths with 50 or more structures, so its usefulness is limited in this case. For the 2017 dataset, there are seemingly many more CDR type-lengths to examine, but many are duplicates of each other and so are not useful. The results of clustering the H3 CDR loops of the 2017 dataset are shown in Table 4-10. To start with, there are two different

Table 4-10

CDR H3 Clustering Results

Run Info	Cluster	Size	Avg Cl Dist	Primary Conf	Conf Pct.	GSE
	H3-5	59	1.37	BAGAB	31	2.60
DBSCAN	H3-5-1	19	0.02	BAGAB	95	0.30
DBSCAN	H3-5-2	10	0.03	BBLLB	100	0.00
LMAX	H3-5-1	19	0.02	BAGAB	95	0.30
LMAX	H3-5-2	10	0.03	BBLLB	100	0.00
SNN	H3-5-1	21	0.44	BBLLB	57	0.99
SNN	H3-5-2	12	0.01	BAGAB	100	0.00
KMC	H3-5-1	18	0.02	BAGAB	100	0.00
KMC	H3-5-2	12	0.07	BBLLB	100	0.00
KMC	H3-5-3	10	0.07	BBBLB	100	0.00
KMC	H3-5-4	8	0.05	BALAB	100	0.00
KMC	H3-5-5	5	0.05	BBBBB	100	0.00
KM	H3-5-1	32	0.34	BAGAB	56	1.69
KM	H3-5-2	12	0.07	BBLLB	100	0.00
KM	H3-5-3	10	0.07	BBBLB	100	0.00
KM	H3-5-4	5	0.05	BBBBB	100	0.00
	H3-6	82	1.15	BBGAAB	44	2.38
DBSCAN	H3-6-1	21	0.08	BBGAAB	100	0.00
DBSCAN	H3-6-2	14	0.07	BBLLAB	100	0.00
LMAX	H3-6-1	21	0.08	BBGAAB	100	0.00
LMAX	H3-6-2	9	0.03	BBLLAB	100	0.00
SNN	H3-6-1	22	0.15	BBLLAB	95	0.27
SNN	H3-6-2	22	0.13	BBGAAB	100	0.00
KMC	H3-6-1	36	0.20	BBGAAB	100	0.00
KMC	H3-6-2	23	0.17	BBLLAB	96	0.26
KMC	H3-6-3	8	0.18	BBAALB	100	0.00
KMC	H3-6-4	4	0.04	BBBLBB	100	0.00
KMC	H3-6-5	3	0.01	BAALAB	100	0.00
KM	H3-6-1	37	0.24	BBGAAB	97	0.18
KM	H3-6-2	28	0.40	BBLLAB	79	1.02
KM	H3-6-3	11	0.36	BBAALB	73	1.10
KM	H3-6-4	6	0.54	BBBLBB	67	1.25
	H3-7	108	0.80	BBAALAB	56	2.55
DBSCAN	H3-7-1	35	0.02	BBAALAB	100	0.00
DBSCAN	H3-7-2	4	0.01	BBBLLAB	100	0.00

Table 4-10

(continued)

Run Info	Cluster	Size	Avg Cl Dist	Primary Conf	Conf Pct.	GSE
DBSCAN	H3-7-3	4	0.00	BBLGAAB	100	0.00
LMAX	H3-7-1	39	0.02	BBAALAB	100	0.00
LMAX	H3-7-2	5	0.01	BBBLLAB	100	0.00
LMAX	H3-7-3	4	0.00	BBLGAAB	100	0.00
SNN	H3-7-1	22	0.56	BBLLLAB	50	1.43
SNN	H3-7-2	21	0.01	BBAALAB	100	0.00
KMC	H3-7-1	62	0.10	BBAALAB	97	0.24
KMC	H3-7-2	12	0.17	BBLLLAB	100	0.00
KMC	H3-7-3	10	0.11	BBBLLAB	100	0.00
KMC	H3-7-4	4	0.00	BBLGAAB	100	0.00
KMC	H3-7-5	3	0.08	BBAAALB	100	0.00
KM	H3-7-1	70	0.26	BBAALAB	86	0.90
KM	H3-7-2	11	0.11	BBLLLAB	100	0.00
KM	H3-7-3	10	0.11	BBBLLAB	100	0.00
KM	H3-7-4	6	0.42	BBLGAAB	67	1.25
KM	H3-7-5	5	0.81	BABABBB	40	1.52
	H3-8	54	1.86	BBGAALAB	30	3.31
DBSCAN	H3-8-1	8	0.01	BBABGBBB	100	0.00
DBSCAN	H3-8-2	7	0.03	BBGAALAB	100	0.00
DBSCAN	H3-8-3	6	0.02	BBBBGBBB	100	0.00
LMAX	H3-8-1	8	0.01	BBABGBBB	100	0.00
LMAX	H3-8-2	6	0.02	BBBBGBBB	100	0.00
LMAX	H3-8-3	6	0.03	BBGAALAB	100	0.00
SNN	H3-8-1	20	0.24	BBGAALAB	75	0.99
SNN	H3-8-2	11	0.22	BBABGBBB	82	0.87
KMC	H3-8-1	16	0.06	BBGAALAB	100	0.00
KMC	H3-8-2	9	0.02	BBABGBBB	100	0.00
KMC	H3-8-3	4	0.01	BBBGGBBB	100	0.00
KMC	H3-8-4	4	0.00	BBBBGBBB	100	0.00
KMC	H3-8-5	4	0.02	BBGABBAB	100	0.00
KM	H3-8-1	28	0.61	BBGAALAB	57	2.09
KM	H3-8-2	10	0.30	BBBBGBBB	60	1.37
KM	H3-8-3	9	0.02	BBABGBBB	100	0.00
KM	H3-8-4	4	0.01	BBBGGBBB	100	0.00
KM	H3-8-5	2	0.32	BBABABBB	50	1.00

Table 4-10

(continued)

Run Info	Cluster	Size	Avg Cl Dist	Primary Conf	Conf Pct.	GSE
	H3-12	558	1.38	BBBBAAALBBAB	11	6.04
DBSCAN	H3-12-1	291	0.94	BBBBAAALBBAB	21	4.37
DBSCAN	H3-12-2	32	0.19	BBBAGAALGBAB	78	0.76
LMAX	H3-12-1	440	1.21	BBBBAAALBBAB	14	5.26
LMAX	H3-12-2	11	0.03	BBBBAAALBLBB	100	0.00
LMAX	H3-12-3	11	1.02	BBBABAABABBB	27	2.66
LMAX	H3-12-4	10	0.47	BBBGBLBABABB	40	2.05
LMAX	H3-12-5	8	0.22	BBBBAABAABBB	100	0.00
SNN	H3-12-1	88	0.43	BBBBAAALBBAB	66	1.88
SNN	H3-12-2	18	0.01	BBBAABAABBAB	100	0.00
KMC	H3-12-1	40	0.04	BBBAABAABBAB	100	0.00
KMC	H3-12-2	28	0.44	BBBBBLBABBAB	75	1.03
KMC	H3-12-3	27	0.22	BBBBAABABBAB	56	1.62
KMC	H3-12-4	25	0.01	BBBAGAALGBAB	100	0.00
KMC	H3-12-5	23	0.24	BBBBAAALBBAB	91	0.43
KM	H3-12-1	113	0.53	BBBBAAALBBAB	54	2.50
KM	H3-12-2	60	0.70	BBBBAABABBAB	25	3.54
KM	H3-12-3	45	0.76	BBBBBLBABBAB	47	2.92
KM	H3-12-4	44	0.15	BBBAABAABBAB	91	0.58
KM	H3-12-5	43	0.89	BBBABAABBBAB	21	3.87
	H3-14	340	1.35	BBBBBAALBBBBAB	21	5.45
DBSCAN	H3-14-1	72	0.08	BBBBBAALBBBBAB	99	0.11
DBSCAN	H3-14-2	20	0.04	BBLBBGABAAABAB	100	0.00
LMAX	H3-14-1	72	0.08	BBBBBAALBBBBAB	99	0.11
LMAX	H3-14-2	18	0.03	BBLBBGABAAABAB	100	0.00

Table 4-10

(continued)

Run Info	Cluster	Size	Avg Cl Dist	Primary Conf	Conf Pct.	GSE
SNN	H3-14-1	83	0.21	BBBBBAALBBBBAB	86	0.92
SNN	H3-14-2	4	0.03	BBBABBBAABBBAB	100	0.00
SNN	H3-14-3	4	0.00	BBLBBGABAAAABAB	100	0.00
KMC	H3-14-1	74	0.11	BBBBBAALBBBBAB	96	0.28
KMC	H3-14-2	30	0.29	BBBABABAABBBAB	40	2.09
KMC	H3-14-3	20	0.04	BBLBBGABAAAABAB	100	0.00
KMC	H3-14-4	19	0.38	BBBABAABAABBAB	68	1.36
KMC	H3-14-5	16	0.81	BBBBAABABBBBAB	38	2.36
KM	H3-14-1	79	0.20	BBBBBAALBBBBAB	90	0.73
KM	H3-14-2	51	0.73	BBBBBBLLBBBBAB	35	3.26
KM	H3-14-3	46	0.58	BBBABABAABBBAB	26	3.31
KM	H3-14-4	32	1.14	BBBLABAABABBAB	13	4.08
KM	H3-14-5	29	0.62	BBBABAABAABBAB	45	2.62

regimes for clustering the H3 loops. The shorter H3 loops, from five to eight residues long, are easier to cluster. H3 loops that are nine residues in length or longer become much more difficult to cluster. This is apparent from the values of the average silhouette coefficients for the different cluster methods for the H3 CDRs. For H3-5 through H3-8, the average silhouette coefficient values are highest for DBSCAN and DBSCAN-SUP, which is also how they have performed for many other CDR type-lengths. However, notice in the outlier percentages table in Appendix B that they also remove a great deal of the H3 datasets as outliers, with the smallest such percentage from H3-5 to H3-8 being 43 percent. With much of the variability removed, the data become much easier to cluster, but the dataset after the outlier removal has had much of the CDR structural diversity present in the original dataset removed. For instance, for H3-5, the most common geometric sequence is BAGAB, but that only represents 31 percent of the dataset. For DBSCAN, H3-5-1, the largest H3-5 cluster, has 95 percent of its members with a geometric sequence of BAGAB. The second most common cluster for DBSCAN, H3-5-2, has 10 members, all of which have a geometric sequence of BBLLB. There are no other clusters in H3-5 for DBSCAN. To see the loss in structural diversity, compare the geometric sequence entropies of the whole CDR type-length (2.6) to that of the DBSCAN clusters (0.30 and zero). Alternatively, k-medoids and k-medoids with constraints do well with H3-5 because they have a large value of the number of clusters *k*, with most clusters having a small membership. This pattern continues until H3-9, which has many more members than the others (with a size of 179 CDR loops) and a large geometric sequence entropy of 5.1. Finally, the average silhouette coefficient values for the larger H3 CDR type-lengths seem to be very inconsistent, with some being small and others being large. For instance, for DBSCAN, the average silhouette coefficient for H3-12 is 0.30, a low value, while that of H3-14 is quite high at 0.93. The answer is in the number of duplicate structures in the database for some CDR type-lengths. This problem is worse for CDR H3 because of its large distribution of structures across many different loop lengths. To demonstrate the problem, notice in Appendix A that the number of unique sequences for H3-12-1 for DBSCAN is 89, which is a fairly large number. DBSCAN performed relatively poorly for H3-12. However, for H3-14, where DBSCAN has a very high average silhouette coefficient, there are 72 members of H3-14-1 for DBSCAN but only one unique sequence among them. They each have the same sequence and so have very similar structures. Due to the overall structural diversity, resolving a useful, good quality clustering out of the CDR H3 data is likely to be impossible. For the larger loop lengths, there aren't enough structures yet to be able to see the real diversity in structures, and that diversity is likely to be so great that the data cannot be grouped into separate, observable structures.

CHAPTER 5

DISCUSSION

Although each of the clustering methods tested on these antibody CDR datasets are successful for some of the CDR type-lengths, there are some clustering methods that are better than others. Overall, the DBSCAN and DBSCAN-SUP clustering methods are the best of the ones tested. Both tend to have high average silhouette coefficient values and low geometric sequence entropies for their clusters. Typically, DBSCAN slightly outperforms DBSCAN-SUP, but in the case of H2-9 for the 2017 dataset DBSCAN is much better. This is due to the supremum distance splitting the largest cluster into two clusters, since the overall closeness of the two putative clusters is not measured by the supremum distance measurement. The affinity clustering of the 2011 dataset is consistently in the middle of the rankings by average silhouette coefficient values. It clusters the data well, but its inability to deal with outliers would necessitate either pre-processing or post-processing the data to remove outliers. K-medoids tends to make large clusters that have multiple geometric sequences present in them, which is apparent from the high geometric sequence entropies of the K-medoids clusters. K-medoids with constraints has very interesting results: the average silhouette coefficient values of its clustering results tend to be very low, but the geometric sequence entropies also tend to be low. In effect, the k-medoids with constraints as proposed here acts similarly to a simple clustering method that puts the CDR loops of each unique geometric sequence into its own cluster. The very small values of the average silhouette coefficient for k-medoids with constraints demonstrates that clustering only by hashing geometric sequence values does not produce a good result.

In comparison to the other clustering methods tested, SNN-DBSCAN has a widely variable degree of success. For some CDR type-lengths, it is the best clustering method or near the best, as measured by the average silhouette coefficient. For many other CDR type-lengths, it performs very poorly. It usually does not perform as well as regular DBSCAN. It produces typically more clusters than DBSCAN or DBSCAN-SUP. It also has the problem of requiring a large value of *k*, the size of the nearest neighbor list, to get a good clustering result for the larger CDR type-lengths. A length of 100 is needed for the more common CDR type-lengths from the 2017 dataset for the clustering to function at all. Without a large value of *k*, the residual SNN-DBSCAN graph is too sparse for a good clustering, even if the value of the Eps parameter is intentionally kept small. For instance, for the CDR type-length L1-11 of the 2017 dataset and a neighbor list length of 20, out of all of the potential links of pairs of data points that are on each other's nearest neighbor lists, only seven percent of them have one or more mutual nearest neighbors other than each other.

Another open question is how to best deal with outlier conformations. All of the DBSCAN variants designate outliers as part of their clustering function. This is very helpful, and DBSCAN and DBSCAN-SUP are both very successful with that for the CDR type-lengths that are more compactly naturally clustered. For instance, for CDR type-length L1-11, DBSCAN and DBSCAN-SUP remove the affinity cluster L1-11-3, which is too diverse in conformation to qualify as a proper cluster. However, when the

CDR type-length in question is not as naturally clustered, DBSCAN and DBSCAN-SUP can be misleading. Instead of showing results for which the clustering fares poorly, DBSCAN and DBSCAN-SUP remove many of the conformations present in the original set, which pares the data down to a much smaller set that are readily clustered. This behavior is clear from the clustering results of CDR type-length H3-5. While the most common geometric sequence of H3-5 is only 30 percent or so of the data, it is all or most of the first cluster of both DBSCAN and DBSCAN-SUP. The second largest cluster for DBSCAN and DBSCAN-SUP is less than half the size of the first cluster, and also has one geometric sequence predominate. The rest of the geometric sequences present in the CDR loops of the CDR H3-5 dataset are just not present in the DBSCAN results. Another alternative outlier removal is suggested by the low geometric sequence entropy results of the k-medoids clustering with constraints. The different CDR loops could be hashed by their geometric sequences using a reduced alphabet, like the XLG alphabet used to form the cannot-link constraints of the k-medoids constrained clustering, and CDR loops that have geometric sequences that are rare in the database can be removed prior to clustering. This method has the advantage of using the biophysical knowledge of how certain regions of the Ramachandran plot have populated regions between them, but other boundaries between the density peaks in the plot do not have populated boundary regions. This allows for a more stringent cutoff without having to fear that too many of the data points are being removed as outliers, as would happen with just a simple dihedral distance cutoff like that used in North *et al.* 2011.

The size of the antibody structure database has grown significantly since our last survey in 2011. Rather than approximately 800 CDR structures per CDR type, there are more than 4,000 of them in the January 2017 database. Another way of conceptualizing the growth of the antibody structure database is by the number of CDR type-lengths with a significant number of unique amino acid sequences present. For the 2011 survey, there are only 16 CDR type lengths that have 20 or more unique sequences. For the 2017 survey, there are 36 CDR type lengths that have 20 or more unique sequences. The twenty CDR type-lengths now have enough diversity that they are good candidates for characterization to potentially learn more about antibody CDR structures.

BIBLIOGRAPHY

Adolf-Bryfogle, J., Xu, Q., North, B., Lehmann, A., & Dunbrack, R. L. (2015). Pyigclassify: A database of antibody cdr structural classifications. *Nucleic Acids Res, 43*(Database issue), D432-438. doi: 10.1093/nar/gku1106

Al-Lazikani, B., Lesk, A. M., & Chothia, C. (1997). Standard conformations for the canonical structures of immunoglobulins. *J Mol Biol, 273*(4), 927-948. doi: 10.1006/jmbi.1997.1354

Basu, S., Banerjee, A., & Mooney, R. J. Active semi-supervision for pairwise constrained clustering *Proceedings of the 2004 siam international conference on data mining* (pp. 333-344).

Basu, S., Bilenko, M., & Mooney, R. J. (2004). *A probabilistic framework for semi-supervised clustering*. Paper presented at the Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining, Seattle, WA, USA.

Bork, P., Holm, L., & Sander, C. (1994). The immunoglobulin fold. Structural classification, sequence patterns and common core. *J Mol Biol, 242*(4), 309-320. doi: 10.1006/jmbi.1994.1582

Chothia, C., & Lesk, A. M. (1987). Canonical structures for the hypervariable regions of immunoglobulins. *J Mol Biol, 196*(4), 901-917.

Creighton, T. E. (1993). *Proteins : Structures and molecular properties* (2nd ed.). New York: W.H. Freeman.

Davidson, I., & Basu, S. (2007). A survey of clustering with instance level constraints. *ACM Transactions on Knowledge Discovery from Data*. Retrieved from

Ertöz, L., Steinbach, M., & Kumar, V. Finding clusters of different sizes, shapes, and densities in noisy, high dimensional data *Proceedings of the 2003 siam international conference on data mining* (pp. 47-58).

Ertöz, L., Steinbach, M., & Kumar, V. (2002). *A new shared nearest neighbor clustering algorithm and its applications*. Paper presented at the Workshop on Clustering High Dimensional Data and its Applications at 2nd SIAM International Conference on Data Mining. www-users.cs.umn.edu/~kumar/papers/siam_hd_snn_cluster.pdf

Ester, M., Kriegel, H.-P., Sander, J., & Xu, X. (1996). *A density-based algorithm for discovering clusters in large spatial databases with noise.* Paper presented at the Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining, Portland, OR.

Frey, B. J., & Dueck, D. (2007). Clustering by passing messages between data points. *Science, 315*(5814), 972-976. doi: 10.1126/science.1136800

Jarvis, R. A., & Patrick, E. A. (1973). Clustering using a similarity measure based on shared near neighbors. *IEEE Transactions on Computers, C-22*(11), 1025-1034. doi: 10.1109/T-C.1973.223640

Kaufman, L., & Rousseeuw, P. (1987). Clustering by means of medoids. *Statistical Data Analysis Based on the L1-Norm and Related Methods*, North-Holland.

Kuroda, D., Shirai, H., Kobori, M., & Nakamura, H. (2008). Structural classification of cdr-h3 revisited: A lesson in antibody modeling. *Proteins, 73*(3), 608-620. doi: 10.1002/prot.22087

Lodish, H., Berk, A., Kaiser, C., Krieger, M., Scott, M., Bretscher, A., et al. (2007). *Molecular cell biology* (Sixth Edition ed.).

MacQueen, J. (1967, 1967). *Some methods for classification and analysis of multivariate observations.* Paper presented at the Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Statistics, Berkeley, Calif.

Martin, A. C., & Thornton, J. M. (1996). Structural families in loops of homologous proteins: Automatic classification, modelling and application to antibodies. *J Mol Biol, 263*(5), 800-815. doi: 10.1006/jmbi.1996.0617

Morea, V., Tramontano, A., Rustici, M., Chothia, C., & Lesk, A. M. (1998). Conformations of the third hypervariable region in the vh domain of immunoglobulins. *J Mol Biol, 275*(2), 269-294. doi: 10.1006/jmbi.1997.1442

North, B., Lehmann, A., & Dunbrack, R. L. (2011). A new clustering of antibody cdr loop conformations. *J Mol Biol, 406*(2), 228-256. doi: 10.1016/j.jmb.2010.10.030

Oliva, B., Bates, P. A., Querol, E., Avilés, F. X., & Sternberg, M. J. (1998). Automated classification of antibody complementarity determining region 3 of the heavy chain (h3) loops into canonical forms and its application to protein structure prediction. *J Mol Biol, 279*(5), 1193-1210. doi: 10.1006/jmbi.1998.1847

Ramachandran, G. N., & Sasisekharan, V. (1968). Conformation of polypeptides and proteins. *Adv Protein Chem, 23,* 283-438.

Rousseeuw, P. J. (1987). Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics, 20*, 53-65. doi: http://dx.doi.org/10.1016/0377-0427(87)90125-7

Schwenker, F., & Trentin, E. (2014). Pattern classification and clustering: A review of partially supervised learning approaches. *Pattern Recognition Letters, 37*, 4-14. doi: http://dx.doi.org/10.1016/j.patrec.2013.10.017

Shannon, C. E., & Weaver, W. (1998). *The mathematical theory of communication*. Urbana: University of Illinois Press.

Shirai, H., Kidera, A., & Nakamura, H. (1999). H3-rules: Identification of cdr-h3 structures in antibodies. *FEBS Lett, 455*(1-2), 188-197.

Tan, P.-N., Steinbach, M., & Kumar, V. (2006). *Introduction to data mining* (1st ed.). Boston: Pearson Addison Wesley.

Ting, D., Wang, G., Shapovalov, M., Mitra, R., Jordan, M. I., & Dunbrack, R. L. (2010). Neighbor-dependent ramachandran probability distributions of amino acids developed from a hierarchical dirichlet process model. *PLoS Comput Biol, 6*(4), e1000763. doi: 10.1371/journal.pcbi.1000763

Wagstaff, K., & Cardie, C. (2000). *Clustering with instance-level constraints*. Paper presented at the Proceedings of the Seventeenth International Conference on Machine Learning.

Wagstaff, K., Cardie, C., Rogers, S., Schr, S., #246, & dl. (2001). *Constrained k-means clustering with background knowledge*. Paper presented at the Proceedings of the Eighteenth International Conference on Machine Learning.

Ward, J. H. (1963). Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association, 58*(301), 236-244. doi: 10.1080/01621459.1963.10500845

Weitzner, Brian D., Dunbrack, Roland L., Jr., & Gray, Jeffrey J. (2015). The origin of cdr h3 structural diversity. *Structure, 23*(2), 302-311. doi: 10.1016/j.str.2014.11.010

APPENDIX A

CLUSTER SEQUENCE DATA

Run Info	Cluster	Size	Consensus Seq	Unique Seqs	No. UGS
2011 Dataset					
	L1-11	134	rASqdisnyla	95	9
2011AFF	L1-11-1	74	rASqsisnyla	56	3
2011AFF	L1-11-2	55	rASqdIsnyla	37	2
2011AFF	L1-11-3	5	ggnnigs-svh	5	4
DBSCAN	L1-11-1	45	rASqsisnnla	33	2
DBSCAN	L1-11-2	31	rASqdIhnyLa	18	1
LMAX	L1-11-1	41	rASqsisnnla	30	1
LMAX	L1-11-2	33	rASqdIhnyla	20	1
SNN	L1-11-1	52	rASqsisnnla	38	2
SNN	L1-11-2	40	rASqdIhnyla	25	1
KMC	L1-11-1	74	rASqsisnyla	56	3
KMC	L1-11-2	52	rASqdIsnyla	35	1
KMC	L1-11-3	2	kASQDiSnaln	2	1
KMC	L1-11-4	2	rAnaapnqsaa	2	2
KMC	L1-11-5	2	GGNNIGSkSVH	2	1
KM	L1-11-1	74	rASqsisnyla	56	3
KM	L1-11-2	55	rASqdIsnyla	37	2
KM	L1-11-3	5	ggnnigs-svh	5	4
	L1-16	64	rSSqslvhsnGnTYLe	48	11
2011AFF	L1-16-1	64	rSSqslvhsnGnTYLe	48	11
DBSCAN	L1-16-1	29	rSSqslvhSnGnTYLh	22	1
DBSCAN	L1-16-2	3	rSSQtIVHnNGNTYLE	3	1
LMAX	L1-16-1	31	rSSqslvhSnGnTYLh	22	1
LMAX	L1-16-2	4	rSSQtIVHsNGnTYLE	4	2
SNN	L1-16-1	36	rSSqSlvhSnGnTYLh	27	1
SNN	L1-16-2	4	rsSQaiVhanGNTYLe	4	2
KMC	L1-16-1	14	RSSQslvHSnGnTYLh	10	1
KMC	L1-16-2	8	rsSqSLlysnGkTYLn	7	1
KMC	L1-16-3	5	rSSqSLvhSnGnTYLh	3	1
KMC	L1-16-4	4	rSSQtIVHtNGNTYlE	4	1
KMC	L1-16-5	3	rsSQsllksNGkTfLn	3	1
KM	L1-16-1	63	rSSqslvhsnGnTYLe	47	10
KM	L1-16-2	1	RSSQIIVHNNGNTYLE	1	1

Run Info	Cluster	Size	Consensus Seq	Unique Seqs	No. UGS
	L2-8	295	y-asnlas	161	11
2011AFF	L2-8-1	279	y-asnlas	155	4
2011AFF	L2-8-2	9	yaasnlas	8	3
2011AFF	L2-8-3	3	segNtlrP	2	2
2011AFF	L2-8-4	2	gGanNRvp	2	1
2011AFF	L2-8-5	2	YkaSnRfS	2	2
DBSCAN	L2-8-1	186	Y-asnlas	108	2
DBSCAN	L2-8-2	10	kyaSqSiS	6	2
LMAX	L2-8-1	193	Ykasnlas	108	2
LMAX	L2-8-2	10	kyaSqSiS	6	2
SNN	L2-8-1	141	Y-asnlas	85	2
SNN	L2-8-2	4	yaaSnlaS	4	3
KMC	L2-8-1	155	Y-asnlas	98	1
KMC	L2-8-2	73	yktsnlfs	45	2
KMC	L2-8-3	12	kyASqsiS	8	1
KMC	L2-8-4	8	ydvsnRpS	8	1
KMC	L2-8-5	7	yaasnlds	6	2
KM	L2-8-1	291	y-asnlas	159	8
KM	L2-8-2	4	ggannRyp	4	3
	L3-9-cis7	219	qqgss-PyT	187	8
2011AFF	L3-9-cis7-1	209	qQgss-PyT	179	4
2011AFF	L3-9-cis7-2	8	QhfwstPrt	7	2
2011AFF	L3-9-cis7-3	2	qQsthvPwT	2	2
DBSCAN	L3-9-cis7-1	176	qQgss-PyT	149	2
DBSCAN	L3-9-cis7-2	6	QHFwSTPRt	5	1
LMAX	L3-9-cis7-1	168	qQgss-PyT	143	2
LMAX	L3-9-cis7-2	6	QHFwSTPRt	5	1
SNN	L3-9-cis7-1	118	qQsns-PyT	101	1
SNN	L3-9-cis7-2	6	Qqs-esPyT	6	1
SNN	L3-9-cis7-3	5	QhsystPwT	5	1
SNN	L3-9-cis7-4	3	QQhdelPfT	3	1
KMC	L3-9-cis7-1	209	qQgss-PyT	179	4
KMC	L3-9-cis7-2	8	QhfwstPrt	7	2
KMC	L3-9-cis7-3	1	QQYYIYPYT	1	1
KMC	L3-9-cis7-4	1	SQSTHVPWT	1	1
KM	L3-9-cis7-1	217	qqgss-PyT	186	6
KM	L3-9-cis7-2	2	qQsthvPwT	2	2

Run Info	Cluster	Size	Consensus Seq	Unique Seqs	No. UGS
	H1-13	303	kasGftftsyymh	246	41
2011AFF	H1-13-1	266	kasGftftsyymh	213	13
2011AFF	H1-13-2	7	kaSgfnitdyain	7	5
2011AFF	H1-13-3	5	kASGytFssyain	5	4
2011AFF	H1-13-4	4	aasGfsfsgyyws	4	3
2011AFF	H1-13-5	4	aASGfiasinymg	4	3
DBSCAN	H1-13-1	232	kasGftftsyymh	186	2
DBSCAN	H1-13-2	2	SVTGdpITSGfWd	2	1
DBSCAN	H1-13-3	2	AASGfiaSikYlG	2	1
LMAX	H1-13-1	243	kasGftftsyymh	196	2
LMAX	H1-13-2	2	SVTGdpITSGfWd	2	1
LMAX	H1-13-3	2	AASGfiaSikYlG	2	1
SNN	H1-13-1	91	kaSGytFtsywmh	80	1
SNN	H1-13-2	19	tvSGFsltdygvh	15	1
KMC	H1-13-1	255	kasGftftsyymh	206	5
KMC	H1-13-2	12	kASGytfssyamg	12	7
KMC	H1-13-3	6	kaSgfnitdyyin	6	4
KMC	H1-13-4	4	aasGftisgyais	4	3
KMC	H1-13-5	3	svtGdpiTSgfwd	3	1
KM	H1-13-1	289	kasGftftsyymh	232	29
KM	H1-13-2	14	aaSgytfs-yymg	14	12
	H2-9	81	yisysgsty	61	7
2011AFF	H2-9-1	77	yisysgsty	57	4
2011AFF	H2-9-2	2	sIrngfiih	2	1
2011AFF	H2-9-3	2	sIrsgGgiY	2	2
DBSCAN	H2-9-1	42	yisysGsTy	27	1
DBSCAN	H2-9-2	5	hIwwDDDkr	4	1
LMAX	H2-9-1	46	yisysGsTy	31	1
LMAX	H2-9-2	4	hIwwDDDkr	4	1
SNN	H2-9-1	43	yisysGsTy	29	1
SNN	H2-9-2	6	dIwwdddkr	5	1
KMC	H2-9-1	77	yisysgsty	57	4
KMC	H2-9-2	2	sIrngfiih	2	1
KMC	H2-9-3	1	YIRYGGGTY	1	1
KMC	H2-9-4	1	NVYDSGDTN	1	1
KM	H2-9-1	79	yisysgsty	59	6
KM	H2-9-2	2	sIrngfiih	2	1
	H2-10	227	-ispgng-ty	196	22

Run Info	Cluster	Size	Consensus Seq	Unique Seqs	No. UGS
2011AFF	H2-10-1	155	wiypgng-t-	131	4
2011AFF	H2-10-2	42	-Issgggyty	40	5
2011AFF	H2-10-3	11	eIlPGsgstn	9	2
2011AFF	H2-10-4	7	tIssgggytd	7	4
2011AFF	H2-10-5	3	AmdsGGggtl	2	1
DBSCAN	H2-10-1	109	-idpgng-tn	97	1
DBSCAN	H2-10-2	17	aisggggyty	17	1
LMAX	H2-10-1	113	-idpgng-tn	100	1
LMAX	H2-10-2	24	tIssgggyty	24	1
SNN	H2-10-1	105	winpgng-tn	90	1
SNN	H2-10-2	31	-Issggsyty	29	1
KMC	H2-10-1	152	winpgng-t-	128	1
KMC	H2-10-2	40	-Issggsyty	38	2
KMC	H2-10-3	8	eilpgsgstk	8	1
KMC	H2-10-4	5	-Ispaggytd	5	2
KMC	H2-10-5	4	yissggggTy	4	2
KM	H2-10-1	171	-iypgng-t-	143	10
KM	H2-10-2	56	-issgggyty	53	13
2017 Dataset					
	L1-10	229	sASSsvsymh	42	9
DBSCAN	L1-10-1	86	SASSsVsYmh	22	1
DBSCAN	L1-10-2	41	SASSSVSYIH	1	1
LMAX	L1-10-1	93	sASSsVsYMh	25	1
LMAX	L1-10-2	37	SASSSVSYIH	1	1
SNN	L1-10-1	128	sASSsVsymh	35	1
SNN	L1-10-2	80	SASSSVSYiH	4	3
KMC	L1-10-1	135	sASSsVsymh	36	2
KMC	L1-10-2	75	SASSSVSYIH	2	1
KMC	L1-10-3	5	KASQGGNAMT	1	1
KMC	L1-10-4	4	SASSSVnYMh	2	1
KMC	L1-10-5	4	kASQGGNSLs	2	1
KM	L1-10-1	139	sASSsVsymh	37	4
KM	L1-10-2	80	SASSSVSYiH	4	3
KM	L1-10-3	10	kasqggnamt	4	2
	L1-11	1839	rasqsissyla	338	86
DBSCAN	L1-11-1	798	rASqsistyla	153	3
DBSCAN	L1-11-2	184	rASqdIsnyla	56	1
LMAX	L1-11-1	847	rASqsistyla	165	4

Run Info	Cluster	Size	Consensus Seq	Unique Seqs	No. UGS
LMAX	L1-11-2	234	rASqdIsnyla	68	1
SNN	L1-11-1	201	rASqdIsnyla	67	1
SNN	L1-11-2	12	RASeniySYLt	3	1
KMC	L1-11-1	986	rASqsistyla	187	9
KMC	L1-11-2	359	rASqdIsnyla	98	5
KMC	L1-11-3	84	gGnnigsksvh	20	2
KMC	L1-11-4	57	rASqsissyla	25	6
KMC	L1-11-5	55	sGDsLrskYas	20	4
KM	L1-11-1	1209	rASqsissyla	219	44
KM	L1-11-2	427	rASqdIsnyla	110	18
KM	L1-11-3	203	sgdnlgskyvs	51	29
	L1-12	295	raSqsvssnyla	57	26
DBSCAN	L1-12-1	80	RASQSVgnnkLA	7	2
DBSCAN	L1-12-2	47	rASsSVsSsylh	18	1
LMAX	L1-12-1	83	RASQSVgnnkLA	8	2
LMAX	L1-12-2	56	rASsSVsSsylh	19	1
SNN	L1-12-1	62	rASssvsssylh	24	2
SNN	L1-12-2	56	RASQSVgnnkLA	7	2
KMC	L1-12-1	116	rASqsvsssyla	33	5
KMC	L1-12-2	91	RASQSVgnnkLA	9	1
KMC	L1-12-3	24	RaShSIrsrrVA	3	1
KMC	L1-12-4	17	TLssQHSTYTIe	5	2
KMC	L1-12-5	10	rASqsVgnnklA	3	2
KM	L1-12-1	115	rASqsvsssyla	36	8
KM	L1-12-2	107	RASQSVgnnkLA	12	4
KM	L1-12-3	29	RaSqSirsnrva	7	4
KM	L1-12-4	18	tlssqhStytie	6	3
KM	L1-12-5	14	RASQSlgssYLA	3	4
	L1-13	256	sgSssnignnyVs	65	22
DBSCAN	L1-13-1	94	SGSsSNIGsntVn	33	2
DBSCAN	L1-13-2	32	tRSsGsiasnYVq	6	1
LMAX	L1-13-1	105	SGSsSNIGsntVn	35	2
LMAX	L1-13-2	33	tRSsGsiasnYVq	7	1
SNN	L1-13-1	115	SGSsSNIGnnyVn	37	2
SNN	L1-13-2	28	TRSSGsiasnYVQ	7	1
KMC	L1-13-1	152	SGSsSNIGnnyVn	46	4
KMC	L1-13-2	41	TRSSGsiasnYVq	9	2
KMC	L1-13-3	19	qssqsVYnknyLa	6	1

Run Info	Cluster	Size	Consensus Seq	Unique Seqs	No. UGS
KMC	L1-13-4	11	SGSSSNVGNGYVS	1	1
KMC	L1-13-5	7	TGSSGSIASNYVQ	1	1
KM	L1-13-1	186	SGsssNiGnnyVs	50	15
KM	L1-13-2	70	trSsgsiasnyvq	15	7
	L1-14	367	tgsssdvgtynyvs	56	73
DBSCAN	L1-14-1	115	RSStGAVTtsNyAn	12	2
DBSCAN	L1-14-2	47	tGTSsDVGgynyVS	12	6
LMAX	L1-14-1	115	RSStGAVTTsNyAn	12	2
LMAX	L1-14-2	46	tGTSsDVGGynyVS	11	6
SNN	L1-14-1	110	RSStGAVTTsNyAn	11	2
SNN	L1-14-2	32	tGTSsDVGgydsVS	10	7
KMC	L1-14-1	121	RSStGAVTTsNyAn	13	3
KMC	L1-14-2	85	tGTSsDVGgynyVS	18	10
KMC	L1-14-3	37	rSstGAvTtsnyAn	8	4
KMC	L1-14-4	25	tGTSsDvGgynyVS	10	2
KMC	L1-14-5	18	TGsSSNIGagYdVH	6	3
KM	L1-14-1	201	tGtSsdvGgynyVs	40	62
KM	L1-14-2	166	RSStGAVTtsNyAn	17	11
	L1-15	205	rASesVdtygnsfmh	62	31
DBSCAN	L1-15-1	108	rASesVdyyGnSfmh	36	4
DBSCAN	L1-15-2	17	RASQSVSTSSYSYMh	3	1
LMAX	L1-15-1	107	rASesVdyyGnSfmh	37	3
LMAX	L1-15-2	13	RASQSVSTSSYSYMh	3	1
SNN	L1-15-1	114	rASesVdyyGnSfmh	38	5
SNN	L1-15-2	5	RASqrVSsStYSYMH	2	2
KMC	L1-15-1	50	rASesVdtyGnSymh	24	3
KMC	L1-15-2	50	rASesVD-yGnSfmh	21	4
KMC	L1-15-3	25	rASqsVdyyGdsfmn	13	5
KMC	L1-15-4	25	RASqsVStSsYsYMh	7	1
KMC	L1-15-5	17	RASESVDnfGkSFMh	2	2
KM	L1-15-1	203	rASesVdtygnSfmh	61	29
KM	L1-15-2	2	RASkdiprSgsGYma	2	2
	L1-16	516	rSSqslvhsnGnTYLe	114	40
DBSCAN	L1-16-1	433	rSSqslvhsnGnTYLe	97	4
DBSCAN	L1-16-2	7	RSSQYIVHRNGNTYLE	1	1
LMAX	L1-16-1	422	rSSqslvhsnGnTYLe	94	4
LMAX	L1-16-2	7	RSSQYIVHRNGNTYLE	1	1
SNN	L1-16-1	371	rSSqslvhsnGnTYLe	85	3

Run Info	Cluster	Size	Consensus Seq	Unique Seqs	No. UGS
SNN	L1-16-2	8	RsSQyIVhrNGnTYLE	3	2
KMC	L1-16-1	347	rSSqSlvhsnGnTYLe	83	5
KMC	L1-16-2	58	rsSQslvhsnGnTyLe	22	6
KMC	L1-16-3	22	rSSqSllhsnGntyLe	13	3
KMC	L1-16-4	14	RSSQsiVHrNGNTYLe	5	2
KMC	L1-16-5	14	RSsqslvhsnGntYLd	11	2
KM	L1-16-1	515	rSSqslvhsnGnTYLe	114	39
KM	L1-16-2	1	RSSQSIVHSNGNSYLE	1	1
	L1-17	249	KSSQSllnSrnqKNYLa	63	22
DBSCAN	L1-17-1	152	KSSQSllnSrnqKNYLA	44	1
DBSCAN	L1-17-2	6	KSSQSVLYSSNQKNYLA	1	1
LMAX	L1-17-1	201	KSSQsllnSrnqKNYLa	59	1
LMAX	L1-17-2	10	KSSQSlLniSNqKnYLA	2	2
SNN	L1-17-1	171	KSSQsllnSrnqKNYLa	54	1
SNN	L1-17-2	6	KSSQSLLdsrNQKNyLA	3	2
KMC	L1-17-1	202	KSSQsllnSrnqKNYLa	60	3
KMC	L1-17-2	11	kSSQSllyssnqKnyLA	3	1
KMC	L1-17-3	9	KSSQSLLnsrnqKNyLA	4	2
KMC	L1-17-4	6	KSSQSVLYSSNQKNYLA	1	1
KMC	L1-17-5	3	kSSqslLnSrnrKnYLA	3	1
KM	L1-17-1	239	KSSQSllnSrnqKNYLa	62	18
KM	L1-17-2	10	kSSqslLnSrnrknYLa	6	8
	L2-8	4038	y-asnras	555	87
DBSCAN	L2-8-1	2880	y-asnras	442	2
DBSCAN	L2-8-2	21	YdtsklaS	8	2
LMAX	L2-8-1	3420	y-asnras	486	5
LMAX	L2-8-2	107	ydtskrps	47	2
SNN	L2-8-1	132	ydtskrps	57	5
SNN	L2-8-2	69	ygasnras	27	4
SNN	L2-8-3	61	yytsnlas	27	2
SNN	L2-8-4	48	KYASeSIS	6	2
SNN	L2-8-5	13	yenskRPS	10	2
KMC	L2-8-1	2221	y-asnlas	381	3
KMC	L2-8-2	1062	y-vsnras	259	3
KMC	L2-8-3	100	Y-asnlas	51	1
KMC	L2-8-4	99	y-askrps	48	7
KMC	L2-8-5	94	yginnras	29	7
KM	L2-8-1	3951	y-asnras	549	65

Run Info	Cluster	Size	Consensus Seq	Unique Seqs	No. UGS
KM	L2-8-2	87	ysvsnrps	40	24
	L3-5	99	Qqyef	15	1
DBSCAN	L3-5-1	43	QQyEf	6	1
DBSCAN	L3-5-2	6	QQSRT	1	1
LMAX	L3-5-1	47	QQyEf	6	1
LMAX	L3-5-2	6	QQSRT	1	1
SNN	L3-5-1	43	QQyEf	6	1
SNN	L3-5-2	4	QQsrs	2	1
KMC	L3-5-1	95	Qqyef	15	1
KMC	L3-5-2	4	QQfEF	2	1
KM	L3-5-1	74	QqyEf	12	1
KM	L3-5-2	25	QQset	7	1
	L3-8	236	qqsynlrt	71	18
DBSCAN	L3-8-1	71	kQsynlrT	14	1
DBSCAN	L3-8-2	24	mQtTHaPT	3	1
LMAX	L3-8-1	73	kQsynlrT	15	1
LMAX	L3-8-2	27	mQtTHaPT	5	1
LMAX	L3-8-3	13	QHLIGLRS	1	1
SNN	L3-8-1	101	kQsynlrT	24	1
SNN	L3-8-2	7	sQsthvPT	3	1
KMC	L3-8-1	116	kQsynlrT	28	2
KMC	L3-8-2	59	mQtthaPT	23	3
KMC	L3-8-3	18	Qhliglrs	3	2
KMC	L3-8-4	10	QqansffT	3	3
KMC	L3-8-5	9	hqyhrglt	6	3
KM	L3-8-1	119	kQsynlrT	30	4
KM	L3-8-2	59	qQtthaPT	23	4
KM	L3-8-3	23	qhliglrs	8	3
KM	L3-8-4	14	qqansffT	6	5
KM	L3-8-5	11	qQwgsnpT	6	4
	L3-9-cis7	2269	qqyss-PyT	525	28
DBSCAN	L3-9-cis7-1	2009	qQyss-PyT	475	4
DBSCAN	L3-9-cis7-2	26	QHfwstPrT	9	1
LMAX	L3-9-cis7-1	1989	qQyss-PyT	472	3
LMAX	L3-9-cis7-2	23	QHfwstPRT	8	1
SNN	L3-9-cis7-1	979	qQyns-PyT	312	2
SNN	L3-9-cis7-2	19	QQlhfyPhT	5	1
SNN	L3-9-cis7-3	6	qQldflPyT	5	1

Run Info	Cluster	Size	Consensus Seq	Unique Seqs	UGS
KMC	L3-9-cis7-1	1877	qQyss-PyT	455	7
KMC	L3-9-cis7-2	162	qqhwstPrT	70	7
KMC	L3-9-cis7-3	102	qqlhfyPhT	54	2
KMC	L3-9-cis7-4	85	qqsss-PyT	52	4
KMC	L3-9-cis7-5	20	QqyysyPyT	12	2
KM	L3-9-cis7-1	2266	qqyss-PyT	523	25
KM	L3-9-cis7-2	3	QQYdgyPLT	3	3
	L3-9	568	qqyysnhwt	127	63
DBSCAN	L3-9-1	141	qQygqslsT	33	3
DBSCAN	L3-9-2	87	ALWySNHwV	5	4
LMAX	L3-9-1	148	qQygqslsT	35	2
LMAX	L3-9-2	91	ALWysNhwV	7	5
SNN	L3-9-1	115	ALWySNhwV	11	8
SNN	L3-9-2	41	alwdsnhwv	16	6
KMC	L3-9-1	104	qQygqslsT	27	4
KMC	L3-9-2	92	qQysdippT	39	4
KMC	L3-9-3	53	QhyagysaT	6	3
KMC	L3-9-4	51	ALWySNHwV	6	3
KMC	L3-9-5	44	qlwdsnhwv	19	3
KM	L3-9-1	315	alwysnhwv	56	35
KM	L3-9-2	253	qqygqspsT	80	31
	L3-10	257	ssydssdnfv	56	53
DBSCAN	L3-10-1	81	ssytssshrv	15	1
DBSCAN	L3-10-2	14	qvwdsfstfv	5	2
LMAX	L3-10-1	81	ssytssshrv	16	1
LMAX	L3-10-2	16	qvwdsfstfv	6	2
LMAX	L3-10-3	11	QQSSPSFLIT	2	1
SNN	L3-10-1	78	ssytssshrv	18	7
SNN	L3-10-2	2	SSYEGSDNFV	1	2
SNN	L3-10-3	2	SSYEGSDNFV	1	2
KMC	L3-10-1	60	ssltsrstrv	14	2
KMC	L3-10-2	27	qsydSsnhvv	5	2
KMC	L3-10-3	25	qvwdsfstfv	12	1
KMC	L3-10-4	18	qqsspsflit	5	2
KMC	L3-10-5	17	csYatsrtlv	8	4
KM	L3-10-1	122	ssytssstrv	27	15
KM	L3-10-2	40	sSyegsdnfv	7	12
KM	L3-10-3	35	qswdsssnfv	16	7

Run Info	Cluster	Size	Consensus Seq	Unique Seqs	UGS
KM	L3-10-4	28	qaydsstavv	9	10
KM	L3-10-5	21	ssYats-tlv	9	8
	L3-11	289	qswdsslsgvv	84	36
DBSCAN	L3-11-1	161	aawDdslngvv	53	2
DBSCAN	L3-11-2	13	QQYPYYSSLIT	1	1
LMAX	L3-11-1	152	aawDdslngvv	52	2
LMAX	L3-11-2	13	QQYPYYSSLIT	1	1
SNN	L3-11-1	112	aawDdslngvv	44	2
SNN	L3-11-2	4	SSYaSgsTPri	2	2
KMC	L3-11-1	47	aaWDDsLnGwV	14	2
KMC	L3-11-2	21	stwDdSlsavv	11	2
KMC	L3-11-3	20	qvwDsssdhvv	12	2
KMC	L3-11-4	18	QVWDsSsdhyv	7	1
KMC	L3-11-5	17	aaWDdSLnvhv	9	1
KM	L3-11-1	243	qswdsslsgvv	75	20
KM	L3-11-2	46	ssyadsstpav	11	17
	L3-12	108	htwdsslsanwv	22	24
DBSCAN	L3-12-1	15	HIWDSRrPTnWV	3	1
DBSCAN	L3-12-2	13	SSRDKSGSRLSV	1	1
LMAX	L3-12-1	16	HIWDSRrPTnWV	3	1
LMAX	L3-12-2	13	SSRDKSGSRLSV	1	1
LMAX	L3-12-3	10	QQGYtsSnvnNa	2	1
LMAX	L3-12-4	8	GTWDSSLSAYVV	1	1
LMAX	L3-12-5	7	ATWDSGLSADWV	1	1
SNN	L3-12-1	28	atWDssLsayvV	7	8
SNN	L3-12-2	7	hiwdsrrptnwv	4	3
KMC	L3-12-1	18	HiWDSRrptnWV	4	2
KMC	L3-12-2	17	aTWDSgLSAdvV	3	2
KMC	L3-12-3	13	SSRDKSGSRLSV	1	1
KMC	L3-12-4	12	HmwDsRsGfsWs	2	1
KMC	L3-12-5	10	QQGYtsSnvnNa	2	1
KM	L3-12-1	25	atWDssLsadwV	7	6
KM	L3-12-2	20	HiWDSRrptnWv	4	2
KM	L3-12-3	14	lggypaasyrta	5	6
KM	L3-12-4	14	SSRDKSGSRLSV	1	2
KM	L3-12-5	12	AaWDdslsahvV	4	5
KM	L3-12-6	12	HmwDsRsGfsWs	2	1
	H1-13	3716	aaSGftfssywmh	922	204

Run Info	Cluster	Size	Consensus Seq	Unique Seqs	UGS
DBSCAN	H1-13-1	2330	kaSGftftsywmh	611	4
DBSCAN	H1-13-2	60	AASGFDYSRYWMS	2	1
DBSCAN	H1-13-3	43	KASGYTFTSDWIH	1	1
LMAX	H1-13-1	2684	aaSGftftsywmh	689	9
LMAX	H1-13-2	69	AASGFDYSRYWMS	3	1
LMAX	H1-13-3	50	KASGYTFTSDWIH	1	1
SNN	H1-13-1	780	kaSGytFtsywmh	278	1
SNN	H1-13-2	115	TVSGFSLtnygVh	31	1
SNN	H1-13-3	84	aasGfnissyyih	34	2
SNN	H1-13-4	51	aASGFtFrnswmh	21	1
SNN	H1-13-5	24	aASgyaytyiyMg	7	6
KMC	H1-13-1	1591	aaSGftFtsywmh	446	8
KMC	H1-13-2	285	aasGfsitdyyih	106	8
KMC	H1-13-3	233	tvSgfsltnygvh	77	5
KMC	H1-13-4	215	kaSGytFtdywmh	121	9
KMC	H1-13-5	151	aaSGftFssywms	85	11
KM	H1-13-1	3525	aaSGftfssywmh	874	160
KM	H1-13-2	191	aaSgstfssyamg	86	52
	H1-14	180	tvtGysitsgyawn	36	29
DBSCAN	H1-14-1	57	tVtgYSITsdYaWn	18	3
DBSCAN	H1-14-2	12	TVTGYSITSDFAWN	2	1
LMAX	H1-14-1	59	tVtgYSITsdYaWn	18	3
LMAX	H1-14-2	13	TVTGYSITSdfaWn	2	1
LMAX	H1-14-3	9	AVSGYSIsSGYyWg	2	1
LMAX	H1-14-4	8	KAYGVNTFGLYAVN	1	1
LMAX	H1-14-5	5	TVTGYSITTNYAWT	1	2
SNN	H1-14-1	151	tVtGySItsgyaWn	32	24
SNN	H1-14-2	23	kAyGvntfglYavn	4	4
KMC	H1-14-1	24	tVtGYSITsdYaWn	10	2
KMC	H1-14-2	19	tvtGYSItSgysWn	12	2
KMC	H1-14-3	15	kvsGysittnYawc	4	2
KMC	H1-14-4	14	KAYGVNTFGLYAVN	1	1
KMC	H1-14-5	13	aVsGYSItSGYsWg	4	2
KM	H1-14-1	75	tVtGYSITsdYaWn	24	9
KM	H1-14-2	38	tvtGySitsnyawn	9	7
KM	H1-14-3	23	kAyGvntfglYavn	4	4
KM	H1-14-4	20	sVtGYsItsGYsWh	4	7
KM	H1-14-5	18	aVsGysItsGYsWg	8	6

Run Info	Cluster	Size	Consensus Seq	Unique Seqs	UGS
	H1-15	213	sfSGfSlstsgmgvg	49	36
DBSCAN	H1-15-1	96	sFSGFSLstsGmgVg	20	1
DBSCAN	H1-15-2	9	AISGDSVSSYNAVWN	1	1
LMAX	H1-15-1	100	sfSGFSLstsGmgVg	21	2
LMAX	H1-15-2	9	AISGDSVSSYNAVWN	1	1
SNN	H1-15-1	116	sfSGFSLstsGmgVg	24	2
SNN	H1-15-2	3	tvSGgSissgdyywS	2	2
KMC	H1-15-1	38	SFSGFSLSDFGVGVG	1	2
KMC	H1-15-2	23	sFSGFSLSTsGMgVg	10	1
KMC	H1-15-3	14	aiSGdSVSSynavWn	5	2
KMC	H1-15-4	13	avSGdStaacnsfwg	4	3
KMC	H1-15-5	12	SFSGFSLtTSGiGVG	2	1
KM	H1-15-1	211	sfSGfSlstsgmgvg	48	34
KM	H1-15-2	2	TVSGGSISSSSYYWG	1	2
	H2-9	944	yIwysgsty	237	31
DBSCAN	H2-9-1	801	yIwysgsty	205	3
DBSCAN	H2-9-2	8	YIRSSVITR	1	1
LMAX	H2-9-1	792	yIwysgsty	203	2
LMAX	H2-9-2	9	YVHdSGDTN	3	1
LMAX	H2-9-3	8	YIRSSVITR	1	1
SNN	H2-9-1	815	yIwysgsty	214	4
SNN	H2-9-2	15	yihssgsTn	9	4
SNN	H2-9-3	6	hIywdgdtr	5	1
KMC	H2-9-1	575	yIwsdgsty	147	2
KMC	H2-9-2	89	yisysGsty	51	2
KMC	H2-9-3	53	yIwwdgdtr	26	3
KMC	H2-9-4	45	vittgGstn	28	1
KMC	H2-9-5	44	yinwsgdtn	22	1
KM	H2-9-1	936	yIwysgsty	235	26
KM	H2-9-2	8	siyYsGgas	3	5
	H2-10	2982	-ispggg-ty	813	104
DBSCAN	H2-10-1	1223	-inp-ng-tn	398	1
DBSCAN	H2-10-2	539	-Issdgstty	162	1
LMAX	H2-10-1	1233	winp-ng-tn	406	1
LMAX	H2-10-2	477	-Issdgs-ty	152	1
SNN	H2-10-1	1015	winpyng-tn	345	1
SNN	H2-10-2	116	aIsgsGgsty	56	1
SNN	H2-10-3	109	eilpgsgstn	41	5

Run Info	Cluster	Size	Consensus Seq	Unique Seqs	UGS
SNN	H2-10-4	51	yIssdgsniy	29	1
SNN	H2-10-5	39	EInpvsstIn	6	1
KMC	H2-10-1	1029	wInp-ng-tn	335	3
KMC	H2-10-2	718	-Issdgstty	198	10
KMC	H2-10-3	299	giipilgitn	111	6
KMC	H2-10-4	142	wInpysGgtn	73	6
KMC	H2-10-5	130	riypgng-tn	75	4
KM	H2-10-1	1977	-inp-ng-tn	570	59
KM	H2-10-2	1005	-Isssgs-ty	267	57
	H2-12	222	fIrnkangytte	51	16
DBSCAN	H2-12-1	119	fIRnKangytTe	29	1
DBSCAN	H2-12-2	10	RiKSrTDGGttD	3	1
LMAX	H2-12-1	144	fIrnkangytte	36	1
LMAX	H2-12-2	11	RIKSrTDGGTTD	3	1
SNN	H2-12-1	189	fIrnkangytte	43	2
SNN	H2-12-2	2	ASRNKGNKYTTE	1	2
KMC	H2-12-1	65	FIRNKakGYTtE	12	1
KMC	H2-12-2	53	rIrskannyaTy	19	1
KMC	H2-12-3	30	eIRlKSdNYATh	9	1
KMC	H2-12-4	18	rIksktdGGtTd	5	1
KMC	H2-12-5	12	rItgpgeGwsvd	4	1
KM	H2-12-1	212	fIrnkangytte	48	10
KM	H2-12-2	10	eirlnsdnyath	5	6
	H3-5	59	trmdy	15	8
DBSCAN	H3-5-1	19	trmDY	3	2
DBSCAN	H3-5-2	10	tsaGr	2	1
LMAX	H3-5-1	19	trmDY	3	2
LMAX	H3-5-2	10	tsaGr	2	1
SNN	H3-5-1	21	traGr	6	2
SNN	H3-5-2	12	TRMDY	1	1
KMC	H3-5-1	18	trmDY	2	1
KMC	H3-5-2	12	tsaGr	3	1
KMC	H3-5-3	10	aRwgY	3	1
KMC	H3-5-4	8	TRKDY	1	1
KMC	H3-5-5	5	tsVhQ	3	1
KM	H3-5-1	32	trmdy	6	5
KM	H3-5-2	12	tsaGr	3	1
KM	H3-5-3	10	aRwgY	3	1

Run Info	Cluster	Size	Consensus Seq	Unique Seqs	UGS
KM	H3-5-4	5	tsVhQ	3	1
	H3-6	82	arg-dy	29	12
DBSCAN	H3-6-1	21	aRgtty	9	1
DBSCAN	H3-6-2	14	asgmey	5	1
LMAX	H3-6-1	21	aRgtty	9	1
LMAX	H3-6-2	9	vsgmey	3	1
SNN	H3-6-1	22	argfey	8	2
SNN	H3-6-2	22	arGtdy	9	1
KMC	H3-6-1	36	argldy	14	1
KMC	H3-6-2	23	argfey	8	2
KMC	H3-6-3	8	AryggY	3	1
KMC	H3-6-4	4	KASGSS	1	1
KMC	H3-6-5	3	TTHFDY	1	1
KM	H3-6-1	37	argldy	15	2
KM	H3-6-2	28	argfdy	10	4
KM	H3-6-3	11	Aryggy	4	3
KM	H3-6-4	6	kasgss	3	3
	H3-7	108	anwdgdy	31	19
DBSCAN	H3-7-1	35	AnWdGDy	4	1
DBSCAN	H3-7-2	4	AsYrfiY	2	1
DBSCAN	H3-7-3	4	ATYGVAY	1	1
LMAX	H3-7-1	39	AnWdGDy	5	1
LMAX	H3-7-2	5	Asyrfiy	3	1
LMAX	H3-7-3	4	ATYGVAY	1	1
SNN	H3-7-1	22	asysfiy	11	4
SNN	H3-7-2	21	ANWDGDY	2	1
KMC	H3-7-1	62	anwdgDy	13	3
KMC	H3-7-2	12	sgwsfly	7	1
KMC	H3-7-3	10	asyrfiy	5	1
KMC	H3-7-4	4	ATYGVAY	1	1
KMC	H3-7-5	3	nadgeDf	3	1
KM	H3-7-1	70	anwdgDy	17	6
KM	H3-7-2	11	sgwsfly	6	1
KM	H3-7-3	10	asyrfiy	5	1
KM	H3-7-4	6	atyGvaY	2	3
KM	H3-7-5	5	krgysgh	3	3
	H3-8	54	arg-gfdy	19	16
DBSCAN	H3-8-1	8	ARGQGRPY	1	1

Run Info	Cluster	Size	Consensus Seq	Unique Seqs	UGS
DBSCAN	H3-8-2	7	ARGFGTDF	1	1
DBSCAN	H3-8-3	6	NVNVGFEY	1	1
LMAX	H3-8-1	8	ARGQGRPY	1	1
LMAX	H3-8-2	6	NVNVGFEY	1	1
LMAX	H3-8-3	6	ARGFGTDF	1	1
SNN	H3-8-1	20	aRgfgtdy	8	3
SNN	H3-8-2	11	argqGrpY	2	3
KMC	H3-8-1	16	aRgfgtDf	6	1
KMC	H3-8-2	9	ARGQGRPY	1	1
KMC	H3-8-3	4	EREGGVNN	1	1
KMC	H3-8-4	4	NVNVGFEY	1	1
KMC	H3-8-5	4	TRSLNMAY	1	1
KM	H3-8-1	28	aRgfgtdy	13	8
KM	H3-8-2	10	nvnvgfeY	2	3
KM	H3-8-3	9	ARGQGRPY	1	1
KM	H3-8-4	4	EREGGVNN	1	1
KM	H3-8-5	2	ARefddai	2	2
	H3-12	558	arg-ydyyafdy	160	138
DBSCAN	H3-12-1	291	arg-ydyyafdy	89	46
DBSCAN	H3-12-2	32	ARfaGgwgaydv	2	2
LMAX	H3-12-1	440	arg-ydyyafdy	131	79
LMAX	H3-12-2	11	AGLLWYDGGAGS	1	1
LMAX	H3-12-3	11	agltgsshaddy	7	7
LMAX	H3-12-4	10	vsygyggDrFsY	2	5
LMAX	H3-12-5	8	syfdsddyameY	4	1
SNN	H3-12-1	88	argdyy-gafDy	38	12
SNN	H3-12-2	18	VRPLYDYYAMDY	1	1
KMC	H3-12-1	40	VRPLYDYYAMDY	2	1
KMC	H3-12-2	28	arggygyyyfdY	10	3
KMC	H3-12-3	27	ARgrwyrraldy	10	5
KMC	H3-12-4	25	ARFAGGWGAYDV	1	1
KMC	H3-12-5	23	arydgiygelDf	8	2
KM	H3-12-1	113	argdyyygamdy	39	15
KM	H3-12-2	60	arg-sdgrafdy	23	18
KM	H3-12-3	45	arggyggywfdy	17	15
KM	H3-12-4	44	vRplydyYAMDY	4	4
KM	H3-12-5	43	ar--glyy-fdy	23	21
	H3-14	340	arlyygygywyfdy	99	102

Run Info	Cluster	Size	Consensus Seq	Unique Seqs	UGS
DBSCAN	H3-14-1	72	ARLYYGYGYWYFDV	1	2
DBSCAN	H3-14-2	20	AAGGyeLrdrtYGq	5	1
LMAX	H3-14-1	72	ARLYYGYGYWYFDV	1	2
LMAX	H3-14-2	18	AAGGYeLRDRTYGq	4	1
SNN	H3-14-1	83	ARlyYgyGyWYFDV	8	7
SNN	H3-14-2	4	AKVKFYDPTAPNDY	1	1
SNN	H3-14-3	4	AAGGYELRDRTYGQ	1	1
KMC	H3-14-1	74	ARLYYGYGYWYFDV	2	3
KMC	H3-14-2	30	AkhgnyyyysgmDv	5	6
KMC	H3-14-3	20	AAGGyeLrdrtYGq	5	1
KMC	H3-14-4	19	trgkncdynwdfeh	7	4
KMC	H3-14-5	16	arrefysyyspmDy	7	6
KM	H3-14-1	79	ARLYYGYGYWYFDV	5	7
KM	H3-14-2	51	ARg-yygg-wyfdy	18	17
KM	H3-14-3	46	ArhggyyyysgmDv	16	16
KM	H3-14-4	32	ar-s-yyydyamdy	17	19
KM	H3-14-5	29	argkncdynwdfeh	14	10
	H3-15	323	ardgyyyrsgyyfdy	101	127
DBSCAN	H3-15-1	48	ARDgdYyrygryfDy	6	7

APPENDIX B

DBSCAN OUTLIER PERCENTAGES

CDR	DBSCAN	LMAX	SNN
2011 Dataset			
L1-11	43	45	31
L1-16	50	45	38
L2-8	34	31	51
L3-9-cis7	17	21	40
H1-13	22	18	64
H2-9	42	38	40
H2-10	44	40	40
H3-12	44	40	38
2017 Dataset			
L1-10	45	43	9
L1-11	47	41	88
L1-12	57	53	60
L1-13	51	46	44
L1-14	56	56	61
L1-15	39	41	42
L1-16	15	17	27
L1-17	37	15	29
L2-8	28	13	88
L3-5	51	46	53
L3-8	60	52	54
L3-9-cis7	10	11	56
L3-9	60	58	73
L3-10	63	58	68
L3-11	40	43	60
L3-12	74	37	68

CDR	DBSCAN	LMAX	SNN
(2017 Dataset, continued)			
H1-13	35	25	69
H1-14	62	48	3
H1-15	51	49	44
H2-9	14	14	11
H2-10	41	43	52
H2-12	42	30	14
H3-5	51	51	44
H3-6	57	63	46
H3-7	60	56	60
H3-8	61	63	43
H3-12	42	11	81
H3-14	73	74	73

APPENDIX C

INPUT PARAMETERS PART 1

Cluster	DB-linkf	DB-coref	LMAX-linkf	LMAX-coref
2011 Dataset				
L1-11	0.05	0.07	0.05	0.07
L2-8	0.05	0.46	0.05	0.51
L3-9-cis7	0.20	0.79	0.10	0.72
L3-9	0.05	0.33	0.05	0.61
H1-13	0.15	0.78	0.20	0.82
H2-9	0.05	0.48	0.05	0.47
H2-10	0.05	0.26	0.05	0.31
H3-12	0.05	0.56	0.05	0.60
2017 Dataset				
L1-10	0.05	0.06	0.05	0.12
L1-11	0.05	0.25	0.05	0.30
L1-12	0.05	0.25	0.05	0.26
L1-13	0.05	0.20	0.05	0.22
L1-14	0.05	0.16	0.05	0.15
L1-15	0.15	0.53	0.10	0.46
L1-16	0.45	0.82	0.25	0.80
L1-17	0.05	0.53	0.50	0.81
L2-8	0.05	0.60	0.30	0.80
L3-5	0.05	0.38	0.05	0.39
L3-8	0.05	0.21	0.05	0.31
L3-9-cis7	0.15	0.81	0.10	0.80
L3-9	0.05	0.11	0.05	0.12
L3-10	0.05	0.32	0.05	0.37
L3-11	0.15	0.53	0.10	0.48
L3-12	0.05	0.14	0.05	0.56

Cluster	DB-linkf	DB-coref	LMAX-linkf	LMAX-coref
(2017 Dataset, continued)				
H1-13	0.05	0.50	0.10	0.65
H1-14	0.05	0.26	0.05	0.46
H1-15	0.05	0.41	0.05	0.43
H2-9	0.10	0.80	0.10	0.82
H2-10	0.05	0.10	0.05	0.10
H2-12	0.05	0.41	0.05	0.52
H3-5	0.10	0.34	0.10	0.32
H3-6	0.05	0.15	0.05	0.16
H3-7	0.05	0.36	0.05	0.38
H3-8	0.05	0.31	0.05	0.24
H3-12	0.05	0.36	0.05	0.79
H3-14	0.05	0.26	0.05	0.26

APPENDIX D

INPUT PARAMETERS PART 2

Cluster	SNN-linkf	SNN-coref	SNN-k	KMC-k	KM-k
2011 Dataset					
L1-11	0.45	0.42	20	7	3
L2-8	0.25	0.33	20	23	2
L3-9-cis7	0.30	0.33	20	4	2
L3-9	0.20	0.67	20	30	19
H1-13	0.24	0.21	20	22	2
H2-9	0.30	0.42	20	4	2
H2-10	0.33	0.30	20	17	2
H3-12	0.24	0.25	20	42	13
2017 Dataset					
L1-10	0.68	0.60	100	9	3
L1-11	0.02	0.05	100	43	3
L1-12	0.07	0.22	100	17	6
L1-13	0.06	0.50	100	17	2
L1-14	0.08	0.05	100	49	2
L1-15	0.10	0.50	100	18	2
L1-16	0.37	0.62	100	31	2
L1-17	0.08	0.44	100	18	2
L2-8	0.01	0.07	100	47	2
L3-5	0.29	0.36	20	2	2
L3-8	0.11	0.37	100	11	6
L3-9-cis7	0.20	0.11	100	16	2
L3-9	0.09	0.05	100	39	2
L3-10	0.02	0.32	100	40	6
L3-11	0.05	0.27	100	41	2
L3-12	0.05	0.21	100	18	9

Cluster	SNN-linkf	SNN-coref	SNN-k	KMC-k	KM-k
(2017 Dataset, continued)					
H1-13	0.13	0.06	100	95	2
H1-14	0.73	0.81	100	31	6
H1-15	0.09	0.52	100	43	2
H2-9	0.40	0.78	100	28	2
H2-10	0.12	0.16	100	58	2
H2-12	0.26	0.86	100	20	2
H3-5	0.30	0.22	20	8	4
H3-6	0.30	0.26	20	11	4
H3-7	0.23	0.26	20	14	8
H3-8	0.26	0.24	20	17	6
H3-12	0.02	0.06	100	88	17
H3-14	0.04	0.22	100	60	13